

# **BMEG 802 – Advanced Biomedical Experimental Design and Analysis**

## Regression

---

Joshua G. A. Cashaback, PhD

# Recap

- Effect Size
- Power
  - Parametric
  - Numerical

# Learning Objectives

- Regression
  - Bivariate
    - Linear (Derivation)
    - Nonlinear
  - Multiple Regression
- Correlation
  - Pearson's  $r$
  - Spearman's  $\rho$

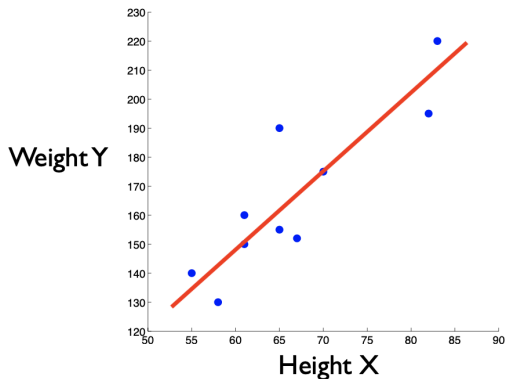
# BIVARIATE LINEAR REGRESSION

# Regression - Bivariate

$$\hat{y}_i = B_0 + B_1 \cdot x_i$$

- want to predict  $\hat{y}_i$  (e.g., height) based on  $x_i$  (e.g., weight).
  - equation of a line
    - $B_1$  (slope),  $B_0$  (intercept)
  - X,Y continuous
  - relationship between X and Y

# Regression - Bivariate



Height (X)	Weight (Y)
55	140
61	150
67	152
83	220
65	190
82	195
70	175
58	130
65	155
61	160

Line of best fit:  $B_0 = -7.2$ ,  $B_1 = 2.6$

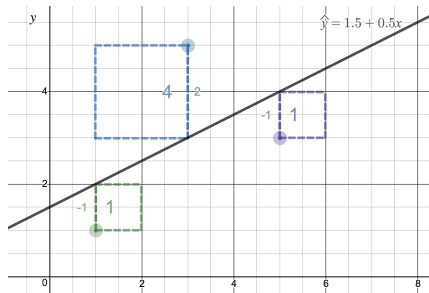
- least squares
- what do squares have to do with this???

# Bivariate Regression - Least Squares

Optimization problem:

$$\min(y_i - \hat{y}_i)^2$$

where  $\hat{y}_i$  and  $y_i$  are respectively the predicted and actual  $y$  values.



We find  $B_1$  (slope),  $B_0$  (intercept) that minimize the squared differences!

# Least Squares Derivation

remember,  $\hat{y}_i = B_0 + B_1 \cdot x_i$  and we want to  $\min(y_i - \hat{y}_i)^2$

$$(y_i - \hat{y}_i)^2 = (y_i - (B_0 + B_1 \cdot x_i))^2$$

$$(y_i - \hat{y}_i)^2 = (y_i - B_0 - B_1 \cdot x_i)^2$$

In particular, we minimum the sum of squares

$$\sum_{i=1}^n (y_i - \hat{y}_i)^2 = \sum_{i=1}^n (y_i - B_0 - B_1 \cdot x_i)^2$$

$$SS = \sum_{i=1}^n (y_i - B_0 - B_1 \cdot x_i)^2$$

$$\frac{\partial SS}{\partial B_0} = 0$$

$$\frac{\partial SS}{\partial B_1} = 0$$

Results in two equations with two unknowns.



# Least Squares Derivation - Intercept

$$\frac{\partial}{\partial B_0} \sum_{i=1}^n (y_i - B_0 - B_1 \cdot x_i)^2 = 0$$

We can move the sum outside:

$$\sum_{i=1}^n \frac{\partial}{\partial B_0} (y_i - B_0 - B_1 \cdot x_i)^2 = 0$$

Taking the derivative (note the chain rule):

