

# Introduction to Regression



(Perhaps surprisingly, this example is still relevant)



<http://www.nature.com/ejhg/journal/v17/n8/full/ejhg20095a.html>

Predicting height: the Victorian approach beats modern genomics

## Questions for this class

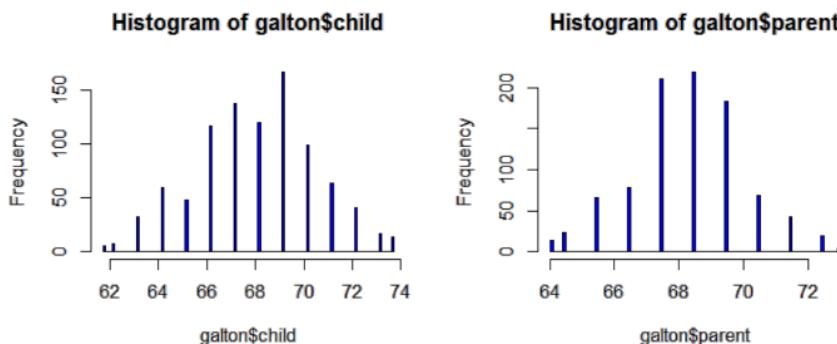
- Consider trying to answer the following kinds of questions:
  - To use the parents' heights to predict childrens' heights.
  - To try to find a parsimonious, easily described mean relationship between parent and children's heights.
  - To investigate the variation in childrens' heights that appears unrelated to parents' heights (residual variation).
  - To quantify what impact genotype information has beyond parental height in explaining child height.
  - To figure out how/whether and what assumptions are needed to generalize findings beyond the data in question.
  - Why do children of very tall parents tend to be tall, but a little shorter than their parents and why children of very short parents tend to be short, but a little taller than their parents? (This is a famous question called 'Regression to the mean'.)

# Galton's Data

- Let's look at the data first, used by Francis Galton in 1885.
- Galton was a statistician who invented the term and concepts of regression and correlation, founded the journal Biometrika, and was the cousin of Charles Darwin.
- You may need to run `install.packages("UsingR")` if the `UsingR` library is not installed.
- Let's look at the marginal (parents disregarding children and children disregarding parents) distributions first.
  - Parent distribution is all heterosexual couples.
  - Correction for gender via multiplying female heights by 1.08.
  - Overplotting is an issue from discretization.

## Code

```
library(UsingR); data(galton)
par(mfrow=c(1,2))
hist(galton$child,col="blue",breaks=100)
hist(galton$parent,col="blue",breaks=100)
```



## Finding the Middle via Least Squares

- Consider only the children's heights.
  - How could one describe the "middle"?
  - One definition, let  $Y_i$  be the height of child  $i$  for  $i = 1, \dots, n = 928$ , then define the middle as the value of  $\mu$  that minimizes

$$\sum_{i=1}^n (Y_i - \mu)^2$$

- This is physical center of mass of the histogram.
- You might have guessed that the answer  $\mu = \bar{X}$ .

# Experiment

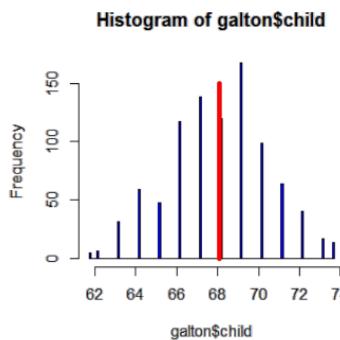
Use R studio's manipulate to see what value of  $\mu$  minimizes the sum of the squared deviations.

```
library(manipulate)
myHist <- function(mu){
  hist(galton$child,col="blue",breaks=100)
  lines(c(mu, mu), c(0, 150),col="red",lwd=5)
  mse <- mean((galton$child - mu)^2)
  text(63, 150, paste("mu = ", mu))
  text(63, 140, paste("MSE = ", round(mse, 2)))
}
manipulate(myHist(mu), mu = slider(62, 74, step = 0.5))
```

## Least Squares Estimate is Empirical Mean

```
hist(galton$child,col="blue",breaks=100)
meanChild <- mean(galton$child)
lines(rep(meanChild,100),seq(0,150,length=100),col="red",lwd=5)
```

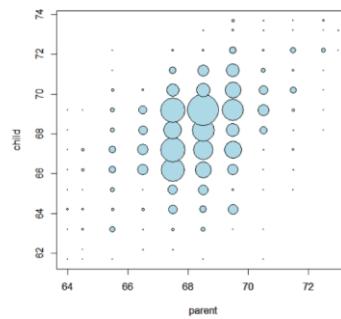
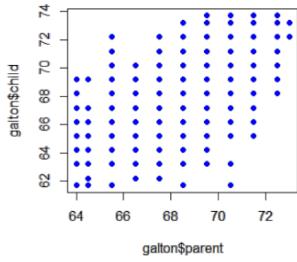
The math follows as:



$$\begin{aligned} \sum_{i=1}^n (Y_i - \mu)^2 &= \sum_{i=1}^n (Y_i - \bar{Y} + \bar{Y} - \mu)^2 \\ &= \sum_{i=1}^n (Y_i - \bar{Y})^2 + 2 \sum_{i=1}^n (Y_i - \bar{Y})(\bar{Y} - \mu) + \sum_{i=1}^n (\bar{Y} - \mu)^2 \\ &= \sum_{i=1}^n (Y_i - \bar{Y})^2 + 2(\bar{Y} - \mu) \sum_{i=1}^n (Y_i - \bar{Y}) + \sum_{i=1}^n (\bar{Y} - \mu)^2 \\ &= \sum_{i=1}^n (Y_i - \bar{Y})^2 + 2(\bar{Y} - \mu)(\sum_{i=1}^n Y_i - n\bar{Y}) + \sum_{i=1}^n (\bar{Y} - \mu)^2 \\ &= \sum_{i=1}^n (Y_i - \bar{Y})^2 + \sum_{i=1}^n (\bar{Y} - \mu)^2 \\ &\geq \sum_{i=1}^n (Y_i - \bar{Y})^2 \end{aligned}$$

## Comparing Children and Parents' Heights

```
plot(galton$parent,galton$child,pch=19,col="blue")
```



# Regression through the Origin

- Suppose that  $X_i$  are the parents' heights.
- Consider picking the slope  $\beta$  that minimizes

$$\sum_{i=1}^n (Y_i - X_i \beta)^2$$

- This is exactly using the origin as a pivot point picking the line that minimizes the sum of the squared vertical distances of the points to the line
- Use R studio's manipulate function to experiment
- Subtract the means so that the origin is the mean of the parent and children's heights

```
myPlot <- function(beta){  
  y <- galton$child - mean(galton$child)  
  x <- galton$parent - mean(galton$parent)  
  freqData <- as.data.frame(table(x, y))  
  names(freqData) <- c("child", "parent", "freq")  
  plot(  
    as.numeric(as.vector(freqData$parent)),  
    as.numeric(as.vector(freqData$child)),  
    pch = 21, col = "black", bg = "lightblue",  
    cex = .15 * freqData$freq,  
    xlab = "parent",  
    ylab = "child"  
  )  
  abline(0, beta, lwd = 3)  
  points(0, 0, cex = 2, pch = 19)  
  mse <- mean( (y - beta * x)^2 )  
  title(paste("beta = ", beta, "mse = ", round(mse, 3)))  
}  
manipulate(myPlot(beta), beta = slider(0.6, 1.2, step = 0.02))
```

## The Solution

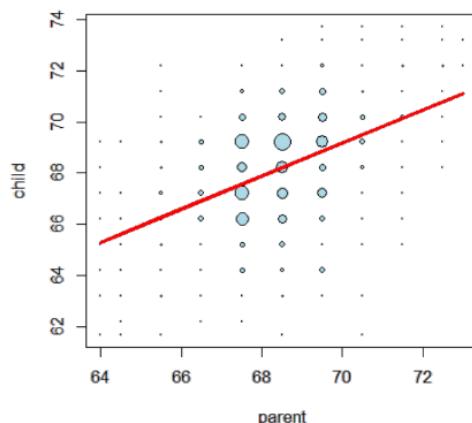
In the next few lectures we'll talk about why this is the solution

```
lm(I(child - mean(child)) ~ I(parent - mean(parent)) - 1, data = galton)
```

```
Call:  
lm(formula = I(child - mean(child)) ~ I(parent - mean(parent)) -  
  1, data = galton)  
  
Coefficients:  
I(parent - mean(parent))  
 0.646
```

## Visualizing the Best Fit Line

Size of points are frequencies at that X, Y combination



# Some Basic Regression Notation and BG

- In this module, we'll cover some basic definitions and notation used throughout the class.
- We will try to minimize the amount of mathematics required for this class.
- No calculus is required.

## Notation for Data

- We write  $X_1, X_2, \dots, X_n$  to describe  $n$  data points.
- As an example, consider the data set  $\{1, 2, 5\}$  then
  - $X_1 = 1, X_2 = 2, X_3 = 5$  and  $n = 3$ .
- We often use a different letter than  $X$ , such as  $Y_1, \dots, Y_n$ .
- We will typically use Greek letters for things we don't know. Such as,  $\mu$  is a mean that we'd like to estimate.
- We will use capital letters for conceptual values of the variables and lowercase letters for realized values.
  - So this way we can write  $P(X_i > x)$ .
  - $X_i$  is a conceptual random variable.
  - $x$  is a number that we plug into.

## Empirical Mean

- Define the empirical mean as
$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i.$$
- Notice if we subtract the mean from data points, we get data that has mean 0. That is, if we define
$$\tilde{X}_i = X_i - \bar{X}.$$
The the mean of the  $\tilde{X}_i$  is 0.

- This process is called "centering" the random variables.
- The mean is a measure of central tendency of the data.
- Recall from the previous lecture that the mean is the least squares solution for minimizing

$$\sum_{i=1}^n (X_i - \mu)^2$$

## Empirical Standard Dev. And Variance

- Define the empirical variance as

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2 = \frac{1}{n-1} \left( \sum_{i=1}^n X_i^2 - n\bar{X}^2 \right)$$

- The empirical standard deviation is defined as  $S = \sqrt{S^2}$ . Notice that the standard deviation has the same units as the data.
- The data defined by  $X_i$ 's have empirical standard deviation 1. This is called "scaling" the data.
- The empirical standard deviation is a measure of spread.
- Sometimes people divide by  $n$  rather than  $n - 1$  (the latter produces an unbiased estimate.)

# Normalization

- The the data defined by

$$Z_i = \frac{X_i - \bar{X}}{S}$$

have empirical mean zero and empirical standard deviation 1.

- The process of centering then scaling the data is called "normalizing" the data.
- Normalized data are centered at 0 and have units equal to standard deviations of the original data.
- Example, a value of 2 form normalized data means that data point was two standard deviations larger than the mean.

# Empirical Covariance

- Consider now when we have pairs of data,  $(X_i, Y_i)$ .
- Their empirical covariance is

$$\text{Cov}(X, Y) = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y}) = \frac{1}{n-1} \left( \sum_{i=1}^n X_i Y_i - n\bar{X}\bar{Y} \right)$$

- Some people prefer to divide by  $n$  rather than  $n - 1$  (the latter produces an unbiased estimate.)
- The correlation is defined is

$$\text{Cor}(X, Y) = \frac{\text{Cov}(X, Y)}{S_x S_y}$$

where  $S_x$  and  $S_y$  are the estimates of standard deviations for the  $X$  observations and  $Y$  observations, respectively.

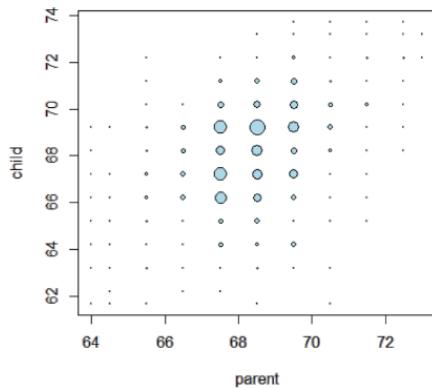
# Facts about Correlation

- $\text{Cor}(X, Y) = \text{Cor}(Y, X)$
- $-1 \leq \text{Cor}(X, Y) \leq 1$
- $\text{Cor}(X, Y) = 1$  and  $\text{Cor}(X, Y) = -1$  only when the  $X$  or  $Y$  observations fall perfectly on a positive or negative sloped line, respectively.
- $\text{Cor}(X, Y)$  measures the strength of the linear relationship between the  $X$  and  $Y$  data, with stronger relationships as  $\text{Cor}(X, Y)$  heads towards  $-1$  or  $1$ .
- $\text{Cor}(X, Y) = 0$  implies no linear relationship.

# Least Sq. Estimation of Regression Lines

## General Least Sq. for Linear Equations

Consider again the parent and child height data from Galton



## Fitting the Best Line

- Let  $Y_i$  be the  $i^{\text{th}}$  child's height and  $X_i$  be the  $i^{\text{th}}$  (average over the pair of) parents' heights.
- Consider finding the best line
  - Child's Height =  $\beta_0 + \text{Parent's Height } \beta_1$
- Use least squares

$$\sum_{i=1}^n \{Y_i - (\beta_0 + \beta_1 X_i)\}^2$$

- How do we do it?

## General Solution

- Let  $\mu_i = \beta_0 + \beta_1 X_i$  and our estimates be  $\hat{\mu}_i = \tilde{\beta}_0 + \tilde{\beta}_1 X_i$ .
- We want to minimize
 
$$\sum_{i=1}^n (Y_i - \mu_i)^2 = \sum_{i=1}^n (Y_i - \hat{\mu}_i)^2 + 2 \sum_{i=1}^n (Y_i - \hat{\mu}_i)(\hat{\mu}_i - \mu_i) + \sum_{i=1}^n (\hat{\mu}_i - \mu_i)^2$$
- Suppose that
 
$$\sum_{i=1}^n (Y_i - \hat{\mu}_i)(\hat{\mu}_i - \mu_i) = 0$$

then

$$\dagger = \sum_{i=1}^n (Y_i - \hat{\mu}_i)^2 + \sum_{i=1}^n (\hat{\mu}_i - \mu_i)^2 \geq \sum_{i=1}^n (Y_i - \hat{\mu}_i)^2$$

# Mean Only Regression

- So we know that if:

$$\sum_{i=1}^n (Y_i - \hat{\mu}_i)(\hat{\mu}_i - \mu_i) = 0$$

where  $\mu_i = \beta_0 + \beta_1 X_i$  and  $\hat{\mu}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$  then the line

$$Y = \hat{\beta}_0 + \hat{\beta}_1 X$$

is the least squares line.

- Consider forcing  $\beta_1 = 0$  and thus  $\hat{\beta}_1 = 0$ ; that is, only considering horizontal lines
- The solution works out to be

$$\hat{\beta}_0 = \bar{Y}.$$

$$\begin{aligned} \sum_{i=1}^n (Y_i - \hat{\mu}_i)(\hat{\mu}_i - \mu_i) &= \sum_{i=1}^n (Y_i - \hat{\beta}_0)(\hat{\beta}_0 - \beta_0) \\ &= (\hat{\beta}_0 - \beta_0) \sum_{i=1}^n (Y_i - \hat{\beta}_0) \end{aligned}$$

Thus, this will equal 0 if  $\sum_{i=1}^n (Y_i - \hat{\beta}_0) = n\bar{Y} - n\hat{\beta}_0 = 0$

Thus  $\hat{\beta}_0 = \bar{Y}$ .

# Regression through the Origin

- Recall that if:

$$\sum_{i=1}^n (Y_i - \hat{\mu}_i)(\hat{\mu}_i - \mu_i) = 0$$

where  $\mu_i = \beta_0 + \beta_1 X_i$  and  $\hat{\mu}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$  then the line

$$Y = \hat{\beta}_0 + \hat{\beta}_1 X$$

is the least squares line.

- Consider forcing  $\beta_0 = 0$  and thus  $\hat{\beta}_0 = 0$ ; that is, only considering lines through the origin
- The solution works out to be

Thus, this will equal 0 if  $\sum_{i=1}^n (Y_i X_i - \hat{\beta}_1 X_i^2) = \sum_{i=1}^n Y_i X_i - \hat{\beta}_1 \sum_{i=1}^n X_i^2 = 0$

Thus

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n Y_i X_i}{\sum_{i=1}^n X_i^2}.$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n Y_i X_i}{\sum_{i=1}^n X_i^2}.$$

# Recapping What We Know

- If we define  $\mu_i = \beta_0$  then  $\hat{\beta}_0 = \bar{Y}$ .
  - If we only look at horizontal lines, the least squares estimate of the intercept of that line is the average of the outcomes.
- If we define  $\mu_i = X_i \beta_1$  then  $\hat{\beta}_1 = \frac{\sum_{i=1}^n Y_i X_i}{\sum_{i=1}^n X_i^2}$ 
  - If we only look at lines through the origin, we get the estimated slope is the cross product of the X and Ys divided by the cross product of the Xs with themselves.
- What about when  $\mu_i = \beta_0 + \beta_1 X_i$ ? That is, we don't want to restrict ourselves to horizontal lines or lines through the origin.

$$\begin{aligned} \sum_{i=1}^n (Y_i - \hat{\mu}_i)(\hat{\mu}_i - \mu_i) &= \sum_{i=1}^n (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i)(\hat{\beta}_0 + \hat{\beta}_1 X_i - \beta_0 - \beta_1 X_i) \\ &= (\hat{\beta}_0 - \beta_0) \sum_{i=1}^n (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i) + (\beta_1 - \hat{\beta}_1) \sum_{i=1}^n (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i)X_i \quad \text{And thus} \\ &= \sum_{i=1}^n ((Y_i - \bar{Y}) - \hat{\beta}_1(X_i - \bar{X}))X_i \\ &\quad \sum_{i=1}^n (Y_i - \bar{Y})X_i - \hat{\beta}_1 \sum_{i=1}^n (X_i - \bar{X})X_i = 0. \end{aligned}$$

Note that

$$0 = \sum_{i=1}^n (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i) = n\bar{Y} - n\hat{\beta}_0 - n\hat{\beta}_1 \bar{X} \text{ implies that } \hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$$

Then

$$\sum_{i=1}^n (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i)X_i = \sum_{i=1}^n (Y_i - \bar{Y} + \hat{\beta}_1 \bar{X} - \hat{\beta}_1 X_i)X_i$$

So we arrive at

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n ((Y_i - \bar{Y})X_i)}{\sum_{i=1}^n (X_i - \bar{X})X_i} = \frac{\sum_{i=1}^n (Y_i - \bar{Y})(X_i - \bar{X})}{\sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})} = \text{Cor}(Y, X) \frac{\text{Sd}(Y)}{\text{Sd}(X)}.$$

And recall

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}.$$

# Consequences

- The least squares model fit to the line  $Y = \beta_0 + \beta_1 X$  through the data pairs  $(X_i, Y_i)$  with  $Y_i$  as the outcome obtains the line  $\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X$  where

$$\hat{\beta}_1 = \text{Cor}(Y, X) \frac{\text{Sd}(Y)}{\text{Sd}(X)} \quad \hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$$

- $\hat{\beta}_1$  has the units of  $Y/X$ ,  $\hat{\beta}_0$  has the units of  $Y$ .
- The line passes through the point  $(\bar{X}, \bar{Y})$
- The slope of the regression line with  $X$  as the outcome and  $Y$  as the predictor is  $\text{Cor}(Y, X)\text{Sd}(X)/\text{Sd}(Y)$ .
- The slope is the same one you would get if you centered the data,  $(X_i - \bar{X}, Y_i - \bar{Y})$ , and did regression through the origin.
- If you normalized the data,  $\left\{ \frac{X_i - \bar{X}}{\text{Sd}(X)}, \frac{Y_i - \bar{Y}}{\text{Sd}(Y)} \right\}$ , the slope is  $\text{Cor}(Y, X)$ .

## Revisiting Galton's Data

Double check our calculations using R Reversing the outcome/predictor relationship

```
y <- galton$child
x <- galton$parent
beta1 <- cor(y, x) * sd(y) / sd(x)
beta0 <- mean(y) - beta1 * mean(x)
rbind(c(beta0, beta1), coef(lm(y ~ x)))
```

```
beta1 <- cor(y, x) * sd(x) / sd(y)
beta0 <- mean(x) - beta1 * mean(y)
rbind(c(beta0, beta1), coef(lm(x ~ y)))
```

	(Intercept)	x
[1,]	23.94	0.6463
[2,]	23.94	0.6463

	(Intercept)	y
[1,]	46.14	0.3256
[2,]	46.14	0.3256

Regression through the origin yields an equivalent slope if you center the data first

```
yc <- y - mean(y)
xc <- x - mean(x)
beta1 <- sum(yc * xc) / sum(xc ^ 2)
c(beta1, coef(lm(y ~ x))[2])
```

x
0.6463 0.6463

Normalizing variables results in the slope being the correlation

```
yn <- (y - mean(y))/sd(y)
xn <- (x - mean(x))/sd(x)
c(cor(y, x), cor(yn, xn), coef(lm(yn ~ xn))[2])
```

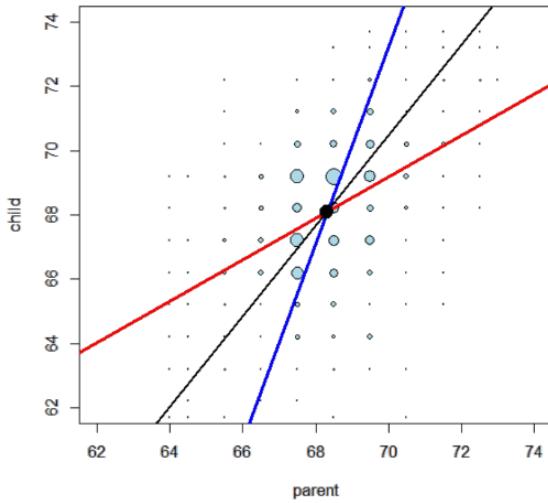
xn
0.4588 0.4588 0.4588

# Plotting the Fit

- Size of points are frequencies at that X, Y combination.
- For the red lie the child is outcome.
- For the blue, the parent is the outcome (accounting for the fact that the response is plotted on the horizontal axis).
- Black line assumes  $\text{Cor}(Y, X) = 1$  (slope is  $Sd(Y)/Sd(x)$ ).
- Big black dot is  $(\bar{X}, \bar{Y})$ .

The code to add the lines

```
abline(mean(y) - mean(x) * cor(y, x) * sd(y) / sd(x),
       sd(y) / sd(x) * cor(y, x),
       lwd = 3, col = "red")
abline(mean(y) - mean(x) * sd(y) / sd(x) / cor(y, x),
       sd(y) * cor(y, x) / sd(x),
       lwd = 3, col = "blue")
abline(mean(y) - mean(x) * sd(y) / sd(x),
       sd(y) / sd(x),
       lwd = 2)
points(mean(x), mean(y), cex = 2, pch = 19)
```



# Statistical Linear Regression Models

## Basic Reg. Model with Add. Gaussian Error

- Least squares is an estimation tool, how do we do inference?
- Consider developing a probabilistic model for linear regression

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

- Here the  $\epsilon_i$  are assumed iid  $N(0, \sigma^2)$ .
- Note,  $E[Y_i | X_i = x_i] = \mu_i = \beta_0 + \beta_1 x_i$
- Note,  $\text{Var}(Y_i | X_i = x_i) = \sigma^2$ .
- Likelihood equivalent model specification is that the  $Y_i$  are independent  $N(\mu_i, \sigma^2)$ .

## Likelihood

$$\mathcal{L}(\beta, \sigma) = \prod_{i=1}^n \left\{ (2\pi\sigma^2)^{-1/2} \exp\left(-\frac{1}{2\sigma^2} (y_i - \mu_i)^2\right) \right\}$$

so that the twice the negative log (base e) likelihood is

$$-2 \log\{\mathcal{L}(\beta, \sigma)\} = \frac{1}{\sigma^2} \sum_{i=1}^n (y_i - \mu_i)^2 + n \log(\sigma^2)$$

Discussion

- Maximizing the likelihood is the same as minimizing -2 log likelihood
- The least squares estimate for  $\mu_i = \beta_0 + \beta_1 x_i$  is exactly the maximum likelihood estimate (regardless of  $\sigma$ )

## Recap

- Model  $Y_i = \mu_i + \epsilon_i = \beta_0 + \beta_1 X_i + \epsilon_i$  where  $\epsilon_i$  are iid  $N(0, \sigma^2)$
- ML estimates of  $\beta_0$  and  $\beta_1$  are the least squares estimates

$$\hat{\beta}_1 = \text{Cor}(Y, X) \frac{\text{Sd}(Y)}{\text{Sd}(X)} \quad \hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$$

- $E[Y | X = x] = \beta_0 + \beta_1 x$
- $\text{Var}(Y | X = x) = \sigma^2$

# Interpreting Regression Coefficients, the itc

- $\beta_0$  is the expected value of the response when the predictor is 0

$$E[Y|X=0] = \beta_0 + \beta_1 \times 0 = \beta_0$$

- Note, this isn't always of interest, for example when  $X = 0$  is impossible or far outside of the range of data. ( $X$  is blood pressure, or height etc.)

- Consider that

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i = \beta_0 + a\beta_1 + \beta_1(X_i - a) + \epsilon_i = \tilde{\beta}_0 + \beta_1(X_i - a) + \epsilon_i$$

So, shifting you  $X$  values by value  $a$  changes the intercept, but not the slope.

- Often  $a$  is set to  $\bar{X}$  so that the intercept is interpreted as the expected response at the average  $X$  value.

# Interpreting Regression Coefficients, Slope

- $\beta_1$  is the expected change in response for a 1 unit change in the predictor

$$E[Y | X = x+1] - E[Y | X = x] = \beta_0 + \beta_1(x+1) - (\beta_0 + \beta_1x) = \beta_1$$

- Consider the impact of changing the units of  $X$ .

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i = \beta_0 + \frac{\beta_1}{a} (X_i a) + \epsilon_i = \beta_0 + \tilde{\beta}_1 (X_i a) + \epsilon_i$$

- Therefore, multiplication of  $X$  by a factor  $a$  results in dividing the coefficient by a factor of  $a$ .

- Example:  $X$  is height in m and  $Y$  is weight in kg. Then  $\beta_1$  is kg/m. Converting  $X$  to cm implies multiplying  $X$  by 100cm/m. To get  $\beta_1$  in the right units, we have to divide by 100cm/m to get it to have the right units.

$$Xm \times \frac{100cm}{m} = (100X)cm \text{ and } \beta_1 \frac{kg}{m} \times \frac{1m}{100cm} = \left( \frac{\beta_1}{100} \right) \frac{kg}{cm}$$

# Using Reg. Coefficients for Prediction

- If we would like to guess the outcome at a particular value of the predictor, say  $X$ , the regression model guesses

$$\hat{\beta}_0 + \hat{\beta}_1 X$$

- Note that at the observed value of  $X$ s, we obtain the predictions

$$\hat{\mu}_i = \hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$$

- Remember that least squares minimizes

$$\sum_{i=1}^n (Y_i - \mu_i)$$

for  $\mu_i$  expressed as points on a line

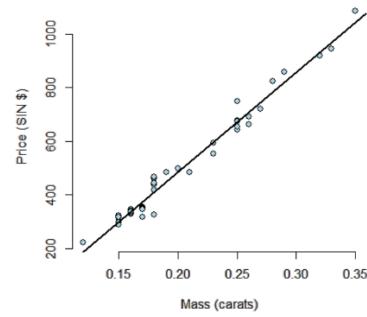
# Example

diamond data set from UsingR

Data is diamond prices (Singapore dollars) and diamond weight in carats (standard measure of diamond mass, 0.2 g). To get the data use `library(UsingR); data(diamond)`

Plotting the fitted regression line and data

```
data(diamond)
plot(diamond$carat, diamond$price,
      xlab = "Mass (carats)",
      ylab = "Price (SIN $)",
      bg = "lightblue",
      col = "black", cex = 1.1, pch = 21, frame = FALSE)
abline(lm(price ~ carat, data = diamond), lwd = 2)
```



## Fitting the Linear Regression Model

```
fit <- lm(price ~ carat, data = diamond)
coef(fit)
```

(Intercept)	carat
-259.6	3721.0

- We estimate an expected 3721.02 (SIN) dollar increase in price for every carat increase in mass of diamond.
- The intercept -259.63 is the expected price of a 0 carat diamond.

## Getting a More Interpretable Intercept

```
fit2 <- lm(price ~ I(carat - mean(carat)), data = diamond)
coef(fit2)
```

(Intercept)	I(carat - mean(carat))
500.1	3721.0

Thus \$500.1 is the expected price for the average sized diamond of the data (0.2042 carats).

## Changing Scale

- A one carat increase in a diamond is pretty big, what about changing units to 1/10th of a carat?
- We can just do this by just dividing the coefficient by 10.
  - We expect a 372.102 (SIN) dollar change in price for every 1/10th of a carat increase in mass of diamond.
- Showing that it's the same if we rescale the Xs and refit

```
fit3 <- lm(price ~ I(carat * 10), data = diamond)
coef(fit3)
```

(Intercept)	I(carat * 10)
-259.6	372.1

# Predicting the Price of a Diamond

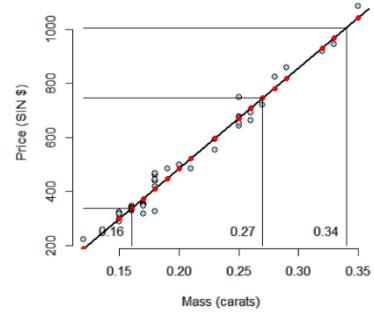
Predicted values at the observed Xs (red) and at the new Xs (lines)

```
newx <- c(0.16, 0.27, 0.34)  
coef(fit)[1] + coef(fit)[2] * newx
```

```
[1] 335.7 745.1 1005.5
```

```
predict(fit, newdata = data.frame(carat = newx))
```

```
1      2      3  
335.7 745.1 1005.5
```



# Residuals and Residual Variation

- Model  $Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$  where  $\epsilon_i \sim N(0, \sigma^2)$ .
- Observed outcome  $i$  is  $Y_i$  at predictor value  $X_i$
- Predicted outcome  $i$  is  $\hat{Y}_i$  at predictor value  $X_i$  is

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$$

- Residual, the between the observed and predicted outcome

$$e_i = Y_i - \hat{Y}_i$$

- The vertical distance between the observed data point and the regression line

- Least squares minimizes  $\sum_{i=1}^n e_i^2$
- The  $e_i$  can be thought of as estimates of the  $\epsilon_i$ .

## Properties of the Residuals

- $E[e_i] = 0$ .
- If an intercept is included,  $\sum_{i=1}^n e_i = 0$
- If a regressor variable,  $X_i$ , is included in the model  $\sum_{i=1}^n e_i X_i = 0$ .
- Residuals are useful for investigating poor model fit.
- Positive residuals are above the line, negative residuals are below.
- Residuals can be thought of as the outcome ( $Y$ ) with the linear association of the predictor ( $X$ ) removed.
- One differentiates residual variation (variation after removing the predictor) from systematic variation (variation explained by the regression model).
- Residual plots highlight poor model fit.

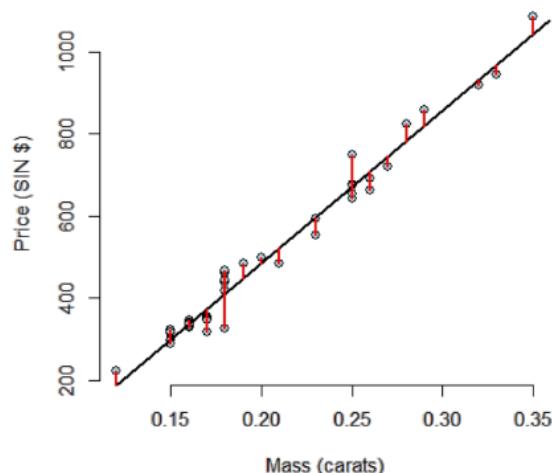
## Code

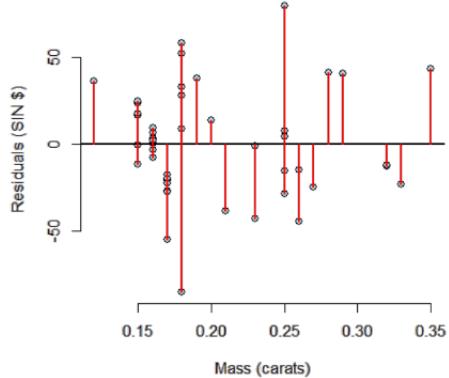
```
data(diamond)
y <- diamond$price; x <- diamond$carat; n <- length(y)
fit <- lm(y ~ x)
e <- resid(fit)
yhat <- predict(fit)
max(abs(e - (y - yhat)))
```

```
[1] 9.486e-13
```

```
max(abs(e - (y - coef(fit)[1] - coef(fit)[2] * x)))
```

```
[1] 9.486e-13
```

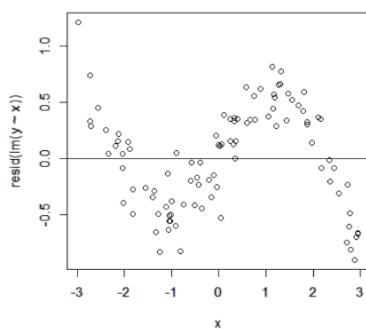
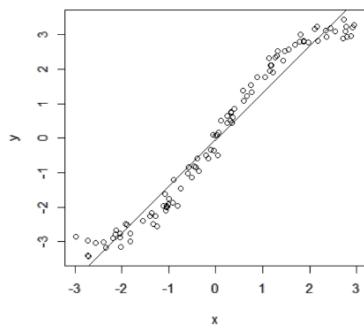




## Non-Linear Data

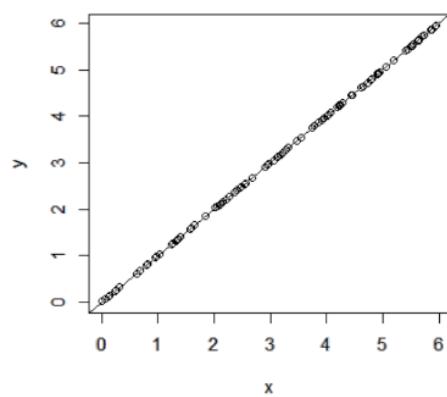
```
x <- runif(100, -3, 3); y <- x + sin(x) + rnorm(100, sd = .2);
plot(x, y); abline(lm(y ~ x))
```

```
plot(x, resid(lm(y ~ x)));
abline(h = 0)
```