$$\sum_{i=1}^n 2(y_i - B_0 - B_1 \cdot x_i)(-1) = 0$$

Rearranging,

$$-2 \sum_{i=1}^n (y_i - B_0 - B_1 \cdot x_i) = 0$$

# Least Squares Derivation - Slope

$$\frac{\partial}{\partial B_1} \sum_{i=1}^n (y_i - B_0 - B_1 \cdot x_i)^2 = 0$$

$$\sum_{i=1}^n \frac{\partial}{\partial B_0} (y_i - B_0 - B_1 \cdot x_i)^2 = 0$$

$$\sum_{i=1}^n 2(y_i - B_0 - B_1 \cdot x_i)(-x_i) = 0$$

$$-2 \sum_{i=1}^n (y_i - B_0 - B_1 \cdot x_i)(x_i) = 0$$

Two equations and two unknowns

# Least Squares Derivation - Tip Interlude

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

Thus,

$$\sum_{i=1}^n x_i = n\bar{x}$$

Similarly,

$$\sum_{i=1}^n y_i = n\bar{y}$$

When summing a constant you can multiply by n. For example,

$$\sum_{i=1}^n \bar{x} = n\bar{x}$$

# Least Squares Derivation - Tip Interlude Cont'd

$$\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) = \sum_{i=1}^n x_i \cdot y_i - \sum_{i=1}^n x_i \cdot \bar{y} - \sum_{i=1}^n y_i \cdot \bar{x} + \sum_{i=1}^n \bar{x} \cdot \bar{y} \quad (1)$$

$$= \sum_{i=1}^n x_i \cdot y_i - \sum_{i=1}^n x_i \cdot \bar{y} - \sum_{i=1}^n y_i \cdot \bar{x} + \sum_{i=1}^n \bar{x} \cdot \bar{y} \quad (2)$$

$$= \sum_{i=1}^n x_i \cdot y_i - n \cdot \bar{x} \cdot \bar{y} - n \cdot \bar{x} \cdot \bar{y} + n \cdot \bar{x} \cdot \bar{y} \quad (3)$$

$$= \sum_{i=1}^n x_i \cdot y_i - n \cdot \bar{x} \cdot \bar{y} \quad (4)$$

# Least Squares Derivation - Tip Interlude Cont'd

$$\sum_{i=1}^n (x_i - \bar{x})^2 = \sum_{i=1}^n (x_i^2 - 2 \cdot x_i \cdot \bar{x} + \bar{x}^2) \quad (5)$$

$$= \sum_{i=1}^n x_i^2 - 2 \cdot \bar{x} \sum_{i=1}^n x_i + \sum_{i=1}^n \bar{x}^2 \quad (6)$$

$$= \sum_{i=1}^n x_i^2 - 2 \cdot \bar{x} \cdot n \cdot \bar{x} + n\bar{x}^2 \quad (7)$$

$$= \sum_{i=1}^n x_i^2 - n\bar{x}^2 \quad (8)$$

# Least Squares Derivation - Intercept

Let's do some algebra on our previous equation:

$$-2 \sum_{i=1}^n (y_i - B_0 - B_1 \cdot x_i) = 0$$

Divide both sides by -2,

$$\sum_{i=1}^n (y_i - B_0 - B_1 \cdot x_i) = 0$$

$$\sum_{i=1}^n y_i - \sum_{i=1}^n B_0 - B_1 \cdot \sum_{i=1}^n x_i = 0$$

$$n \cdot \bar{y} - n \cdot B_0 - B_1 \cdot n \cdot \bar{x} = 0$$

$$\bar{y} - B_0 - B_1 \cdot \bar{x} = 0$$

$$B_0 = \bar{y} - B_1 \cdot \bar{x}$$

# Least Squares Derivation - Slope

$$-2 \sum_{i=1}^n (y_i - B_0 - B_1 \cdot x_i)(x_i) = 0$$

$$\sum_{i=1}^n (x_i \cdot y_i - B_0 \cdot x_i - B_1 \cdot x_i^2) = 0$$

$$\sum_{i=1}^n x_i \cdot y_i - B_0 \sum_{i=1}^n x_i - B_1 \sum_{i=1}^n x_i^2 = 0$$

$$\sum_{i=1}^n x_i \cdot y_i - B_0 \sum_{i=1}^n x_i - B_1 \sum_{i=1}^n x_i^2 = 0$$