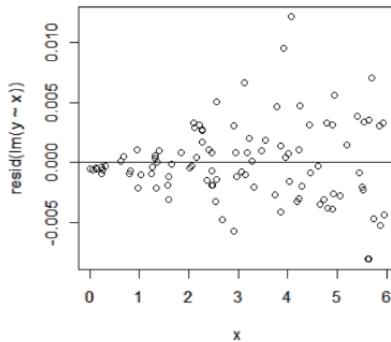
## Heteroscedasticity

```
x <- runif(100, 0, 6); y <- x + rnorm(100, mean = 0, sd = .001 * x);
plot(x, y); abline(lm(y ~ x))
```



# Getting Rid of Blank Space can Help

```
plot(x, resid(lm(y ~ x)));
abline(h = 0)
```



## Estimating Residual Variation

- Model  $Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$  where  $\epsilon_i \sim N(0, \sigma^2)$ .
- The ML estimate of  $\sigma^2$  is  $\frac{1}{n} \sum_{i=1}^n e_i^2$ , the average squared residual.
- Most people use

$$\hat{\sigma}^2 = \frac{1}{n-2} \sum_{i=1}^n e_i^2.$$

- The  $n-2$  instead of  $n$  is so that  $E[\hat{\sigma}^2] = \sigma^2$

## Diamond Example

```
y <- diamonds$price; x <- diamonds$carat; n <- length(y)
fit <- lm(y ~ x)
summary(fit)$sigma
```

```
[1] 31.84
```

```
sqrt(sum(resid(fit)^2) / (n - 2))
```

```
[1] 31.84
```

# Summarizing Variation

$$\begin{aligned}\sum_{i=1}^n (Y_i - \bar{Y})^2 &= \sum_{i=1}^n (Y_i - \hat{Y}_i + \hat{Y}_i - \bar{Y})^2 \\ &= \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 + 2 \sum_{i=1}^n (Y_i - \hat{Y}_i)(\hat{Y}_i - \bar{Y}) + \sum_{i=1}^n (\hat{Y}_i - \bar{Y})^2\end{aligned}$$


---

$$\sum_{i=1}^n (Y_i - \bar{Y})^2 = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 + \sum_{i=1}^n (\hat{Y}_i - \bar{Y})^2$$

Scratch work

$$(Y_i - \hat{Y}_i) = \{Y_i - (\bar{Y} - \hat{\beta}_1 \bar{X}) - \hat{\beta}_1 X_i\} = (Y_i - \bar{Y}) - \hat{\beta}_1(X_i - \bar{X})$$

$$(\hat{Y}_i - \bar{Y}) = (\bar{Y} - \hat{\beta}_1 \bar{X} - \hat{\beta}_1 X_i - \bar{Y}) = \hat{\beta}_1(X_i - \bar{X})$$

$$\sum_{i=1}^n (Y_i - \hat{Y}_i)(\hat{Y}_i - \bar{Y}) = \sum_{i=1}^n \{(Y_i - \bar{Y}) - \hat{\beta}_1(X_i - \bar{X})\} \{(\hat{\beta}_1(X_i - \bar{X})\}$$

$$= \hat{\beta}_1 \sum_{i=1}^n (Y_i - \bar{Y})(X_i - \bar{X}) - \hat{\beta}_1^2 \sum_{i=1}^n (X_i - \bar{X})^2$$

$$= \hat{\beta}_1^2 \sum_{i=1}^n (X_i - \bar{X})^2 - \hat{\beta}_1^2 \sum_{i=1}^n (X_i - \bar{X})^2 = 0$$

Or

Total Variation = Residual Variation + Regression Variation

Define the percent of total variation described by the model as

$$R^2 = \frac{\sum_{i=1}^n (\hat{Y}_i - \bar{Y})^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2} = 1 - \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2}$$

## Relation Between R^2 and r

Recall that  $(\hat{Y}_i - \bar{Y}) = \hat{\beta}_1(X_i - \bar{X})$  so that

$$R^2 = \frac{\sum_{i=1}^n (\hat{Y}_i - \bar{Y})^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2} = \hat{\beta}_1^2 \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2} = \text{Cor}(Y, X)^2$$

Since, recall,

$$\hat{\beta}_1 = \text{Cor}(Y, X) \frac{\text{Sd}(Y)}{\text{Sd}(X)}$$

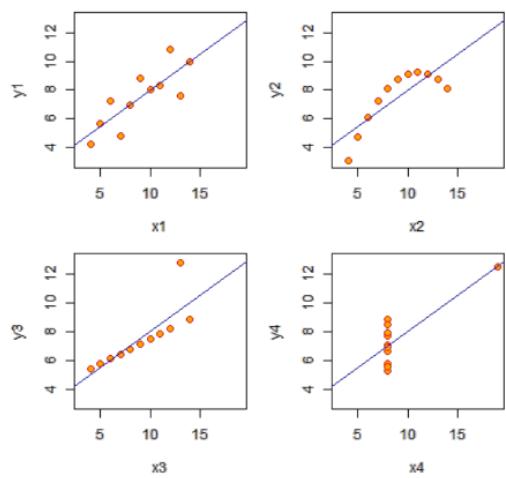
So,  $R^2$  is literally r squared.

## Facts about R^2

- $R^2$  is the percentage of variation explained by the regression model.
- $0 \leq R^2 \leq 1$
- $R^2$  is the sample correlation squared.
- $R^2$  can be a misleading summary of model fit.
  - Deleting data can inflate  $R^2$ .
  - (For later.) Adding terms to a regression model always increases  $R^2$ .
- Do `example(anscombe)` to see the following data.
  - Basically same mean and variance of X and Y.
  - Identical correlations (hence same  $R^2$ ).
  - Same linear regression relationship.

# Data(anscombe); example(anscombe)

Anscombe's 4 Regression data sets



# Inference in Regression

## Our Model and Fitted Values

- Consider the model

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

- $\epsilon \sim N(0, \sigma^2)$ .
- We assume that the true model is known.
- We assume that you've seen confidence intervals and hypothesis tests before.
- $\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$
- $\hat{\beta}_1 = \text{Cor}(Y, X) \frac{\text{Sd}(Y)}{\text{Sd}(X)}$ .

## Review

- Statistics like  $\frac{\hat{\theta} - \theta}{\hat{\sigma}_{\hat{\theta}}}$  often have the following properties.
  - Is normally distributed and has a finite sample Student's T distribution if the estimated variance is replaced with a sample estimate (under normality assumptions).
  - Can be used to test  $H_0 : \theta = \theta_0$  versus  $H_a : \theta >, <, \neq \theta_0$ .
  - Can be used to create a confidence interval for  $\theta$  via  $\hat{\theta} \pm Q_{1-\alpha/2} \hat{\sigma}_{\hat{\theta}}$  where  $Q_{1-\alpha/2}$  is the relevant quantile from either a normal or T distribution.
- In the case of regression with iid sampling assumptions and normal errors, our inferences will follow very similarly to what you saw in your inference class.
- We won't cover asymptotics for regression analysis, but suffice it to say that under assumptions on the ways in which the  $X$  values are collected, the iid sampling model, and mean model, the normal results hold to create intervals and confidence intervals

## Standard Error (Conditioned on X)

$$\begin{aligned} \text{Var}(\hat{\beta}_1) &= \text{Var}\left(\frac{\sum_{i=1}^n (Y_i - \bar{Y})(X_i - \bar{X})}{\sum_{i=1}^n (X_i - \bar{X})^2}\right) \\ &= \frac{\text{Var}(\sum_{i=1}^n Y_i(X_i - \bar{X}))}{\left(\sum_{i=1}^n (X_i - \bar{X})^2\right)^2} \\ &= \frac{\sum_{i=1}^n \sigma^2 (X_i - \bar{X})^2}{\left(\sum_{i=1}^n (X_i - \bar{X})^2\right)^2} \\ &= \frac{\sigma^2}{\sum_{i=1}^n (X_i - \bar{X})^2} \end{aligned}$$

# Results

- $\sigma_{\hat{\beta}_1}^2 = \text{Var}(\hat{\beta}_1) = \sigma^2 / \sum_{i=1}^n (X_i - \bar{X})^2$
- $\sigma_{\hat{\beta}_0}^2 = \text{Var}(\hat{\beta}_0) = \left( \frac{1}{n} + \frac{\bar{X}^2}{\sum_{i=1}^n (X_i - \bar{X})^2} \right) \sigma^2$
- In practice,  $\sigma$  is replaced by its estimate.
- It's probably not surprising that under iid Gaussian errors

$$\frac{\hat{\beta}_j - \beta_j}{\hat{\sigma}_{\hat{\beta}_j}}$$

follows a t distribution with  $n - 2$  degrees of freedom and a normal distribution for large  $n$ .

- This can be used to create confidence intervals and perform hypothesis tests.

## Example Diamond Data Set

```
library(UsingR); data(diamond)
y <- diamond$price; x <- diamond$carat; n <- length(y)
beta1 <- cor(y, x) * sd(y) / sd(x)
beta0 <- mean(y) - beta1 * mean(x)
e <- y - beta0 - beta1 * x
sigma <- sqrt(sum(e^2) / (n-2))
ssx <- sum((x - mean(x))^2)
seBeta0 <- (1 / n + mean(x)^2 / ssx) ^ .5 * sigma
seBeta1 <- sigma / sqrt(ssx)
tBeta0 <- beta0 / seBeta0; tBeta1 <- beta1 / seBeta1
pBeta0 <- 2 * pt(abs(tBeta0), df = n - 2, lower.tail = FALSE)
pBeta1 <- 2 * pt(abs(tBeta1), df = n - 2, lower.tail = FALSE)
coefTable <- rbind(c(beta0, seBeta0, tBeta0, pBeta0), c(beta1, seBeta1, tBeta1, pBeta1))
colnames(coefTable) <- c("Estimate", "Std. Error", "t value", "P(>|t|)")
rownames(coefTable) <- c("(Intercept)", "x")
```

coefTable

	Estimate	Std. Error	t value	P(> t )
(Intercept)	-259.6	17.32	-14.99	2.523e-19
x	3721.0	81.79	45.50	6.751e-40

```
fit <- lm(y ~ x);
summary(fit)$coefficients
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-259.6	17.32	-14.99	2.523e-19
x	3721.0	81.79	45.50	6.751e-40

## Getting a Confidence Interval

```
sumCoef <- summary(fit)$coefficients
sumCoef[1,1] + c(-1, 1) * qt(.975, df = fit$df) * sumCoef[1, 2]
```

[1] -294.5 -224.8

```
sumCoef[2,1] + c(-1, 1) * qt(.975, df = fit$df) * sumCoef[2, 2]
```

[1] 3556 3886

With 95% confidence, we estimate that a 0.1 carat increase in diamond size results in a 355.6 to 388.6 increase in price in (Singapore) dollars.

# Prediction of Outcomes

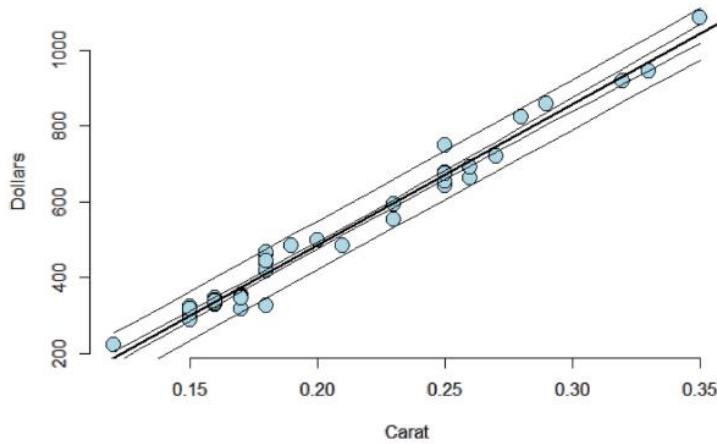
- Consider predicting Y at a value of X
  - Predicting the price of a diamond given the carat
  - Predicting the height of a child given the height of the parents
- The obvious estimate for prediction at point  $x_0$  is

$$\hat{\beta}_0 + \hat{\beta}_1 x_0$$

- A standard error is needed to create a prediction interval.
- There's a distinction between intervals for the regression line at point  $x_0$  and the prediction of what a y would be at point  $x_0$ .
- Line at  $x_0$  se,  $\hat{\sigma} \sqrt{\frac{1}{n} + \frac{(x_0 - \bar{X})^2}{\sum_{i=1}^n (X_i - \bar{X})^2}}$
- Prediction interval se at  $x_0$ ,  $\hat{\sigma} \sqrt{1 + \frac{1}{n} + \frac{(x_0 - \bar{X})^2}{\sum_{i=1}^n (X_i - \bar{X})^2}}$

## Plotting the Prediction Intervals

```
plot(x, y, frame=FALSE, xlab="Carat", ylab="Dollars", pch=21, col="black", bg="lightblue", cex=2)
abline(fit, lwd = 2)
xVals <- seq(min(x), max(x), by = .01)
yVals <- beta0 + beta1 * xVals
se1 <- sigma * sqrt(1 / n + (xVals - mean(x))^2/ssx)
se2 <- sigma * sqrt(1 + 1 / n + (xVals - mean(x))^2/ssx)
lines(xVals, yVals + 2 * se1)
lines(xVals, yVals - 2 * se1)
lines(xVals, yVals + 2 * se2)
lines(xVals, yVals - 2 * se2)
```

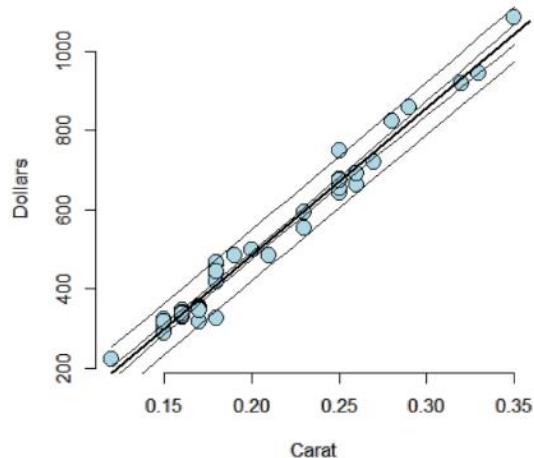


# Discussion

- Both intervals have varying widths.
  - Least width at the mean of the Xs.
- We are quite confident in the regression line, so that interval is very narrow.
  - If we knew  $\beta_0$  and  $\beta_1$  this interval would have zero width.
- The prediction interval must incorporate the variability in the data around the line.
  - Even if we knew  $\beta_0$  and  $\beta_1$  this interval would still have width.

## In R