Substitute in  $B_0$

$$\sum_{i=1}^n x_i \cdot y_i - (\bar{y} - B_1 \cdot \bar{x})n \cdot \bar{x} - B_1 \sum_{i=1}^n x_i^2 = 0$$

# Least Squares Derivation - Slope Cont'd

$$\sum_{i=1}^n x_i \cdot y_i - (\bar{y} - B_1 \cdot \bar{x})n \cdot \bar{x} - B_1 \sum_{i=1}^n x_i^2 = 0$$

$$\sum_{i=1}^n x_i \cdot y_i - n \cdot \bar{x} \cdot \bar{y} + n \cdot B_1 \cdot \bar{x}^2 - B_1 \sum_{i=1}^n x_i^2 = 0$$

$$\sum_{i=1}^n x_i \cdot y_i - n \cdot \bar{x} \cdot \bar{y} = B_1 \sum_{i=1}^n x_i^2 - n \cdot B_1 \cdot \bar{x}^2$$

$$\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) = B_1 \sum_{i=1}^n (x_i - \bar{x})^2$$

$$B_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$



# Derivation Summary

$$B_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

$$B_0 = \bar{y} - B_1 \cdot \bar{x}$$

# Calculate B's

Height weight example.

```
weight = c(55,61,67,83,65,82,70,58,65,61)
height = c(140,150,152,220,190,195,175,130,155,160)
B1 = sum((weight - mean(weight)) * (height - mean(height))) / sum((weight - mean(weight))^2)
B0 = mean(height) - B1 * mean(weight)
B0
```

```
## [1] -7.17693
```

```
B1
```

```
## [1] 2.606851
```

# Bivariate Prediction

We can now predict height based on weight

```
weight_bob = 80  
height_bob = B0 + B1 * weight_bob  
height_bob
```

```
## [1] 201.3711
```

- Assumes residuals are normally distributed (plot residuals  $(y_i - \hat{y}_i)$ 
  - flat line with equal noise throughout range of x-values
- Cannot extrapolate
- Least squares sensitive to outliers (robust = absolute deviations)
- Make sure to plot data to see if linear relationship holds

# Bivariate - Standard Error of Estimate

$$SE = \sqrt{\frac{\sum_i^n (y_i - \hat{y})^2}{N - 2}}$$

- Quality of the fit
- $N$  = number of pairs of data
  - $N-2$  = degrees of freedom (lose two because we have two parameters,  $B_1, B_0$ )

```
SE = sqrt(sum((height - (B0 + B1 * weight))^2) / (length(weight) - 2))  
SE
```

```
## [1] 14.11408
```

# Correlation Coefficient (r)

Pearson correlation coefficient (r) is the covariance of the two variables divided by the product of their standard deviations.

- metric for goodness of fit

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}$$

- range [-1,1]

- 0 = no relationship
- 1 = perfect positive correlation
- -1 = perfect negative correlation

# Correlation Coefficient (r)

```
r = sum((weight - mean(weight)) * (height - mean(height))) / (sqrt(sum((weight - mean(weight)) ** 2) * sum((height - mean(height)) ** 2)))  
r
```

```
## [1] 0.8786421
```

OR

```
cor(height, weight)
```

```
## [1] 0.8786421
```

# Coefficient of Determination ( $r$ )

$r^2$  (square correlation coefficient)

- amount of variance explained
- another way to measure quality of fit
- range  $[0,1]$ 
  - 0 = no relationship
  - 1 = perfect correlation (explains 100% of the relationship)

```
rsquared = cor(weight,height)^2  
rsquared
```

```
## [1] 0.772012
```

# Bivariate Regression - Significance Test

H0: (population)  $r = 0$ ;

H1: (population)  $r$  not equal to 0 (two-tailed)

H1: (population)  $r < 0$  (or  $> 0$ ) : one-tailed

## Sampling distribution of $r$

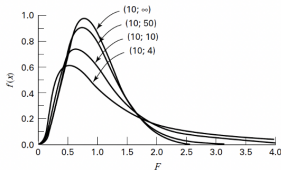
- IF we were to randomly draw two samples from two populations that were not correlated at all, what proportion of the time would we get a value of  $r$  as extreme as we observe?
- if  $p < .05$  we reject H0

NOTE: Correlation DOES NOT equal Causation

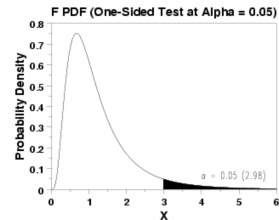


# Bivariate Regression - Significance Test Cont'd

F-Distribution (Fisher-Snedecor):



**FIGURE 6.10.1** The F distribution for various degrees of freedom.  
(From *Documenta Geigy, Scientific Tables*, Seventh Edition, 1970. Courtesy of Ciba-Geigy Limited, Basel, Switzerland.)



- Compare if regression explains significantly more data with  $B_1$
- $p(df_{k-1}, df_{N-k})$  - probability of F-statistic
  - $k$  = number of parameters (e.g.,  $k = 2$  since  $B_1, B_0$ )
  - $N$  = number of pairs

# Bivariate Regression - Significance Test Cont'd

F-statistic for Bivariate regression:

$$F = \frac{r^2(N-2)}{1-r^2}$$

```
N = length(weight)
Fstat0 <- (r*r*(N-2)) / (1-(r*r))
# Area under F-Distribution, using pf() function
pval0 <- 1 - pf(Fstat0, 1, N-2) # Fstat, parameters (k - 1), df (N - k)
Fstat0

## [1] 27.08957

pval0

## [1] 0.0008176335
```

# Bivariate Regression in R

## Linear

```
m1 <- lm(weight ~ height) # Linear model function!
```

```
summary(m1)
```

```
##
## Call:
## lm(formula = weight ~ height)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -8.6002 -2.9754  0.6787  2.0677  6.9190
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  17.3322     9.6037   1.805 0.108760
## height        0.2962     0.0569   5.205 0.000818 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.757 on 8 degrees of freedom
## Multiple R-squared:  0.772, Adjusted R-squared:  0.7435
## F-statistic: 27.09 on 1 and 8 DF, p-value: 0.0008176
```

# Bivariate Regression in R

```
summary(m1)$coefficients[1] #  $B_0$ , coefficient
```

```
## [1] 17.33223
```

```
summary(m1)$coefficients[2] #  $B_1$ , coefficient
```

```
## [1] 0.2961474
```

```
summary(m1)$coefficients[2,4] #  $p$ -value
```

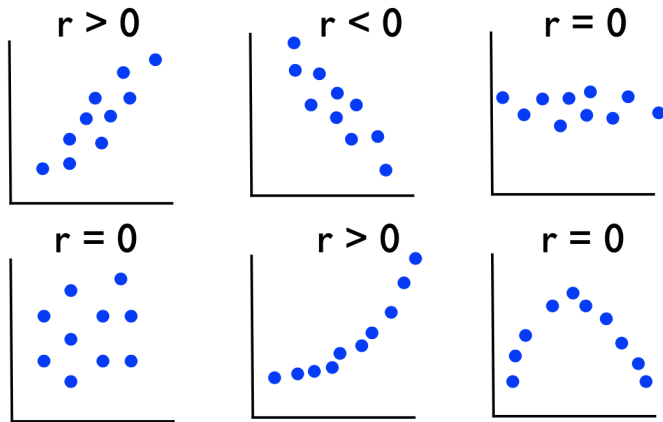
```
## [1] 0.0008176335
```

```
summary(m1)$r.squared #  $r^2$  of model
```

```
## [1] 0.772012
```