```
newdata <- data.frame(x = xVals)
p1 <- predict(fit, newdata, interval = ("confidence"))
p2 <- predict(fit, newdata, interval = ("prediction"))
plot(x, y, frame=FALSE, xlab="Carat", ylab="Dollars", pch=21, col="black", bg="lightblue", cex=2)
abline(fit, lwd = 2)
lines(xVals, p1[,2]); lines(xVals, p1[,3])
lines(xVals, p2[,2]); lines(xVals, p2[,3])
```



# Multivariable Regression Analysis

- If I were to present evidence of a relationship between breath mint usage (mints per day, X) and pulmonary function (measured in FEV), you would be skeptical.
  - Likely, you would say, 'smokers tend to use more breath mints than non smokers, smoking is related to a loss in pulmonary function. That's probably the culprit.'
  - If asked what would convince you, you would likely say, 'If non-smoking breath mint users had lower lung function than non-smoking non-breath mint users and, similarly, if smoking breath mint users had lower lung function than smoking non-breath mint users, I'd be more inclined to believe you'.
- In other words, to even consider my results, I would have to demonstrate that they hold while holding smoking status fixed.
- An insurance company is interested in how last year's claims can predict a person's time in the hospital this year.
  - They want to use an enormous amount of data contained in claims to predict a single number. Simple linear regression is not equipped to handle more than one predictor.
- How can one generalize SLR to incorporate lots of regressors for the purpose of prediction?
- What are the consequences of adding lots of regressors?
  - Surely there must be consequences to throwing variables in that aren't related to Y?
  - Surely there must be consequences to omitting variables that are?

## Linear Model

- The general linear model extends simple linear regression (SLR) by adding terms linearly into the model.

$$Y_i = \beta_1 X_{1i} + \beta_2 X_{2i} + \dots + \beta_p X_{pi} + \epsilon_i = \sum_{k=1}^p X_{ki} \beta_j + \epsilon_i$$

- Here  $X_{1i} = 1$  typically, so that an intercept is included.
- Least squares (and hence ML estimates under iid Gaussianity of the errors) minimizes

$$\sum_{i=1}^n \left( Y_i - \sum_{k=1}^p X_{ki} \beta_j \right)^2$$

- Note, the important linearity is linearity in the coefficients. Thus

$$Y_i = \beta_1 X_{1i}^2 + \beta_2 X_{2i}^2 + \dots + \beta_p X_{pi}^2 + \epsilon_i$$

is still a linear model. (We've just squared the elements of the predictor variables.)

## How to Get Estimates

- The real way requires linear algebra. We'll go over an intuitive development instead.
- Recall that the LS estimate for regression through the origin,  $E[Y_i] = X_{1i}\beta_1$ , was  $\sum X_i Y_i / \sum X_i^2$ .
- Let's consider two regressors,  $E[Y_i] = X_{1i}\beta_1 + X_{2i}\beta_2 = \mu_i$ .
- Also, recall, that if  $\hat{\mu}_i$  satisfies

$$\sum_{i=1}^n (Y_i - \hat{\mu}_i)(\hat{\mu}_i - \mu_i) = 0$$

for all possible values of  $\mu_i$ , then we've found the LS estimates.

$$\sum_{i=1}^n (Y_i - \hat{\mu}_i)(\hat{\mu}_i - \mu_i) = \sum_{i=1}^n (Y_i - \hat{\beta}_1 X_{1i} - \hat{\beta}_2 X_{2i}) \left\{ X_{1i}(\hat{\beta}_1 - \beta_1) + X_{2i}(\hat{\beta}_2 - \beta_2) \right\}$$

- Thus we need
  - $\sum_{i=1}^n (Y_i - \hat{\beta}_1 X_{1i} - \hat{\beta}_2 X_{2i}) X_{1i} = 0$
  - $\sum_{i=1}^n (Y_i - \hat{\beta}_1 X_{1i} - \hat{\beta}_2 X_{2i}) X_{2i} = 0$
- Hold  $\hat{\beta}_1$  fixed in 2. and solve and we get that
 
$$\hat{\beta}_2 = \frac{\sum_{i=1}^n (Y_i - X_{1i}\hat{\beta}_1) X_{2i}}{\sum_{i=1}^n X_{2i}^2}$$
- Plugging this into 1. we get that
 
$$0 = \sum_{i=1}^n \left\{ Y_i - \frac{\sum_j X_{2j} Y_j}{\sum_j X_{2j}^2} X_{2i} + \beta_1 \left( X_{1i} - \frac{\sum_j X_{2j} X_{1j}}{\sum_j X_{2j}^2} X_{2i} \right) \right\} X_{1i}$$
- Re writing this we get
 
$$0 = \sum_{i=1}^n \left\{ e_{i,Y|X_2} - \hat{\beta}_1 e_{i,X_1|X_2} \right\} X_{1i}$$

where  $e_{i,a|b} = a_i - \frac{\sum_{j=1}^n a_j b_j}{\sum_{j=1}^n b_j^2} b_i$  is the residual when regressing  $b$  from  $a$  without an intercept.
- We get the solution
 
$$\hat{\beta}_1 = \frac{\sum_{i=1}^n e_{i,Y|X_2} e_{i,X_1|X_2}}{\sum_{i=1}^n e_{i,X_1|X_2} X_{1i}}$$

But note that

$$\begin{aligned} \sum_{i=1}^n e_{i,X_1|X_2}^2 &= \sum_{i=1}^n e_{i,X_1|X_2} \left( X_{1i} - \frac{\sum_j X_{2j} X_{1j}}{\sum_j X_{2j}^2} X_{2i} \right) \\ &= \sum_{i=1}^n e_{i,X_1|X_2} X_{1i} - \frac{\sum_j X_{2j} X_{1j}}{\sum_j X_{2j}^2} \sum_{i=1}^n e_{i,X_1|X_2} X_{2i} \end{aligned}$$

But  $\sum_{i=1}^n e_{i,X_1|X_2} X_{2i} = 0$ . So we get that

$$\sum_{i=1}^n e_{i,X_1|X_2}^2 = \sum_{i=1}^n e_{i,X_1|X_2} X_{1i}$$

Thus we get that

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n e_{i,Y|X_2} e_{i,X_1|X_2}}{\sum_{i=1}^n e_{i,X_1|X_2}^2}$$

## Summing Up Fitting with Two Regressors

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n e_{i,Y|X_2} e_{i,X_1|X_2}}{\sum_{i=1}^n e_{i,X_1|X_2}^2}$$

- That is, the regression estimate for  $\beta_1$  is the regression through the origin estimate having regressed  $X_2$  out of both the response and the predictor.
- (Similarly, the regression estimate for  $\beta_2$  is the regression through the origin estimate having regressed  $X_1$  out of both the response and the predictor.)
- More generally, multivariate regression estimates are exactly those having removed the linear relationship of the other variables from both the regressor and response.

## Examples with 2 var, simple linear reg.

$Y_i = \beta_1 X_{1i} + \beta_2 X_{2i}$  where  $X_{2i} = 1$  is an intercept term.

Then  $\frac{\sum_j X_{2j} X_{1j}}{\sum_j X_{2j}^2} X_{2i} = \frac{\sum_j X_{1j}}{n} = \bar{X}_1$ .

$e_{i,X_1|X_2} = X_{1i} - \bar{X}_1$ .

Similarly  $e_{i,Y|X_2} = Y_i - \bar{Y}$ .

Thus

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n e_{i,Y|X_2} e_{i,X_1|X_2}}{\sum_{i=1}^n e_{i,X_1|X_2}^2} = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^n (X_i - \bar{X})^2} = Cor(X, Y) \frac{Sd(Y)}{Sd(X)}$$

# The General Case

- The equations

$$\sum_{i=1}^n (Y_i - X_{1i}\hat{\beta}_1 - \dots - X_{ip}\hat{\beta}_p)X_k = 0$$

for  $k = 1, \dots, p$  yields  $p$  equations with  $p$  unknowns.

- Solving them yields the least squares estimates. (With obtaining a good, fast, general solution requiring some knowledge of linear algebra.)
- The least squares estimate for the coefficient of a multivariate regression model is exactly regression through the origin with the linear relationships with the other regressors removed from both the regressor and outcome by taking residuals.
- In this sense, multivariate regression "adjusts" a coefficient for the linear impact of the other variables.

## Fitting LS Equations

Just so I don't leave you hanging, let's show a way to get estimates. Recall the equations:

$$\sum_{i=1}^n (Y_i - X_{1i}\hat{\beta}_1 - \dots - X_{ip}\hat{\beta}_p)X_k = 0$$

If I hold  $\hat{\beta}_1, \dots, \hat{\beta}_{p-1}$  fixed then we get that

$$\hat{\beta}_p = \frac{\sum_{i=1}^n (Y_i - X_{1i}\hat{\beta}_1 - \dots - X_{ip-1}\hat{\beta}_{p-1})X_{ip}}{\sum_{i=1}^n X_{ip}^2}$$

Plugging this back into the equations, we wind up with

$$\sum_{i=1}^n (e_{i,Y|X_p} - e_{i,X_1|X_p}\hat{\beta}_1 - \dots - e_{i,X_{p-1}|X_p}\hat{\beta}_{p-1})X_k = 0$$

Note that

$$X_k = e_{i,X_k|X_p} + \frac{\sum_{i=1}^n X_{ik}X_{ip}}{\sum_{i=1}^n X_{ip}^2} X_p$$

and  $\sum_{i=1}^n e_{i,X_j|X_p}X_{ip} = 0$ . Thus

$$\sum_{i=1}^n (e_{i,Y|X_p} - e_{i,X_1|X_p}\hat{\beta}_1 - \dots - e_{i,X_{p-1}|X_p}\hat{\beta}_{p-1})X_k = 0$$

is equal to

$$\sum_{i=1}^n (e_{i,Y|X_p} - e_{i,X_1|X_p}\hat{\beta}_1 - \dots - e_{i,X_{p-1}|X_p}\hat{\beta}_{p-1})e_{i,X_k|X_p} = 0$$

## Summary

- We've reduced  $p$  LS equations and  $p$  unknowns to  $p - 1$  LS equations and  $p - 1$  unknowns.
  - Every variable has been replaced by its residual with  $X_p$ .
  - This process can then be iterated until only Y and one variable remains.
- Think of it as follows. If we want an adjusted relationship between y and x, keep taking residuals over confounders and do regression through the origin.
  - The order that you do the confounders doesn't matter.
  - (It can't because our choice of doing  $p$  first was arbitrary.)
- This isn't a terribly efficient way to get estimates. But, it's nice conceptually, as it shows how regression estimates are adjusted for the linear relationship with other variables.

# Demonstration that it Works – Example

Linear model with two variables and an intercept

```
n <- 100; x <- rnorm(n); x2 <- rnorm(n); x3 <- rnorm(n)
y <- x + x2 + x3 + rnorm(n, sd = .1)
e <- function(a, b) a - sum( a * b ) / sum( b ^ 2) * b
ey <- e(e(y, x2), e(x3, x2))
ex <- e(e(x, x2), e(x3, x2))
sum(ey * ex) / sum(ex ^ 2)
```

```
[1] 1.004
```

```
coef(lm(y ~ x + x2 + x3 - 1)) #the -1 removes the intercept term
```

```
      x      x2      x3
1.0040 0.9899 1.0078
```

## Showing Order Doesn't Matter

```
ey <- e(e(y, x3), e(x2, x3))
ex <- e(e(x, x3), e(x2, x3))
sum(ey * ex) / sum(ex ^ 2)
```

```
[1] 1.004
```

```
coef(lm(y ~ x + x2 + x3 - 1)) #the -1 removes the intercept term
```

```
      x      x2      x3
1.0040 0.9899 1.0078
```

## Residuals Again

```
ey <- resid(lm(y ~ x2 + x3 - 1))
ex <- resid(lm(x ~ x2 + x3 - 1))
sum(ey * ex) / sum(ex ^ 2)
```

```
[1] 1.004
```

```
coef(lm(y ~ x + x2 + x3 - 1)) #the -1 removes the intercept term
```

```
      x      x2      x3
1.0040 0.9899 1.0078
```

# Interpretation of the Coefficient

$$E[Y|X_1 = x_1, \dots, X_p = x_p] = \sum_{k=1}^p x_k \beta_k$$

So that

$$\begin{aligned} E[Y|X_1 = x_1 + 1, \dots, X_p = x_p] - E[Y|X_1 = x_1, \dots, X_p = x_p] \\ = (x_1 + 1)\beta_1 + \sum_{k=2}^p x_k + \sum_{k=1}^p x_k \beta_k = \beta_1 \end{aligned}$$

So that the interpretation of a multivariate regression coefficient is the expected change in the response per unit change in the regressor, holding all of the other regressors fixed.

In the next lecture, we'll do examples and go over context-specific interpretations.

# Fitted val, residuals, and residual variation

All of our SLR quantities can be extended to linear models

- Model  $Y_i = \sum_{k=1}^p X_{ik} \beta_k + \epsilon_i$  where  $\epsilon_i \sim N(0, \sigma^2)$
- Fitted responses  $\hat{Y}_i = \sum_{k=1}^p X_{ik} \hat{\beta}_k$
- Residuals  $e_i = Y_i - \hat{Y}_i$
- Variance estimate  $\hat{\sigma}^2 = \frac{1}{n-p} \sum_{i=1}^n e_i^2$
- To get predicted responses at new values,  $x_1, \dots, x_p$ , simply plug them into the linear model  $\sum_{k=1}^p x_k \hat{\beta}_k$
- Coefficients have standard errors,  $\hat{\sigma}_{\hat{\beta}_k}$ , and  $\frac{\hat{\beta}_k - \beta_k}{\hat{\sigma}_{\hat{\beta}_k}}$  follows a  $T$  distribution with  $n - p$  degrees of freedom.
- Predicted responses have standard errors and we can calculate predicted and expected response intervals.

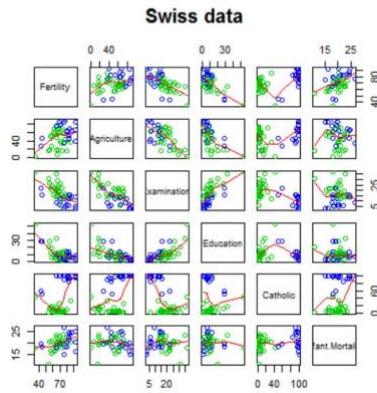
# Linear Models

- Linear models are the single most important applied statistical and machine learning technique, by far.
- Some amazing things that you can accomplish with linear models
  - Decompose a signal into its harmonics.
  - Flexibly fit complicated functions.
  - Fit factor variables as predictors.
  - Uncover complex multivariate relationships with the response.
  - Build accurate prediction models.

# Multivariable Regression Examples

## Swiss Fertility Data

```
library(datasets); data(swiss); require(stats); require(graphics)
pairs(swiss, panel = panel.smooth, main = "Swiss data", col = 3 + (swiss$Catholic > 50))
```



## ? SWISS

### Description

Standardized fertility measure and socio-economic indicators for each of 47 French-speaking provinces of Switzerland at about 1888.

A data frame with 47 observations on 6 variables, each of which is in percent, i.e., in [0, 100].

- [1] Fertility lg, 'common standardized fertility measure'
- [2] Agriculture % of males involved in agriculture as occupation
- [3] Examination % draftees receiving highest mark on army examination
- [4] Education % education beyond primary school for draftees.
- [5] Catholic % 'catholic' (as opposed to 'protestant').
- [6] Infant.Mortality live births who live less than 1 year.

All variables but 'Fertility' give proportions of the population.

## Calling lm

```
summary(lm(Fertility ~ . , data = swiss))
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	66.9152	10.70604	6.250	1.906e-07
Agriculture	-0.1721	0.07030	-2.448	1.873e-02
Examination	-0.2580	0.25388	-1.016	3.155e-01
Education	-0.8709	0.18303	-4.758	2.431e-05
Catholic	0.1041	0.03526	2.953	5.190e-03
Infant.Mortality	1.0770	0.38172	2.822	7.336e-03

# Example Interpretation

- Agriculture is expressed in percentages (0 - 100)
- Estimate is -0.1721.
- We estimate an expected 0.17 decrease in standardized fertility for every 1% increase in percentage of males involved in agriculture in holding the remaining variables constant.
- The t-test for  $H_0 : \beta_{Agric} = 0$  versus  $H_a : \beta_{Agric} \neq 0$  is significant.
- Interestingly, the unadjusted estimate is

```
summary(lm(Fertility ~ Agriculture, data = swiss))$coefficients
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	60.3044	4.25126	14.185	3.21e-18
Agriculture	0.1942	0.07671	2.532	1.492e-02

How can adjustment reverse the sign of an effect? Let's try a simulation.

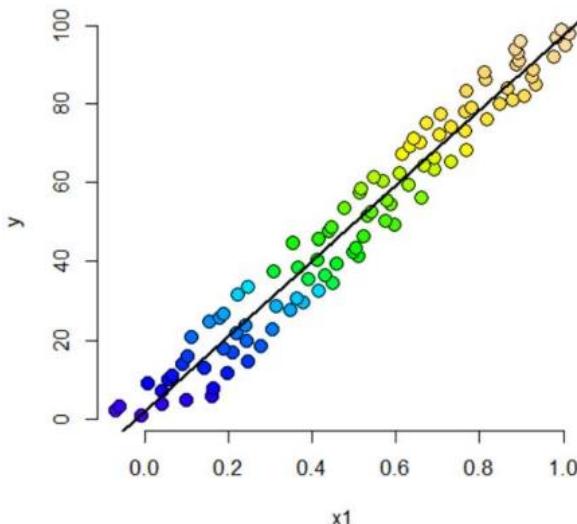
```
n <- 100; x2 <- 1 : n; x1 <- .01 * x2 + runif(n, -.1, .1); y = -x1 + x2 + rnorm(n, sd = .01)
summary(lm(y ~ x1))$coef
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	1.618	1.200	1.349	1.806e-01
x1	95.854	2.058	46.579	1.153e-68

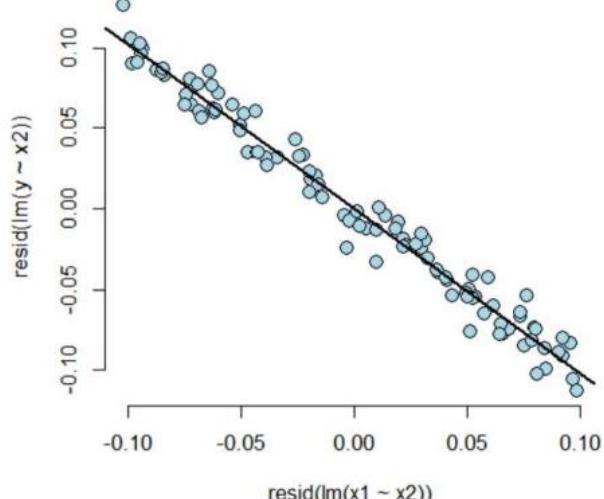
```
summary(lm(y ~ x1 + x2))$coef
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	0.0003683	0.0020141	0.1829	8.553e-01
x1	-1.0215256	0.0166372	-61.4001	1.922e-79
x2	1.0001909	0.0001681	5950.1818	1.369e-271

Unadjusted, color is X2



Adjusted



# In this Dataset

- The sign reverses itself with the inclusion of Examination and Education, but of which are negatively correlated with Agriculture.
- The percent of males in the province working in agriculture is negatively related to educational attainment (correlation of -0.6395) and Education and Examination (correlation of 0.6984) are obviously measuring similar things.
  - Is the positive marginal an artifact for not having accounted for, say, Education level? (Education does have a stronger effect, by the way.)
- At the minimum, anyone claiming that provinces that are more agricultural have higher fertility rates would immediately be open to criticism.

## If we include an unnecessary variable?

$z$  adds no new linear information, since it's a linear combination of variables already included. R just drops terms that are linear combinations of other terms.