# BIVARIATE NONLINEAR REGRESSION

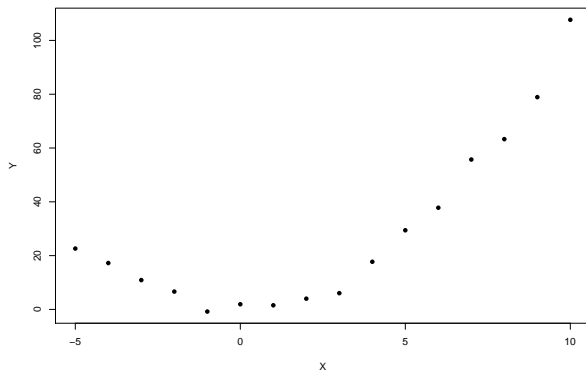
# Regression - Nonlinear



remember:  $r$  measures **linear** correlation

# Regression - Nonlinear

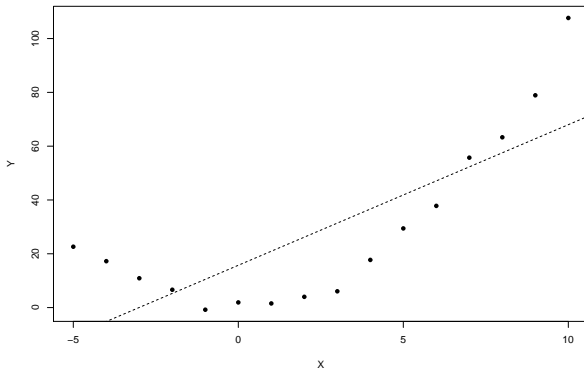
```
X <- c(-5,-4,-3,-2,-1,0,1,2,3,4,5,6,7,8,9,10)
Y <- X^2 + rnorm(16,0,3)
plot(X,Y,pch=16)
```



# Regression - Nonlinear

Linear regression would produce:

```
m2 <- lm(Y ~ X)
plot(X,Y,pch=16)
abline(m2, lty=2)
```





# Regression - Nonlinear

We can include nonlinear terms:  $\hat{y}_i = B_0 + B_1 \cdot x_i^2$

```
Xsquared <- X*X # make a nonlinear variable in R
m0 <- lm(Y ~ Xsquared)
summary(m0)
```

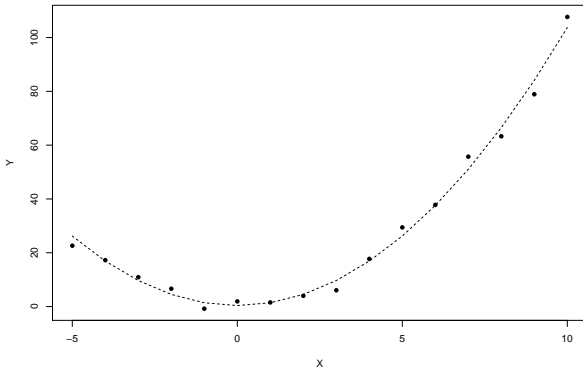
```
##
## Call:
## lm(formula = Y ~ Xsquared)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.1724 -2.4521  0.2852  1.6950  4.7109
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.36900     1.01829   0.362   0.722
## Xsquared       1.03352     0.02511  41.159 5.22e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.994 on 14 degrees of freedom
## Multiple R-squared:  0.9918, Adjusted R-squared:  0.9912
## F-statistic: 1694 on 1 and 14 DF,  p-value: 5.225e-16
```

Note: In practice you typically assume linearity a priori (unless you have theoretical reason)

# Regression - Nonlinear

Plotting the result:

```
Xsquared <- X*X # make a nonlinear variable in R  
yfit <- predict(m0,data.frame(X=Xsquared)) # predicted y-values from linear model  
plot(X,Y,pch=16)  
lines(X,yfit,lty=2)
```



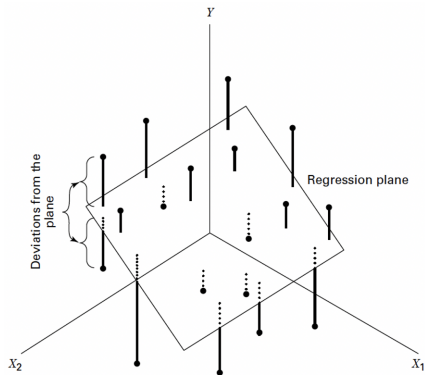
Note: In practice you typically assume linearity a priori (unless you have theoretical reason)

# MULTIPLE REGRESSION

# Multiple Regression

Same idea as bivariate, but just adding more terms

$$\hat{y}_i = B_0 + B_1 \cdot x_{i1} + B_2 \cdot x_{i2} + \dots + B_n \cdot x_{in}$$