```
z <- swiss$Agriculture + swiss$Education  
lm(Fertility ~ . + z, data = swiss)
```

```
Call:  
lm(formula = Fertility ~ . + z, data = swiss)  
  
Coefficients:  
(Intercept) Agriculture Examination Education Catholic  
66.915 -0.172 -0.258 -0.871 0.104  
Infant.Mortality z  
1.077 NA
```

## Dummy Variables are Smart

- Consider the linear model

$$Y_i = \beta_0 + X_{i1}\beta_1 + \epsilon_i$$

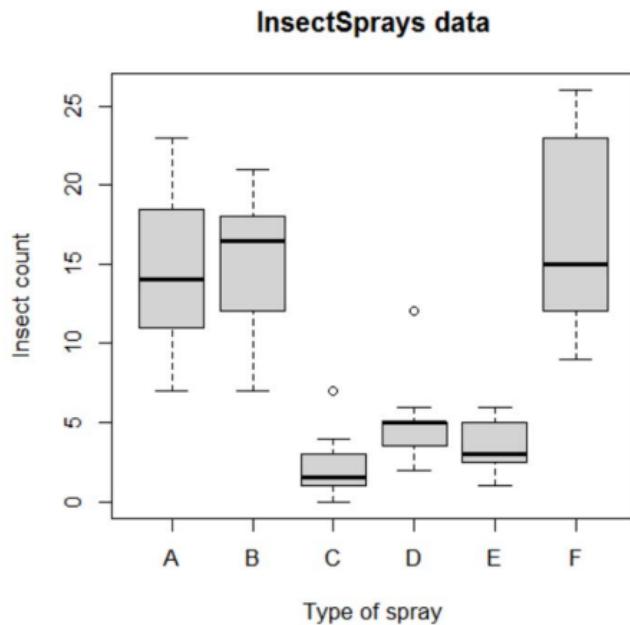
where each  $X_{i1}$  is binary so that it is a 1 if measurement  $i$  is in a group and 0 otherwise. (Treated versus not in a clinical trial, for example.)

- Then for people in the group  $E[Y_i] = \beta_0 + \beta_1$
- And for people not in the group  $E[Y_i] = \beta_0$
- The LS fits work out to be  $\hat{\beta}_0 + \hat{\beta}_1$  is the mean for those in the group and  $\hat{\beta}_0$  is the mean for those not in the group.
- $\beta_1$  is interpreted as the increase or decrease in the mean comparing those in the group to those not.
- Note including a binary variable that is 1 for those not in the group would be redundant. It would create three parameters to describe two means.

# More than 2 levels

- Consider a multilevel factor level. For didactic reasons, let's say a three level factor (example, US political party affiliation: Republican, Democrat, Independent)
- $Y_i = \beta_0 + X_{i1}\beta_1 + X_{i2}\beta_2 + \epsilon_i$ .
- $X_{i1}$  is 1 for Republicans and 0 otherwise.
- $X_{i2}$  is 1 for Democrats and 0 otherwise.
- If  $i$  is Republican  $E[Y_i] = \beta_0 + \beta_1$
- If  $i$  is Democrat  $E[Y_i] = \beta_0 + \beta_2$ .
- If  $i$  is Independent  $E[Y_i] = \beta_0$ .
- $\beta_1$  compares Republicans to Independents.
- $\beta_2$  compares Democrats to Independents.
- $\beta_1 - \beta_2$  compares Republicans to Democrats.
- (Choice of reference category changes the interpretation.)

## Insect Sprays



# Linear Model Fit, Group A as Reference

```
summary(lm(count ~ spray, data = InsectSprays))$coef
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	14.5000	1.132	12.8074	1.471e-19
sprayB	0.8333	1.601	0.5205	6.045e-01
sprayC	-12.4167	1.601	-7.7550	7.267e-11
sprayD	-9.5833	1.601	-5.9854	9.817e-08
sprayE	-11.0000	1.601	-6.8702	2.754e-09
sprayF	2.1667	1.601	1.3532	1.806e-01

## Hard coding dummy Variables

```
summary(lm(count ~
           I(1 * (spray == "B")) + I(1 * (spray == "C")) +
           I(1 * (spray == "D")) + I(1 * (spray == "E")) +
           I(1 * (spray == "F")),
           data = InsectSprays))$coef
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	14.5000	1.132	12.8074	1.471e-19
I(1 * (spray == "B"))	0.8333	1.601	0.5205	6.045e-01
I(1 * (spray == "C"))	-12.4167	1.601	-7.7550	7.267e-11
I(1 * (spray == "D"))	-9.5833	1.601	-5.9854	9.817e-08
I(1 * (spray == "E"))	-11.0000	1.601	-6.8702	2.754e-09
I(1 * (spray == "F"))	2.1667	1.601	1.3532	1.806e-01

## Including 6 Variables

```
summary(lm(count ~
           I(1 * (spray == "B")) + I(1 * (spray == "C")) +
           I(1 * (spray == "D")) + I(1 * (spray == "E")) +
           I(1 * (spray == "F")),
           data = InsectSprays))$coef
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	14.5000	1.132	12.8074	1.471e-19
I(1 * (spray == "B"))	0.8333	1.601	0.5205	6.045e-01
I(1 * (spray == "C"))	-12.4167	1.601	-7.7550	7.267e-11
I(1 * (spray == "D"))	-9.5833	1.601	-5.9854	9.817e-08
I(1 * (spray == "E"))	-11.0000	1.601	-6.8702	2.754e-09
I(1 * (spray == "F"))	2.1667	1.601	1.3532	1.806e-01

# Omitting the Intercept

```
summary(lm(count ~ spray - 1, data = InsectSprays))$coef
```

	Estimate	Std. Error	t value	Pr(> t )
sprayA	14.500	1.132	12.807	1.471e-19
sprayB	15.333	1.132	13.543	1.002e-20
sprayC	2.083	1.132	1.840	7.024e-02
sprayD	4.917	1.132	4.343	4.953e-05
sprayE	3.500	1.132	3.091	2.917e-03
sprayF	16.667	1.132	14.721	1.573e-22

```
unique(ave(InsectSprays$count, InsectSprays$spray))
```

```
[1] 14.500 15.333 2.083 4.917 3.500 16.667
```

# Summary

- If we treat Spray as a factor, R includes an intercept and omits the alphabetically first level of the factor.
  - All t-tests are for comparisons of Sprays versus Spray A.
  - Empirical mean for A is the intercept.
  - Other group means are the itc plus their coefficient.
- If we omit an intercept, then it includes terms for all levels of the factor.
  - Group means are the coefficients.
  - Tests are tests of whether the groups are different than zero. (Are the expected counts zero for that spray.)
- If we want comparisons between, Spray B and C, say we could refit the model with C (or B) as the reference level.

# Reordering the Levels

```
spray2 <- relevel(InsectSprays$spray, "C")
summary(lm(count ~ spray2, data = InsectSprays))$coef
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	2.083	1.132	1.8401	7.024e-02
spray2A	12.417	1.601	7.7550	7.267e-11
spray2B	13.250	1.601	8.2755	8.510e-12
spray2D	2.833	1.601	1.7696	8.141e-02
spray2E	1.417	1.601	0.8848	3.795e-01
spray2F	14.583	1.601	9.1083	2.794e-13

# Manually

Equivalently

$$Var(\hat{\beta}_B - \hat{\beta}_C) = Var(\hat{\beta}_B) + Var(\hat{\beta}_C) - 2Cov(\hat{\beta}_B, \hat{\beta}_C)$$

```
fit <- lm(count ~ spray, data = InsectSprays) #A is ref
bbmbc <- coef(fit)[2] - coef(fit)[3] #B - C
temp <- summary(fit)
se <- temp$sigma * sqrt(temp$cov.unscaled[2, 2] + temp$cov.unscaled[3, 3] - 2 * temp$cov.unscaled[2, 3]
t <- (bbmbc) / se
p <- pt(-abs(t), df = fit$df)
out <- c(bbmbc, se, t, p)
names(out) <- c("B - C", "SE", "T", "P")
round(out, 3)
```

B - C	SE	T	P
13.250	1.601	8.276	0.000

- Counts are bounded from below by 0, violates the assumption of normality of the errors.
  - Also there are counts near zero, so both the actual assumption and the intent of the assumption are violated.
- Variance does not appear to be constant.
- Perhaps taking logs of the counts would help.
  - There are 0 counts, so maybe log(Count + 1)
- Also, we'll cover Poisson GLMs for fitting count data.

## Example- Millennium Development Goal 1

[http://www.un.org/millenniumgoals/pdf/MDG\\_FS\\_1\\_EN.pdf](http://www.un.org/millenniumgoals/pdf/MDG_FS_1_EN.pdf)

[http://apps.who.int/gho/athena/data/GHO/WHOSIS\\_000008.csv?profile=text&filter=COUNTRY;;SEX:](http://apps.who.int/gho/athena/data/GHO/WHOSIS_000008.csv?profile=text&filter=COUNTRY;;SEX:)

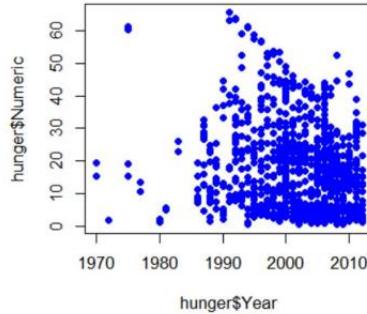
## WHO Childhood Hunger Data

```
#download.file("http://apps.who.int/gho/athena/data/GHO/WHOSIS_000008.csv?profile=text&filter=COUNT
hunger <- read.csv("hunger.csv")
hunger <- hunger[hunger$Sex!="Both sexes",]
head(hunger)
```

	Indicator	Data.Source	PUBLISH STATES	Year	WHO.region		
1	Children aged <5 years underweight (%)	NLIS_310044	Published	1986	Africa		
2	Children aged <5 years underweight (%)	NLIS_310233	Published	1990	Americas		
3	Children aged <5 years underweight (%)	NLIS_312902	Published	2005	Americas		
5	Children aged <5 years underweight (%)	NLIS_312522	Published	2002	Eastern Mediterranean		
6	Children aged <5 years underweight (%)	NLIS_312955	Published	2008	Africa		
8	Children aged <5 years underweight (%)	NLIS_312963	Published	2008	Africa		
	Country	Sex	Display.Value	Numeric	Low	High	Comments
1	Senegal	Male	19.3	19.3	NA	NA	NA
2	Paraguay	Male	2.2	2.2	NA	NA	NA
3	Nicaragua	Male	5.3	5.3	NA	NA	NA
5	Jordan	Female	3.2	3.2	NA	NA	NA
6	Guinea-Bissau	Female	17.0	17.0	NA	NA	NA
8	Ghana	Male	15.7	15.7	NA	NA	NA

# Plot Percent Hungry vs. Time

```
lm1 <- lm(hunger$Numeric ~ hunger$Year)
plot(hunger$Year, hunger$Numeric, pch=19, col="blue")
```



## Add Linear Model

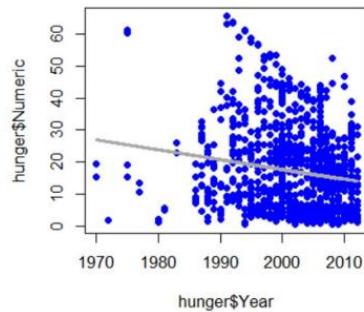
$$Hu_i = b_0 + b_1 Y_i + e_i$$

$b_0$  = percent hungry at Year 0

$b_1$  = decrease in percent hungry per year

$e_i$  = everything we didn't measure

```
lm1 <- lm(hunger$Numeric ~ hunger$Year)
plot(hunger$Year, hunger$Numeric, pch=19, col="blue")
lines(hunger$Year, lm1$fitted, lwd=3, col="darkgrey")
```



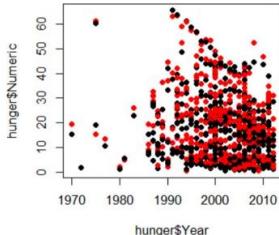
# Color by male / female

```

plot(hunger$Year,hunger$Numeric,pch=19)
points(hunger$Year,hunger$Numeric,pch=19,col=((hunger$Sex=="Male")*1+1))

lmM <- lm(hunger$Numeric[hunger$Sex=="Male"] ~ hunger$Year[hunger$Sex=="Male"])
lmF <- lm(hunger$Numeric[hunger$Sex=="Female"] ~ hunger$Year[hunger$Sex=="Female"])
plot(hunger$Year,hunger$Numeric,pch=19)
points(hunger$Year,hunger$Numeric,pch=19,col=((hunger$Sex=="Male")*1+1))
lines(hunger$Year[hunger$Sex=="Male"],lmM$fitted,col="black",lwd=3)
lines(hunger$Year[hunger$Sex=="Female"],lmF$fitted,col="red",lwd=3)

```



$$HuF_i = b f_0 + b f_1 Y F_i + e_f i$$

$b f_0$  = percent of girls hungry at Year 0

$b f_1$  = decrease in percent of girls hungry per year

$e f_i$  = everything we didn't measure

$$Hu_i = b_0 + b_1 1(Sex_i = "Male") + b_2 Y_i + e_i^*$$

$$HuM_i = b m_0 + b m_1 Y M_i + e m_i \quad b_0 - \text{percent hungry at year zero for females}$$

$b m_0$  = percent of boys hungry at Year 0

$b_0 + b_1$  - percent hungry at year zero for males

$b m_1$  = decrease in percent of boys hungry per year

$b_2$  - change in percent hungry (for either males or females) in one year

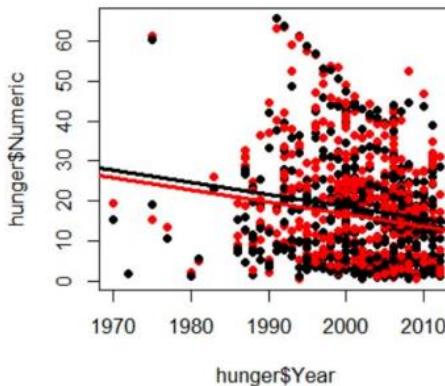
$e m_i$  = everything we didn't measure

$e_i^*$  - everything we didn't measure

```

lmBoth <- lm(hunger$Numeric ~ hunger$Year + hunger$Sex)
plot(hunger$Year,hunger$Numeric,pch=19)
points(hunger$Year,hunger$Numeric,pch=19,col=((hunger$Sex=="Male")*1+1))
abline(c(lmBoth$coeff[1],lmBoth$coeff[2]),col="red",lwd=3)
abline(c(lmBoth$coeff[1] + lmBoth$coeff[3],lmBoth$coeff[2] ),col="black",lwd=3)

```



# Two Lines, Different Slopes (Interactions)

$$Hu_i = b_0 + b_1 1(Sex_i = "Male") + b_2 Y_i + b_3 1(Sex_i = "Male") \times Y_i + e_i^+$$

$b_0$  - percent hungry at year zero for females

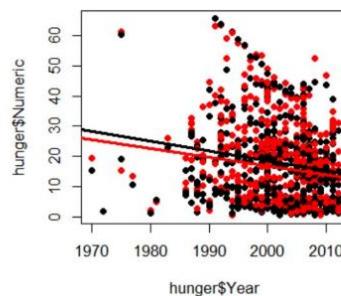
$b_0 + b_1$  - percent hungry at year zero for males

$b_2$  - change in percent hungry (females) in one year

$b_2 + b_3$  - change in percent hungry (males) in one year

$e_i^+$  - everything we didn't measure

```
lmBoth <- lm(hunger$Numeric ~ hunger$Year + hunger$Sex + hunger$Sex*hunger$Year)
plot(hunger$Year,hunger$Numeric,pch=19)
points(hunger$Year,hunger$Numeric,pch=19,col=((hunger$Sex=="Male")*1+1))
abline(c(lmBoth$coeff[1],lmBoth$coeff[2]),col="red",lwd=3)
abline(c(lmBoth$coeff[1] + lmBoth$coeff[3],lmBoth$coeff[2] + lmBoth$coeff[4]),col="black",lwd=3)
```



```
summary(lmBoth)
```

```
Call:
lm(formula = hunger$Numeric ~ hunger$Year + hunger$Sex + hunger$Sex *
    hunger$Year)

Residuals:
    Min      1Q  Median      3Q     Max 
-25.91 -11.25 -1.85   7.09  46.15 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 603.5058   171.0552   3.53  0.00044 ***
hunger$Year  -0.2934     0.0855  -3.43  0.00062 ***
hunger$SexMale 61.9477   241.9086   0.26  0.79795  
hunger$Year:hunger$SexMale -0.0300     0.1209  -0.25  0.80402  
---
Signif. codes:  0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.2 on 944 degrees of freedom
Multiple R-squared:  0.0318,    Adjusted R-squared:  0.0287 
F-statistic: 10.3 on 3 and 944 DF,  p-value: 1.06e-06
```

# Interpreting a Continuous Interaction

$$E[Y_i | X_{1i} = x_1, X_{2i} = x_2] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2$$

Holding  $X_2$  constant we have

$$E[Y_i | X_{1i} = x_1 + 1, X_{2i} = x_2] - E[Y_i | X_{1i} = x_1, X_{2i} = x_2] = \beta_1 + \beta_3 x_2$$

And thus the expected change in  $Y$  per unit change in  $X_1$  holding all else constant is not constant.

$\beta_1$  is the slope when  $x_2 = 0$ . Note further that:

$$E[Y_i | X_{1i} = x_1 + 1, X_{2i} = x_2 + 1] - E[Y_i | X_{1i} = x_1, X_{2i} = x_2 + 1]$$

$$-E[Y_i | X_{1i} = x_1 + 1, X_{2i} = x_2] - E[Y_i | X_{1i} = x_1, X_{2i} = x_2]$$

$$= \beta_3$$

Thus,  $\beta_3$  is the change in the expected change in  $Y$  per unit change in  $X_1$ , per unit change in  $X_2$ .

Or, the change in the slope relating  $X_1$  and  $Y$  per unit change in  $X_2$ .

## Example

$$Hu_i = b_0 + b_1 In_i + b_2 Y_i + b_3 In_i \times Y_i + e_i^+$$

$b_0$  - percent hungry at year zero for children with whose parents have no income

$b_1$  - change in percent hungry for each dollar of income in year zero

$b_2$  - change in percent hungry in one year for children whose parents have no income

$b_3$  - increased change in percent hungry by year for each dollar of income - e.g. if income is \$10,000, then change in percent hungry in one year will be

$$b_2 + 1e4 \times b_3$$

$e_i^+$  - everything we didn't measure

**Lot's of care/caution needed!**

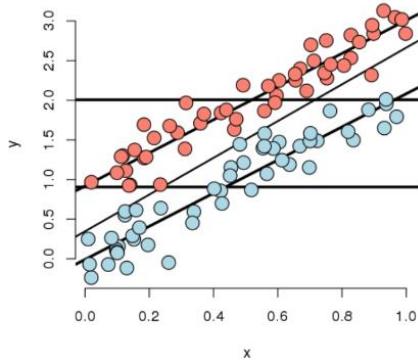
# Multivariable Regression

## Consider the Following Simulated Data

Code for the first plot, rest omitted (See the git repo for the rest of the code.)