# Multiple Regression

**TABLE 10.6.1 Bone Toughness and Collagen Network Properties for 29 Femurs**

W	P	S
193.6	6.24	30.1
137.5	8.03	22.2
145.4	11.62	25.7
117.0	7.68	28.9
105.4	10.72	27.3
99.9	9.28	33.4
74.0	6.23	26.4
74.4	8.67	17.2
112.8	6.91	15.9
125.4	7.51	12.2
126.5	10.01	30.0
115.9	8.70	24.0
98.8	5.87	22.6
94.3	7.96	18.2
99.9	12.27	11.5
83.3	7.33	23.9
72.8	11.17	11.2
83.5	6.03	15.6
59.0	7.90	10.6

W	P	S
87.2	8.27	24.7
84.4	11.05	25.6
78.1	7.61	18.4
51.9	6.21	13.5
57.1	7.24	12.2
54.7	8.11	14.8
78.6	10.05	8.9
53.7	8.79	14.9
96.0	10.40	10.3
89.0	11.72	15.4

W (Bone Toughness: Force required to fracture bone)

P (Porosity)

S (Tensile Strength)

# Multiple Regression

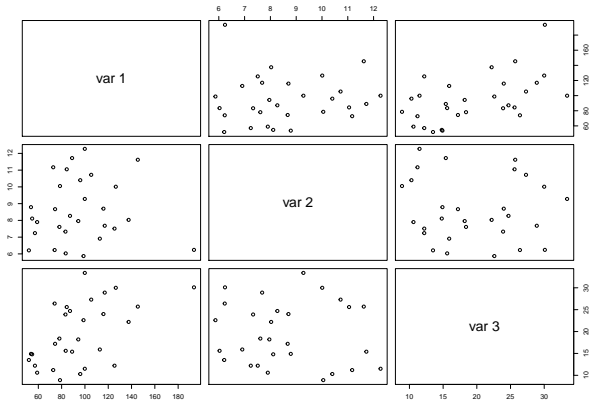
Inputting data (note, you can also have R read in files)

```
W = c(193.6, 137.5, 145.4, 117, 105.4, 99.9, 74, 74.4, 112.8, 125.4, 126.5)
P = c(6.24, 8.03, 11.62, 7.68, 10.72, 9.28, 6.23, 8.67, 6.91, 7.51, 10.01,
S = c(30.1, 22.2, 25.7, 28.9, 27.3, 33.4, 26.4, 17.2, 15.9, 12.2, 30., 24.
data0 <- array(c(W,P,S), dim = c(length(W), 3))
```

# Multiple Regression

Take a look at the data

```
pairs(data0)
```



# Multiple Regression

Look at individual correlations between variables

```
cor(data0)
```

```
##           [,1]           [,2]           [,3]
## [1,] 1.00000000  0.04331177  0.53547194
## [2,] 0.04331177  1.00000000 -0.08015793
## [3,] 0.53547194 -0.08015793  1.00000000
```



# Multiple Regression in R

```
m2 <- lm(W ~ S + P)
```

```
summary(m2)
```

```
##
## Call:
## lm(formula = W ~ S + P)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -33.907 -19.594  -0.517  10.159  76.813
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  35.6138    29.1296   1.223  0.23245
## S             2.3960     0.7301   3.282  0.00294 **
## P             1.4509     2.7632   0.525  0.60397
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 27.42 on 26 degrees of freedom
## Multiple R-squared:  0.2942, Adjusted R-squared:  0.2399
## F-statistic: 5.419 on 2 and 26 DF,  p-value: 0.01078
```

# Bivariate Regression in R

```
summary(m2)$coefficients[1] # intercept coefficient
```

```
## [1] 35.61383
```

```
summary(m2)$coefficients[2] # S coefficient
```

```
## [1] 2.395993
```

```
summary(m2)$coefficients[3] # P coefficient
```

```
## [1] 1.450938
```

```
summary(m2)$coefficients[2,4] # S p-value
```

```
## [1] 0.002940326
```

```
summary(m2)$coefficients[3,4] # P p-value
```

```
## [1] 0.6039738
```

```
summary(m2)$r.squared # r2 of model
```

```
## [1] 0.2942146
```

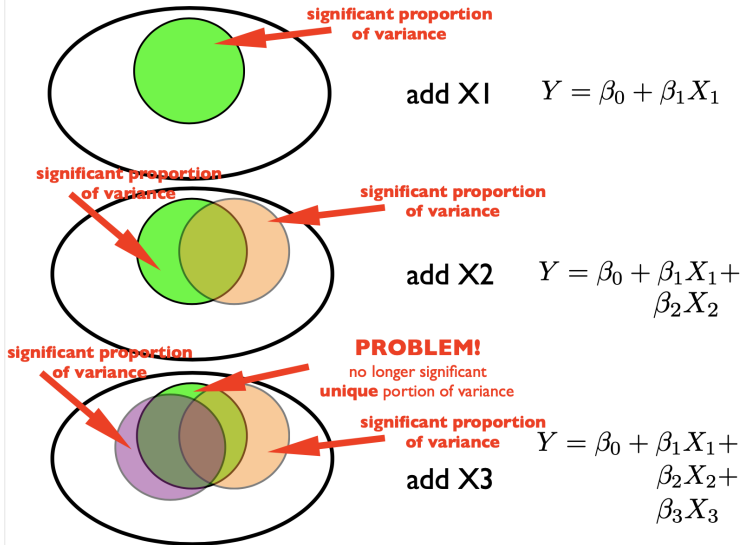
# Multiple Regression

- Overall, we find that this multiple regression is significant ( $p = 0.01078$ ).
- BUT, Bone Toughness is not significantly explained by Porosity ( $p = 0.604$ ).
- More terms in our model reduces residuals BUT WE MAY BE JUST FITTING NOISE!
  - We want to choose the model the best explains the data with the least amount of parameters: to prevent over-fitting
    - Model comparison (e.g., F-tests, AIC  
[https://en.wikipedia.org/wiki/Akaike\\_information\\_criterion](https://en.wikipedia.org/wiki/Akaike_information_criterion))

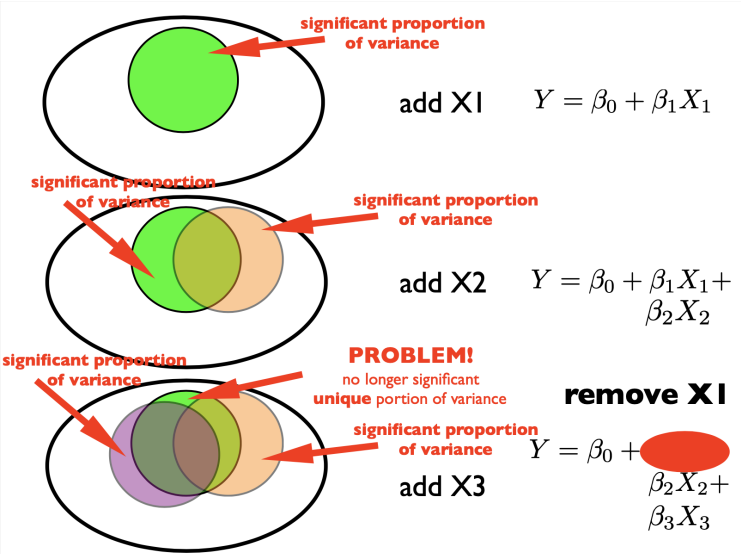
## Other considerations

- sometimes several independent variables explain the same data (covariance)
- What is the best procedure to determine which variables to use?
  - one solution: step-wise regression

# Multiple Regression - Covariates



# Multiple Regression - Covariates



# Stepwise - Multiple Regression

## STEPS:

1. start with no IVs in the equation
2. check to see if any IVs significantly predict the DV
3. if no, STOP. if yes, add best IV (largest  $r^2$ ) and go to step 4
4. check to see if any IVs add significantly to the equation
5. if no, stop. if yes, add best IV (largest  $r^2$ ), go to step 6
6. check each IV currently in the equation to make sure they contribute unique portions of variance
7. **remove** any that don't
8. go to step 4

# Stepwise - Multiple Regression

Define Minimal Model (no IVs) and Full Model (all IVs)

```
m_min <- lm(W ~ 1) # minimal model  
m_all <- lm(W ~ S + P) # define full model
```

# Stepwise - Multiple Regression

Run Step-wise regression (this is a big output)

```
# perform stepwise regression
```

```
mbest <- step(m_min, list(lower=m_min, upper=m_all), direction="both")
```

```
## Start:  AIC=201
```

```
## W ~ 1
```

```
##
```

```
##           Df Sum of Sq   RSS   AIC
```

```
## + S       1      7943.8 19761 193.20
```

```
## <none>                27705 201.00
```

```
## + P       1        52.0 27653 202.94
```

```
##
```

```
## Step:  AIC=193.2
```

```
## W ~ S
```



# Stepwise - Multiple Regression

The best model uses just tensile strength (S) to explain bone toughness (W).