```
n <- 100; t <- rep(c(0, 1), c(n/2, n/2)); x <- c(runif(n/2), runif(n/2));
beta0 <- 0; beta1 <- 2; tau <- 1; sigma <- .2
y <- beta0 + x * beta1 + t * tau + rnorm(n, sd = sigma)
plot(x, y, type = "n", frame = FALSE)
abline(lm(y ~ x), lwd = 2)
abline(h = mean(y[1 : (n/2)]), lwd = 3)
abline(h = mean(y[(n/2 + 1) : n]), lwd = 3)
fit <- lm(y ~ x + t)
abline(coef(fit)[1], coef(fit)[2], lwd = 3)
abline(coef(fit)[1] + coef(fit)[3], coef(fit)[2], lwd = 3)
points(x[1 : (n/2)], y[1 : (n/2)], pch = 21, col = "black", bg = "lightblue", cex = 2)
points(x[(n/2 + 1) : n], y[(n/2 + 1) : n], pch = 21, col = "black", bg = "salmon", cex = 2)
```

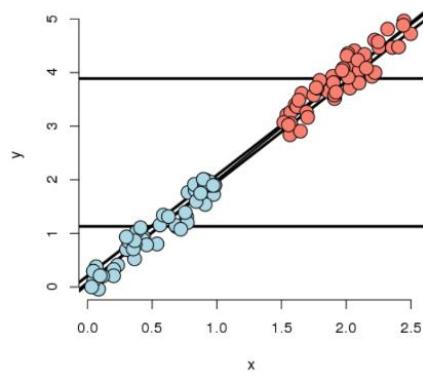
## Simulation 1



Some things to note in this simulation

- The X variable is unrelated to group status
- The X variable is related to Y, but the intercept depends on group status.
- The group variable is related to Y.
  - The relationship between group status and Y is constant depending on X.
  - The relationship between group and Y disregarding X is about the same as holding X constant

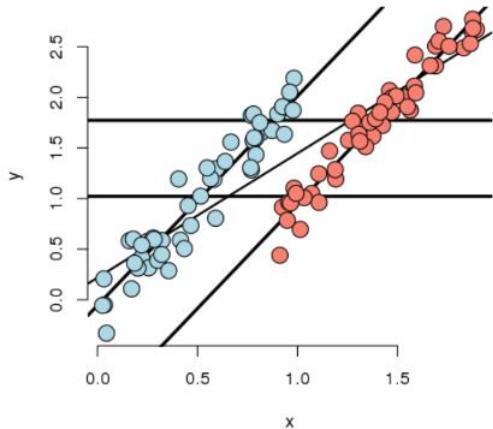
## Simulation 2



Some things to note in this simulation

- The X variable is highly related to group status
- The X variable is related to Y, the intercept doesn't depend on the group variable.
  - The X variable remains related to Y holding group status constant
- The group variable is marginally related to Y disregarding X.
- The model would estimate no adjusted effect due to group.
  - There isn't any data to inform the relationship between group and Y.
  - This conclusion is entirely based on the model.

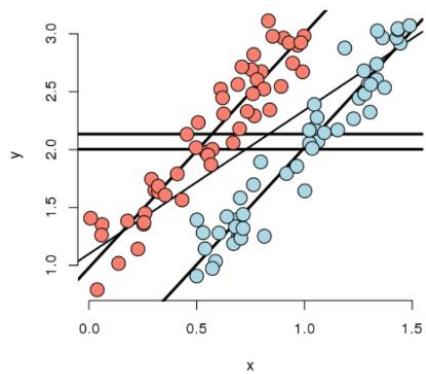
## Simulation 3



Some things to note in this simulation

- Marginal association has red group higher than blue.
- Adjusted relationship has blue group higher than red.
- Group status related to X.
- There is some direct evidence for comparing red and blue holding X fixed.

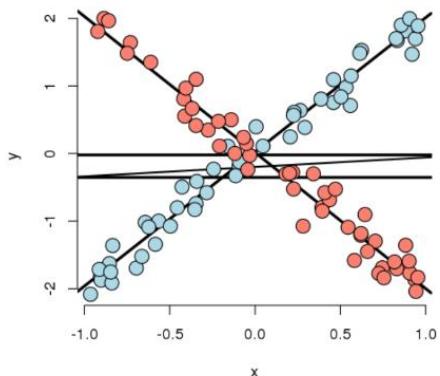
## Simulation 4



Some things to note in this simulation

- No marginal association between group status and Y.
- Strong adjusted relationship.
- Group status not related to X.
- There is lots of direct evidence for comparing red and blue holding X fixed.

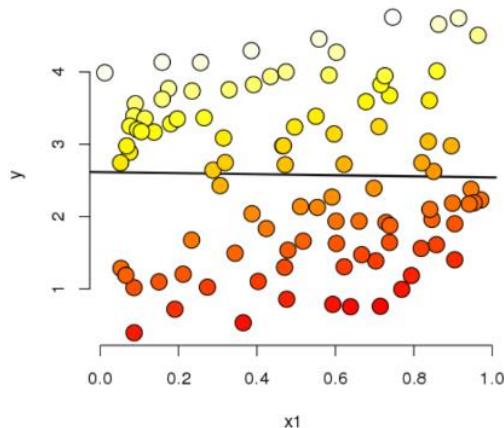
## Simulation 5



Some things to note from this simulation

- There is no such thing as a group effect here.
  - The impact of group reverses itself depending on X.
  - Both intercept and slope depends on group.
- Group status and X unrelated.
  - There's lots of information about group effects holding X fixed.

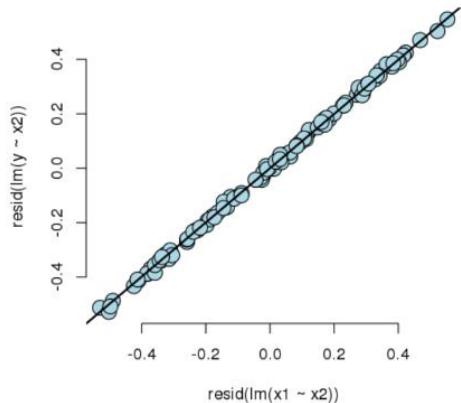
# Simulation 6



Do this to investigate the bivariate relationship

```
library(rgl)  
plot3d(x1, x2, y)
```

## Residual Relationship



Some things to note from this simulation

- X1 unrelated to X2
- X2 strongly related to Y
- Adjusted relationship between X1 and Y largely unchanged by considering X2.
  - Almost no residual variability after accounting for X2.

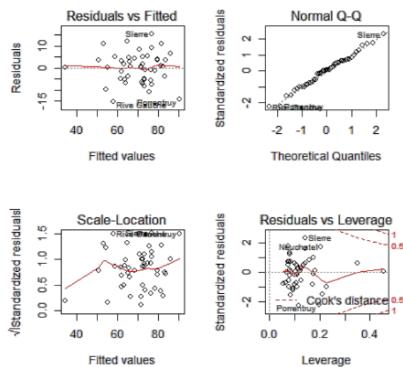
## Final Thoughts

- Modeling multivariate relationships is difficult.
- Play around with simulations to see how the inclusion or exclusion of another variable can change analyses.
- The results of these analyses deal with the impact of variables on associations.
  - Ascertaining mechanisms or cause are difficult subjects to be added on top of difficulty in understanding multivariate associations.

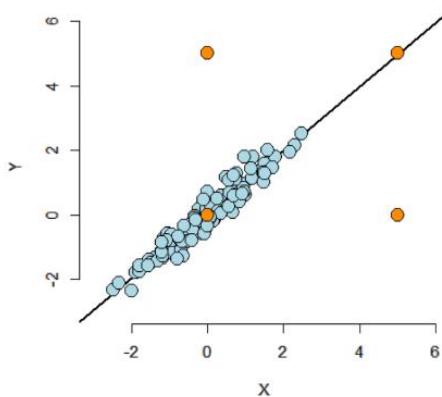
# Residuals, Diagnostics, Variation Linear Model

- Specified as  $Y_i = \sum_{k=1}^p X_{ik} \beta_j + \epsilon_i$
- We'll also assume here that  $\epsilon_i \stackrel{iid}{\sim} N(0, \sigma^2)$
- Define the residuals as  $e_i = Y_i - \hat{Y}_i = Y_i - \sum_{k=1}^p X_{ik} \hat{\beta}_j$
- Our estimate of residual variation is  $\hat{\sigma}^2 = \frac{\sum_{i=1}^n e_i^2}{n-p}$ , the  $n-p$  so that  $E[\hat{\sigma}^2] = \sigma^2$

```
data(swiss); par(mfrow = c(2, 2))
fit <- lm(Fertility ~ ., data = swiss); plot(fit)
```



## Influential, High Leverage and Outlying Pts.



Calling a point an outlier is vague.

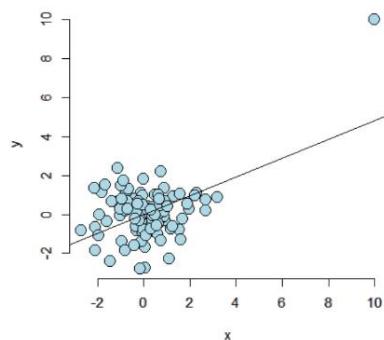
- Outliers can be the result of spurious or real processes.
- Outliers can have varying degrees of influence.
- Outliers can conform to the regression relationship (i.e. being marginally outlying in X or Y, but not outlying given the regression relationship).
  - Upper left hand point has low leverage, low influence, outliers in a way not conforming to the regression relationship.
  - Lower left hand point has low leverage, low influence and is not to be an outlier in any sense.
  - Upper right hand point has high leverage, but chooses not to exert it and thus would have low actual influence by conforming to the regression relationship of the other points.
  - Lower right hand point has high leverage and would exert it if it were included in the fit.

# Influence Measures

- Do `?influence.measures` to see the full suite of influence measures in stats. The measures include
  - `rstandard` - standardized residuals, residuals divided by their standard deviations)
  - `rstudent` - standardized residuals, residuals divided by their standard deviations, where the  $i^{\text{th}}$  data point was deleted in the calculation of the standard deviation for the residual to follow a t distribution
  - `hatvalues` - measures of leverage
  - `dffits` - change in the predicted response when the  $i^{\text{th}}$  point is deleted in fitting the model.
  - `dfbetas` - change in individual coefficients when the  $i^{\text{th}}$  point is deleted in fitting the model.
  - `cooks.distance` - overall change in the coefficients when the  $i^{\text{th}}$  point is deleted.
  - `resid` - returns the ordinary residuals
  - `resid(fit) / (1 - hatvalues(fit))` where `fit` is the linear model fit returns the PRESS residuals, i.e. the leave one out cross validation residuals - the difference in the response and the predicted response at data point  $i$ , where it was not included in the model fitting.

## How do I Use all of these Things?

### Case 1



```
n <- 100; x <- c(10, rnorm(n)); y <- c(10, rnorm(n))
plot(x, y, frame = FALSE, cex = 2, pch = 21, bg = "lightblue", col = "black")
abline(lm(y ~ x))
```

- The point `c(10, 10)` has created a strong regression relationship where there shouldn't be one.

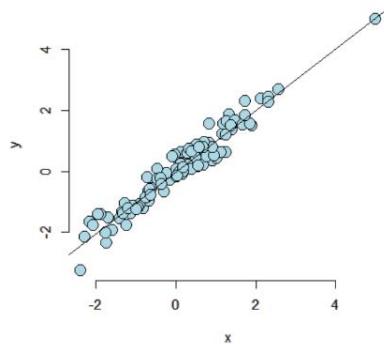
```
fit <- lm(y ~ x)
round(dfbetas(fit)[1 : 10, 2], 3)
```

1	2	3	4	5	6	7	8	9	10
6.007	-0.019	-0.007	0.014	-0.002	-0.083	-0.034	-0.045	-0.112	-0.008

```
round(hatvalues(fit)[1 : 10], 3)
```

1	2	3	4	5	6	7	8	9	10
0.445	0.010	0.011	0.011	0.030	0.017	0.012	0.033	0.021	0.010

### Case 2



```
round(dfbetas(fit2)[1 : 10, 2], 3)
```

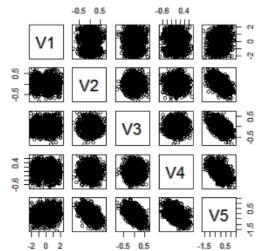
1	2	3	4	5	6	7	8	9	10
-0.072	-0.041	-0.007	0.012	0.008	-0.187	0.017	0.100	-0.059	0.035

```
round(hatvalues(fit2)[1 : 10], 3)
```

1	2	3	4	5	6	7	8	9	10
0.164	0.011	0.014	0.012	0.010	0.030	0.017	0.017	0.013	0.021

# Example- Stefanski Tas 2007 Vol 61

```
## Don't everyone hit this server at once. Read the paper first.  
dat <- read.table('http://www4.stat.ncsu.edu/~stefanski/NSF_Supported/Hidden_Images/orly_owl_files/orly  
pairs(dat)
```



## P-values, why bother with a residual plot?

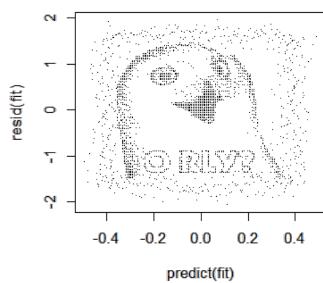
```
summary(lm(V1 ~ . -1, data = dat))$coef
```

	Estimate	Std. Error	t value	Pr(> t )
V2	0.9856	0.12798	7.701	1.989e-14
V3	0.9715	0.12664	7.671	2.500e-14
V4	0.8606	0.11958	7.197	8.301e-13
V5	0.9267	0.08328	11.127	4.778e-28

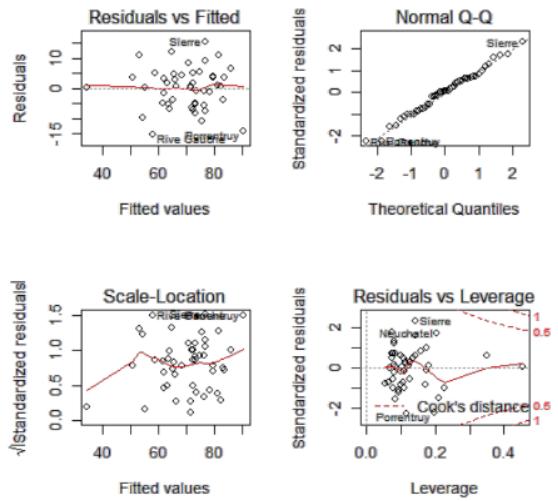
## Residual Plot

P-values significant, O RLY?

```
fit <- lm(V1 ~ . - 1, data = dat); plot(predict(fit), resid(fit), pch = '.')
```



# Back to Swiss Data



# MultiVariable – Regression

- We have an entire class on prediction and machine learning, so we'll focus on modeling.
  - Prediction has a different set of criteria, needs for interpretability and standards for generalizability.
  - In modeling, our interest lies in parsimonious, interpretable representations of the data that enhance our understanding of the phenomena under study.
  - A model is a lens through which to look at your data. (I attribute this quote to Scott Zeger)
  - Under this philosophy, what's the right model? Whatever model connects the data to a true, parsimonious statement about what you're studying.
- There are nearly uncountable ways that a model can be wrong, in this lecture, we'll focus on variable inclusion and exclusion.
- Like nearly all aspects of statistics, good modeling decisions are context dependent.
  - A good model for prediction versus one for studying mechanisms versus one for trying to establish causal effects may not be the same.

## Rumsfeldian Triplet

*There are known knowns. These are things we know that we know. There are known unknowns. That is to say, there are things that we know we don't know. But there are also unknown unknowns. There are things we don't know we don't know.* Donald Rumsfeld

In our context

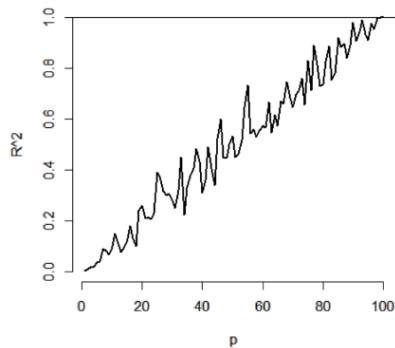
- (Known knowns) Regressors that we know we should check to include in the model and have.
- (Known Unknowns) Regressors that we would like to include in the model, but don't have.
- (Unknown Unknowns) Regressors that we don't even know about that we should have included in the model.

## General Rules

- Omitting variables results in bias in the coefficients of interest - unless their regressors are uncorrelated with the omitted ones.
  - This is why we randomize treatments, it attempts to uncorrelate our treatment indicator with variables that we don't have to put in the model.
  - (If there's too many unobserved confounding variables, even randomization won't help you.)
- Including variables that we shouldn't have increases standard errors of the regression variables.
  - Actually, including any new variables increases (actual, not estimated) standard errors of other regressors. So we don't want to idly throw variables into the model.
- The model must tend toward perfect fit as the number of non-redundant regressors approaches n.
- $R^2$  increases monotonically as more regressors are included.
- The SSE decreases monotonically as more regressors are included.

# Plot of R^2 Versus n

For simulations as the number of variables included equals increases to n = 100. No actual regression relationship exist in any simulation



## Variance Inflation

```
n <- 100; nosim <- 1000
x1 <- rnorm(n); x2 <- rnorm(n); x3 <- rnorm(n);
betas <- sapply(1 : nosim, function(i){
  y <- x1 + rnorm(n, sd = .3)
  c(coef(lm(y ~ x1))[2],
    coef(lm(y ~ x1 + x2))[2],
    coef(lm(y ~ x1 + x2 + x3))[2])
})
round(apply(betas, 1, sd), 5)
```

```
n <- 100; nosim <- 1000
x1 <- rnorm(n); x2 <- x1/sqrt(2) + rnorm(n) /sqrt(2)
x3 <- x1 * 0.95 + rnorm(n) * sqrt(1 - 0.95^2);
betas <- sapply(1 : nosim, function(i){
  y <- x1 + rnorm(n, sd = .3)
  c(coef(lm(y ~ x1))[2],
    coef(lm(y ~ x1 + x2))[2],
    coef(lm(y ~ x1 + x2 + x3))[2])
})
round(apply(betas, 1, sd), 5)
```

x1	x1	x1
0.02839	0.02872	0.02884

x1	x1	x1
0.03131	0.04270	0.09653

## Variance Inflation Factors

- Notice variance inflation was much worse when we included a variable that was highly related to x1.
- We don't know  $\sigma$ , so we can only estimate the increase in the actual standard error of the coefficients for including a regressor.
- However,  $\sigma$  drops out of the relative standard errors. If one sequentially adds variables, one can check the variance (or sd) inflation for including each one.
- When the other regressors are actually orthogonal to the regressor of interest, then there is no variance inflation.
- The variance inflation factor (VIF) is the increase in the variance for the i<sup>th</sup> regressor compared to the ideal setting where it is orthogonal to the other regressors.
  - (The square root of the VIF is the increase in the sd ...)
- Remember, variance inflation is only part of the picture. We want to include certain variables, even if they dramatically inflate our variance.

# Previous Simulation

```
##doesn't depend on which y you use,  
y <- x1 + rnorm(n, sd = .3)  
a <- summary(lm(y ~ x1))$cov.unscaled[2,2]  
c(summary(lm(y ~ x1 + x2))$cov.unscaled[2,2],  
  summary(lm(y ~ x1 + x2 + x3))$cov.unscaled[2,2]) / a
```

```
[1] 1.895 9.948
```

```
temp <- apply(betas, 1, var); temp[2 : 3] / temp[1]
```

```
x1 x1  
1.860 9.506
```

# Swiss Data

```
data(swiss);  
fit1 <- lm(Fertility ~ Agriculture, data = swiss)  
a <- summary(fit1)$cov.unscaled[2,2]  
fit2 <- update(fit, Fertility ~ Agriculture + Examination)  
fit3 <- update(fit, Fertility ~ Agriculture + Examination + Education)  
c(summary(fit2)$cov.unscaled[2,2],  
  summary(fit3)$cov.unscaled[2,2]) / a
```

```
[1] 1.892 2.089
```

# VIFs

```
library(car)  
fit <- lm(Fertility ~ . , data = swiss)  
vif(fit)
```

Agriculture	Examination	Education	Catholic	Infant.Mortality
2.284	3.675	2.775	1.937	1.108

```
sqrt(vif(fit)) #I prefer sd
```

Agriculture	Examination	Education	Catholic	Infant.Mortality
1.511	1.917	1.666	1.392	1.052

# What about residual variance estimation?

- Assuming that the model is linear with additive iid errors (with finite variance), we can mathematically describe the impact of omitting necessary variables or including unnecessary ones.
  - If we underfit the model, the variance estimate is biased.
  - If we correctly or overfit the model, including all necessary covariates and/or unnecessary covariates, the variance estimate is unbiased.
  - However, the variance of the variance is larger if we include unnecessary variables.

# Covariate Model Selection

- Automated covariate selection is a difficult topic. It depends heavily on how rich of a covariate space one wants to explore.
  - The space of models explodes quickly as you add interactions and polynomial terms.
- In the prediction class, we'll cover many modern methods for traversing large model spaces for the purposes of prediction.
- Principal components or factor analytic models on covariates are often useful for reducing complex covariate spaces.
- Good design can often eliminate the need for complex model searches at analyses; though often control over the design is limited.
- If the models of interest are nested and without lots of parameters differentiating them, it's fairly uncontroversial to use nested likelihood ratio tests. (Example to follow.)
- My favorite approach is as follows. Given a coefficient that I'm interested in, I like to use covariate adjustment and multiple models to probe that effect to evaluate it for robustness and to see what other covariates knock it out. This isn't a terribly systematic approach, but it tends to teach you a lot about the data as you get your hands dirty.

# How to do nested model testing in R

```
fit1 <- lm(Fertility ~ Agriculture, data = swiss)
fit3 <- update(fit1, Fertility ~ Agriculture + Examination + Education)
fit5 <- update(fit3, Fertility ~ Agriculture + Examination + Education + Catholic + Infant.Mortality)
anova(fit1, fit3, fit5)
```

```
Analysis of Variance Table

Model 1: Fertility ~ Agriculture
Model 2: Fertility ~ Agriculture + Examination + Education
Model 3: Fertility ~ Agriculture + Examination + Education + Catholic +
Infant.Mortality
Res.Df RSS Df Sum of Sq    F Pr(>F)
1     45 6283
2     43 3181  2      3102 30.2 8.6e-09 ***
3     41 2105  2      1076 10.5 0.00021 ***
...
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# Generalized Linear Models

- Linear models are the most useful applied statistical technique. However, they are not without their limitations.
  - Additive response models don't make much sense if the response is discrete, or strictly positive.
  - Additive error models often don't make sense, for example if the outcome has to be positive.
  - Transformations are often hard to interpret.
  - There's value in modeling the data on the scale that it was collected.
  - Particularly interpretable transformations, natural logarithms in specific, aren't applicable for negative or zero values.
- Introduced in a 1972 RSSB paper by Nelder and Wedderburn.
- Involves three components
  - An *exponential family* model for the response.
  - A systematic component via a linear predictor.
  - A link function that connects the means of the response to the linear predictor.

## Example – Linear Model

- Assume that  $Y_i \sim N(\mu_i, \sigma^2)$  (the Gaussian distribution is an exponential family distribution.)
- Define the linear predictor to be  $\eta_i = \sum_{k=1}^p X_{ik} \beta_k$ .
- The link function as  $g$  so that  $g(\mu) = \eta$ .
  - For linear models  $g(\mu) = \mu$  so that  $\mu_i = \eta_i$ .
- This yields the same likelihood model as our additive error Gaussian linear model

$$Y_i = \sum_{k=1}^p X_{ik} \beta_k + \epsilon_i$$

where  $\epsilon_i \stackrel{\text{iid}}{\sim} N(0, \sigma^2)$

## Example – Logistic Regression

- Assume that  $Y_i \sim \text{Bernoulli}(\mu_i)$  so that  $E[Y_i] = \mu_i$  where  $0 \leq \mu_i \leq 1$ .
- Linear predictor  $\eta_i = \sum_{k=1}^p X_{ik} \beta_k$
- Link function  $g(\mu) = \eta = \log\left(\frac{\mu}{1-\mu}\right)$   $g$  is the (natural) log odds, referred to as the **logit**.
- Note then we can invert the logit function as

$$\mu_i = \frac{\exp(\eta_i)}{1 + \exp(\eta_i)} \quad \text{and} \quad 1 - \mu_i = \frac{1}{1 + \exp(\eta_i)}$$

Thus the likelihood is

$$\prod_{i=1}^n \mu_i^{y_i} (1 - \mu_i)^{1-y_i} = \exp\left(\sum_{i=1}^n y_i \eta_i\right) \prod_{i=1}^n (1 + \eta_i)^{-1}$$

# Example – Poisson Regression

- Assume that  $Y_i \sim \text{Poisson}(\mu_i)$  so that  $E[Y_i] = \mu_i$  where  $0 \leq \mu_i$
- Linear predictor  $\eta_i = \sum_{k=1}^p X_{ik} \beta_k$
- Link function  $g(\mu) = \eta = \log(\mu)$
- Recall that  $e^x$  is the inverse of  $\log(x)$  so that

$$\mu_i = e^{\eta_i}$$

Thus, the likelihood is

$$\prod_{i=1}^n (y_i!)^{-1} \mu_i^{y_i} e^{-\mu_i} \propto \exp\left(\sum_{i=1}^n y_i \eta_i - \sum_{i=1}^n \mu_i\right)$$

- In each case, the only way in which the likelihood depends on the data is through

$$\sum_{i=1}^n y_i \eta_i = \sum_{i=1}^n y_i \sum_{k=1}^p X_{ik} \beta_k = \sum_{k=1}^p \beta_k \sum_{i=1}^n X_{ik} y_i$$

Thus if we don't need the full data, only  $\sum_{i=1}^n X_{ik} y_i$ . This simplification is a consequence of choosing so-called 'canonical' link functions.

- (This has to be derived). All models achieve their maximum at the root of the so called normal equations

$$0 = \sum_{i=1}^n \frac{(Y_i - \mu_i)}{\text{Var}(Y_i)} W_i$$

where  $W_i$  are the derivative of the inverse of the link function.

## About Variances

$$0 = \sum_{i=1}^n \frac{(Y_i - \mu_i)}{\text{Var}(Y_i)} W_i$$

- For the linear model  $\text{Var}(Y_i) = \sigma^2$  is constant.
- For Bernoulli case  $\text{Var}(Y_i) = \mu_i(1 - \mu_i)$
- For the Poisson case  $\text{Var}(Y_i) = \mu_i$ .
- In the latter cases, it is often relevant to have a more flexible variance model, even if it doesn't correspond to an actual likelihood

$$0 = \sum_{i=1}^n \frac{(Y_i - \mu_i)}{\phi \mu_i(1 - \mu_i)} W_i \quad \text{and} \quad 0 = \sum_{i=1}^n \frac{(Y_i - \mu_i)}{\phi \mu_i} W_i$$

- These are called 'quasi-likelihood' normal equations

# Odds and Ends

- The normal equations have to be solved iteratively. Resulting in  $\hat{\beta}_k$  and, if included,  $\hat{\phi}$ .
- Predicted linear predictor responses can be obtained as  $\hat{\eta} = \sum_{k=1}^p X_k \hat{\beta}_k$
- Predicted mean responses as  $\hat{\mu} = g^{-1}(\hat{\eta})$
- Coefficients are interpreted as

$$g(E[Y|X_k = x_k, X_{\sim k} = x_{\sim k}]) - g(E[Y|X_k = x_k, X_{\sim k} = x_{\sim k}]) = \beta_k$$

or the change in the link function of the expected response per unit change in  $X_k$  holding other regressors constant.

- Variations on Newton/Raphson's algorithm are used to do it.
- Asymptotics are used for inference usually.
- Many of the ideas from linear models can be brought over to GLMs.

# Generalized Linear Models, Binary Data

- Frequently we care about outcomes that have two values
  - Alive/dead
  - Win/loss
  - Success/Failure
  - etc
- Called binary, Bernoulli or 0/1 outcomes
- Collection of exchangeable binary outcomes for the same covariate data are called binomial outcomes.

## Example – Baltimore Ravens win/loss

Ravens Data

```
download.file("https://dl.dropboxusercontent.com/u/7710864/data/ravensData.rda"
             , destfile="./data/ravensData.rda", method="curl")
load("./data/ravensData.rda")
head(ravensData)
```

	ravenWinNum	ravenWin	ravenScore	opponentScore
1	1	W	24	9
2	1	W	38	35
3	1	W	28	13
4	1	W	34	31
5	1	W	44	13
6	0	L	23	24

## Linear Regression

$$RW_i = b_0 + b_1 RS_i + e_i$$

$RW_i$  - 1 if a Ravens win, 0 if not

$RS_i$  - Number of points Ravens scored

$b_0$  - probability of a Ravens win if they score 0 points

$b_1$  - increase in probability of a Ravens win for each additional point

$e_i$  - residual variation due

```
lmRavens <- lm(ravensData$ravenWinNum ~ ravensData$ravenScore)
summary(lmRavens)$coef
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	0.2850	0.256643	1.111	0.28135
ravensData\$ravenScore	0.0159	0.009059	1.755	0.09625

## Odds

Binary Outcome 0/1

$$RW_i$$

Probability (0,1)

$$\Pr(RW_i | RS_i, b_0, b_1)$$

Odds (0,  $\infty$ )

$$\frac{\Pr(RW_i | RS_i, b_0, b_1)}{1 - \Pr(RW_i | RS_i, b_0, b_1)}$$

Log odds ( $-\infty, \infty$ )

$$\log\left(\frac{\Pr(RW_i | RS_i, b_0, b_1)}{1 - \Pr(RW_i | RS_i, b_0, b_1)}\right)$$

# Linear vs. Logistic Regression

Linear

$$RW_i = b_0 + b_1 RS_i + e_i$$

or

$$E[RW_i | RS_i, b_0, b_1] = b_0 + b_1 RS_i$$

Logistic

$$\Pr(RW_i | RS_i, b_0, b_1) = \frac{\exp(b_0 + b_1 RS_i)}{1 + \exp(b_0 + b_1 RS_i)}$$

or

$$\log\left(\frac{\Pr(RW_i | RS_i, b_0, b_1)}{1 - \Pr(RW_i | RS_i, b_0, b_1)}\right) = b_0 + b_1 RS_i$$

## Interpreting Logistic Regression

$$\log\left(\frac{\Pr(RW_i | RS_i, b_0, b_1)}{1 - \Pr(RW_i | RS_i, b_0, b_1)}\right) = b_0 + b_1 RS_i$$

$b_0$  - Log odds of a Ravens win if they score zero points

$b_1$  - Log odds ratio of win probability for each point scored (compared to zero points)

$\exp(b_1)$  - Odds ratio of win probability for each point scored (compared to zero points)

## Odds

- Imagine that you are playing a game where you flip a coin with success probability  $p$ .
- If it comes up heads, you win X. If it comes up tails, you lose Y.
- What should we set X and Y for the game to be fair?

$$E[\text{earnings}] = Xp - Y(1 - p) = 0$$

- Implies

$$\frac{Y}{X} = \frac{p}{1-p}$$

- The odds can be said as "How much should you be willing to pay for a  $p$  probability of winning a dollar?"
  - (If  $p > 0.5$  you have to pay more if you lose than you get if you win.)
  - (If  $p < 0.5$  you have to pay less if you lose than you get if you win.)

## Visualizing Fitting Logistic Reg. Curves

```
x <- seq(-10, 10, length = 1000)
manipulate(
  plot(x, exp(beta0 + beta1 * x) / (1 + exp(beta0 + beta1 * x)),
       type = "l", lwd = 3, frame = FALSE),
  beta1 = slider(-2, 2, step = .1, initial = 2),
  beta0 = slider(-2, 2, step = .1, initial = 0)
)
```

# Ravens Logistic Regression

```
logRegRavens <- glm(ravensData$ravenWinNum ~ ravensData$ravenScore,family="binomial")
summary(logRegRavens)
```

```
Call:
glm(formula = ravensData$ravenWinNum ~ ravensData$ravenScore,
     family = "binomial")

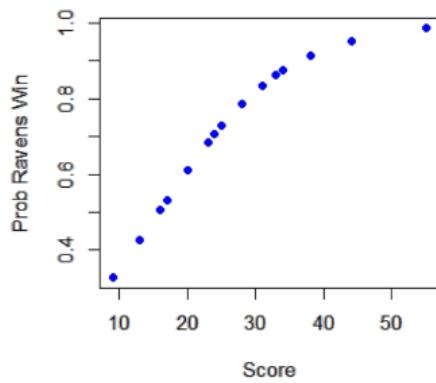
Deviance Residuals:
    Min      1Q  Median      3Q      Max 
-1.758 -1.100  0.530  0.806  1.495 

Coefficients:
            Estimate Std. Error z value Pr(>|z|)    
(Intercept) -1.6800    1.5541  -1.08    0.28    
ravensData$ravenScore 0.1066    0.0667   1.60    0.11    
(Dispersion parameter for binomial family taken to be 1)

Null deviance: 24.435 on 19 degrees of freedom
Residual deviance: 20.895 on 18 degrees of freedom
AIC: 24.89
```

# Ravens Fitted Values

```
plot(ravensData$ravenScore,logRegRavens$fitted,pch=19,col="blue",xlab="Score",ylab="Prob Ravens Win")
```



# Odds Ratios and Confidence Intervals

```
exp(logRegRavens$coeff)
```

```
(Intercept) ravensData$ravenScore  
0.1864      1.1125
```

```
exp(confint(logRegRavens))
```

```
2.5 % 97.5 %  
(Intercept) 0.005675 3.106  
ravensData$ravenScore 0.996230 1.303
```

## ANOVA for logistic regression

```
anova(logRegRavens, test="Chisq")
```

```
Analysis of Deviance Table  
  
Model: binomial, link: logit  
  
Response: ravensData$ravenWinNum  
  
Terms added sequentially (first to last)  
  
Df Deviance Resid. Df Resid. Dev Pr(>Chi)  
NULL          19      24.4  
ravensData$ravenScore 1     3.54      18      20.9    0.06 .  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## Interpreting Odds Ratios

- Not probabilities
- Odds ratio of 1 = no difference in odds
- Log odds ratio of 0 = no difference in odds
- Odds ratio < 0.5 or > 2 commonly a "moderate effect"
- Relative risk  $\frac{\Pr(RW_i|RS_i=10)}{\Pr(RW_i|RS_i=0)}$  often easier to interpret, harder to estimate
- For small probabilities RR  $\approx$  OR but **they are not the same!**

### [Wikipedia on Odds Ratio](#)

- [Wikipedia on Logistic Regression](#)
- [Logistic regression and glms in R](#)
- Brian Caffo's lecture notes on: [Simpson's paradox, Case-control studies](#)
- [Open Intro Chapter on Logistic Regression](#)

# Poisson GLNs

- Many data take the form of counts
  - Calls to a call center
  - Number of flu cases in an area
  - Number of cars that cross a bridge
- Data may also be in the form of rates
  - Percent of children passing a test
  - Percent of hits to a website from a country
- Linear regression with transformation is an option

# Poisson Distribution

- The Poisson distribution is a useful model for counts and rates
- Here a rate is count per some monitoring time
- Some examples uses of the Poisson distribution
  - Modeling web traffic hits
  - Incidence rates
  - Approximating binomial probabilities with small  $p$  and large  $n$
  - Analyzing contingency table data

# Poisson Mass Function

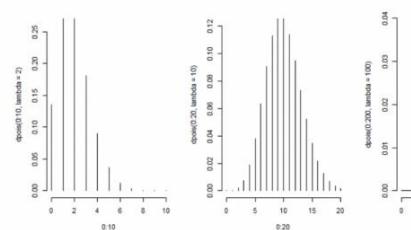
- $X \sim Poisson(t\lambda)$  if

$$P(X = x) = \frac{(t\lambda)^x e^{-t\lambda}}{x!}$$

For  $x = 0, 1, \dots$

- The mean of the Poisson is  $E[X] = t\lambda$ , thus  $E[X/t] = \lambda$
- The variance of the Poisson is  $Var(X) = t\lambda$ .
- The Poisson tends to a normal as  $t\lambda$  gets large.

```
par(mfrow = c(1, 3))
plot(0 : 10, dpois(0 : 10, lambda = 2), type = "h", frame = FALSE)
plot(0 : 20, dpois(0 : 20, lambda = 10), type = "h", frame = FALSE)
plot(0 : 200, dpois(0 : 200, lambda = 100), type = "h", frame = FALSE)
```



Sort of, showing that the mean and variance are equal

```
x <- 0 : 10000; lambda = 3
mu <- sum(x * dpois(x, lambda = lambda))
sigmasq <- sum((x - mu)^2 * dpois(x, lambda = lambda))
c(mu, sigmasq)
```

```
[1] 3 3
```

# Example: Leek Group Website Traffic

- Consider the daily counts to Jeff Leek's web site

<http://biostat.jhsph.edu/~leek/>

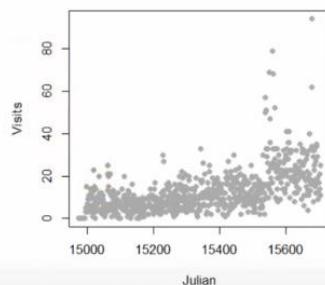
- Since the unit of time is always one day, set  $t = 1$  and then the Pois: web hits per day. (If we set  $t = 24$ , it would be web hits per hour).

```
download.file("https://dl.dropboxusercontent.com/u/7710864/data/gaData.rda", destfile=".~/data/gaData.rda", method="curl")
load("./data/gaData.rda")
gaData$Julian <- julian(gaData$date)
head(gaData)
```

	date	visits	simplystats	julian
1	2011-01-01	0	0	14975
2	2011-01-02	0	0	14976
3	2011-01-03	0	0	14977
4	2011-01-04	0	0	14978
5	2011-01-05	0	0	14979
6	2011-01-06	0	0	14980

<http://skardhamar.github.com/rga/>

```
plot(gaData$Julian, gaData$visits, pch=19, col="darkgrey", xlab="Julian", ylab="Visits")
```



## Linear Regression

$$NH_i = b_0 + b_1 JD_i + e_i$$

$NH_i$  - number of hits to the website

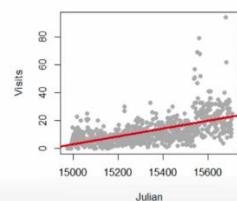
$JD_i$  - day of the year (Julian day)

$b_0$  - number of hits on Julian day 0 (1970-01-01)

$b_1$  - increase in number of hits per unit day

$e_i$  - variation due to everything we didn't measure

```
plot(gaData$Julian, gaData$visits, pch=19, col="darkgrey", xlab="Julian", ylab="Visits")
lm1 <- lm(gaData$visits ~ gaData$Julian)
abline(lm1, col="red", lwd=3)
```



# Taking Log of Outcome

- Taking the natural log of the outcome has a specific interpretation.
- Consider the model

$$\log(NH_i) = b_0 + b_1 JD_i + e_i$$

$NH_i$  - number of hits to the website

$JD_i$  - day of the year (Julian day)

$b_0$  - log number of hits on Julian day 0 (1970-01-01)

$b_1$  - increase in log number of hits per unit day

$e_i$  - variation due to everything we didn't measure

## Exponentiating Coefficients

- $e^{E[\log(Y)]}$  geometric mean of  $Y$ .
  - With no covariates, this is estimated by  $e^{\frac{1}{n} \sum_{i=1}^n \log(y_i)} = (\prod_{i=1}^n y_i)^{1/n}$
- When you take the natural log of outcomes and fit a regression model, your exponentiated coefficients estimate things about geometric means.
- $e^{\beta_0}$  estimated geometric mean hits on day 0
- $e^{\beta_1}$  estimated relative increase or decrease in geometric mean hits per day
- There's a problem with logs when you have zero counts, adding a constant

```
round(exp(coef(lm(I(log(gaData$visits + 1)) ~ gaData$julian))), 5)
```

```
(Intercept) gaData$julian  
0.000 1.002
```

## Linear vs. Poisson Regression

Linear

$$NH_i = b_0 + b_1 JD_i + e_i$$

or

$$E[NH_i | JD_i, b_0, b_1] = b_0 + b_1 JD_i$$

Poisson/log-linear

$$\log(E[NH_i | JD_i, b_0, b_1]) = b_0 + b_1 JD_i$$

or

$$E[NH_i | JD_i, b_0, b_1] = \exp(b_0 + b_1 JD_i)$$

# Multiplicative Differences

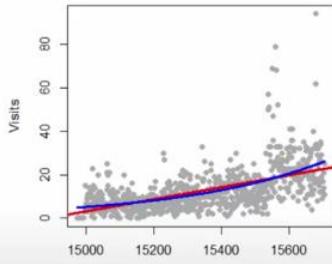
$$E[NH_i | JD_i, b_0, b_1] = \exp(b_0 + b_1 JD_i)$$

$$E[NH_i | JD_i, b_0, b_1] = \exp(b_0) \exp(b_1 JD_i)$$

If  $JD_i$  is increased by one unit,  $E[NH_i | JD_i, b_0, b_1]$  is multiplied by  $\exp(b_1)$

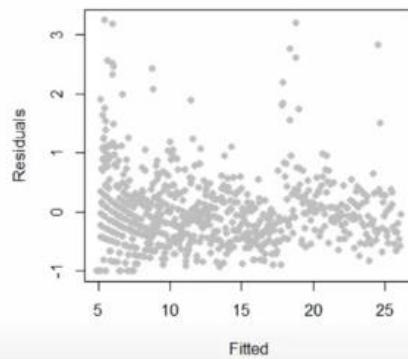
## Poisson Regression

```
plot(gaData$julian,gaData$visits,pch=19,col="darkgrey",xlab="Julian",ylab="Visits  
glm1 <- glm(gaData$visits ~ gaData$julian,family="poisson")  
abline(lm1,col="red",lwd=3); lines(gaData$julian,glm1$fitted,col="blue",lwd=3)
```



## Mean-variance Relationship

```
plot(glm1$fitted,glm1$residuals,pch=19,col="grey",ylab="Residuals",xlab="Fitted")
```



# Model Agnostic Standard Error

```
library(sandwich)
confint.agnostic <- function (object, parm, level = 0.95, ...)
{
  cf <- coef(object); pnames <- names(cf)
  if (missing(parm))
    parm <- pnames
  else if (is.numeric(parm))
    parm <- pnames[parm]
  a <- (1 - level)/2; a <- c(a, 1 - a)
  pct <- stats::format.perc(a, 3)
  fac <- qnorm(a)
  ci <- array(NA, dim = c(length(parm), 2L), dimnames = list(parm,
    pct))
  ses <- sqrt(diag(sandwich::vcovHC(object)))[parm]
  ci[] <- cf[parm] + ses %*% fac
  ci
}
```

<http://stackoverflow.com/questions/3817182/vcovhc-and-confidence-interval>

## Estimating Confidence Intervals

```
confint(glm1)
```

	2.5 %	97.5 %
(Intercept)	-34.34658	-31.159716
gaData\$julian	0.00219	0.002396

```
confint.agnostic(glm1)
```

	2.5 %	97.5 %
(Intercept)	-36.362675	-29.136997
gaData\$julian	0.002058	0.002528

## Rates and Proportions

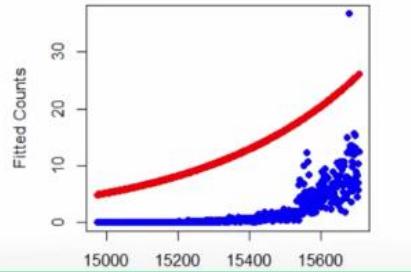
$$E[NHSS_i | JD_i, b_0, b_1] / NH_i = \exp(b_0 + b_1 JD_i)$$

$$\log(E[NHSS_i | JD_i, b_0, b_1]) - \log(NH_i) = b_0 + b_1 JD_i$$

$$\log(E[NHSS_i | JD_i, b_0, b_1]) = \log(NH_i) + b_0 + b_1 JD_i$$

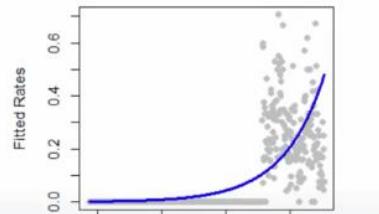
# Fitting Rates in R

```
glm2 <- glm(gaData$simplystats ~ julian(gaData$date), offset=log(visits+1),
            family="poisson", data=gaData)
plot(julian(gaData$date), glm2$fitted, col="blue", pch=19, xlab="Date", ylab="Fitted Counts")
points(julian(gaData$date), glm1$fitted, col="red", pch=19)
```



# Fitted Model

```
glm2 <- glm(gaData$simplystats ~ julian(gaData$date), offset=log(visits+1),
            family="poisson", data=gaData)
plot(julian(gaData$date), gaData$simplystats/(gaData$visits+1), col="grey", xlab="Date",
      ylab="Fitted Rates", pch=19)
lines(julian(gaData$date), glm2$fitted/(gaData$visits+1), col="blue", lwd=3)
```



# More Info

- [Log-linear models and multiway tables](#)
- [Wikipedia on Poisson regression](#), [Wikipedia on overdispersion](#)
- [Regression models for count data in R](#)
- [pscl package](#) - the function `zeroinfl` fits zero inflated models.

# Hodgepodge – Regression Models

## How to Fit Function using Linear Models

- Consider a model  $Y_i = f(X_i) + \epsilon_i$ .
- How can we fit such a model using linear models (called scatterplot smoothing)
- Consider the model

$$Y_i = \beta_0 + \beta_1 X_i + \sum_{k=1}^d (x_i - \xi_k)_+ Y_k + \epsilon_i$$

where  $(a)_+ = a$  if  $a > 0$  and 0 otherwise and  $\xi_1 \le \dots \le \xi_d$  are known knot points.

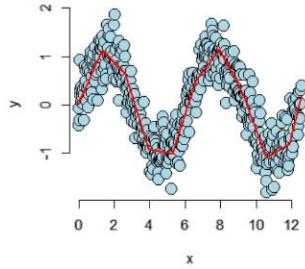
- Prove to yourself that the mean function

$$\beta_0 + \beta_1 X_i + \sum_{k=1}^d (x_i - \xi_k)_+ Y_k$$

is continuous at the knot points.

## Simulated Example

```
n <- 500; x <- seq(0, 4 * pi, length = n); y <- sin(x) + rnorm(n, sd = .3)
knots <- seq(0, 8 * pi, length = 20);
splineTerms <- sapply(knots, function(knot) (x > knot) * (x - knot))
xMat <- cbind(1, x, splineTerms)
yhat <- predict(lm(y ~ xMat - 1))
plot(x, y, frame = FALSE, pch = 21, bg = "lightblue", cex = 2)
lines(x, yhat, col = "red", lwd = 2)
```

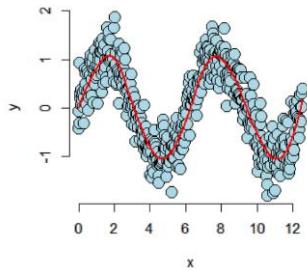


# Adding Squared Terms

- Adding squared terms makes it continuously differentiable at the knot points.
- Adding cubic terms makes it twice continuously differentiable at the knot points; etcetera.

$$Y_i = \beta_0 + \beta_1 X_i + \beta_2 X_i^2 + \sum_{k=1}^d (x_i - \xi_k)^2 \gamma_k + \epsilon_i$$

```
splineTerms <- sapply(knots, function(knot) (x > knot) * (x - knot)^2)
xMat <- cbind(1, x, x^2, splineTerms)
yhat <- predict(lm(y ~ xMat - 1))
plot(x, y, frame = FALSE, pch = 21, bg = "lightblue", cex = 2)
lines(x, yhat, col = "red", lwd = 2)
```

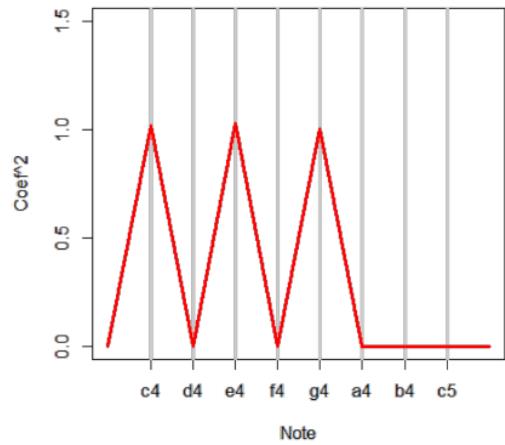


## Notes

- The collection of regressors is called a basis.
  - People have spent a lot of time thinking about bases for this kind of problem. So, consider this as just a teaser.
- Single knot point terms can fit hockey stick like processes.
- These bases can be used in GLMs as well.
- An issue with these approaches is the large number of parameters introduced.
  - Requires some method of so called regularization.

## Harmonics using Linear Models

```
##Chord finder, playing the white keys on a piano from octave c4 - c5
notes4 <- c(261.63, 293.66, 329.63, 349.23, 392.00, 440.00, 493.88, 523.25)
t <- seq(0, 2, by = .001); n <- length(t)
c4 <- sin(2 * pi * notes4[1] * t); e4 <- sin(2 * pi * notes4[3] * t);
g4 <- sin(2 * pi * notes4[5] * t)
chord <- c4 + e4 + g4 + rnorm(n, 0, 0.3)
x <- sapply(notes4, function(freq) sin(2 * pi * freq * t))
fit <- lm(chord ~ x - 1)
```



```
##(How you would really do it)
a <- fft(chord); plot(Re(a)^2, type = "l")
```