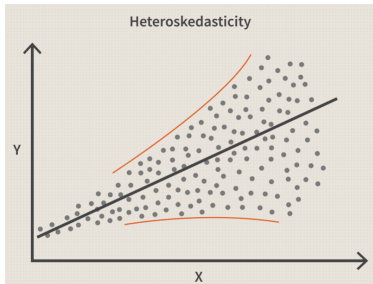
```
summary(mbest) # Summary of best model
```

```
##
## Call:
## lm(formula = W ~ S)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -37.143 -20.456  -2.391   10.433   73.705
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  48.7003     14.8762   3.274  0.00291 **
## S            2.3653       0.7179   3.295  0.00276 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 27.05 on 27 degrees of freedom
## Multiple R-squared:  0.2867, Adjusted R-squared:  0.2603
## F-statistic: 10.85 on 1 and 27 DF,  p-value: 0.002759
```

# BIVARIATE NONPARAMETRIC

# Nonparametric - Spearman's rank correlation coefficient

- violations of normality
  - e.g., heteroskedastic
- nonlinear (monotonic functions)
  - do not have to fiddle with different powers on IVs (squaring, cubing, etc) etc.
- less sensitive to outliers

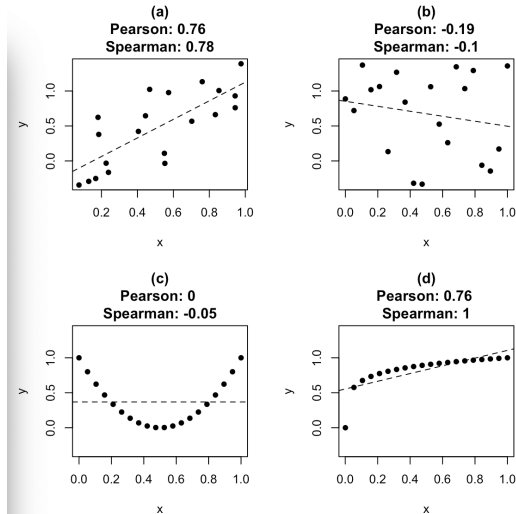


# Nonparametric - Spearman's

Spearman's rank correlation coefficient ( $\rho$ )

- It assesses how well the relationship between two variables can be described using a monotonic (increasing or decreasing) function
- rank order method
- range  $[-1,+1]$

# Nonparametric - Spearman's



Spearman does not work on (c) – non-monotonic

# Nonparametric - Spearman's

<b>IQ, <math>X_i</math></b> ♦	<b>Hours of TV per week, <math>Y_i</math></b> ♦	<b>rank <math>x_i</math></b> ♦	<b>rank <math>y_i</math></b> ♦	<b><math>d_i</math></b> ♦	<b><math>d_i^2</math></b> ♦
86	2	1	1	0	0
97	20	2	6	-4	16
99	28	3	8	-5	25
100	27	4	7	-3	9
101	50	5	10	-5	25
103	29	6	9	-3	9
106	7	7	3	4	16
110	17	8	5	3	9
112	6	9	2	7	49
113	12	10	4	6	36

# Nonparametric - Spearman's

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

$$F = \frac{r^2(N - 2)}{1 - r^2}$$

# Nonparametric - Spearman's

```
IQ = c(86, 97, 99, 100, 101, 103, 106, 110, 112, 113)
TVhours = c(2, 20, 28, 27, 50, 29, 7, 17, 6, 12)
dsquared = c(0, 16, 25, 9, 25, 9, 16, 9, 49, 36)
N = length(IQ)
rho = 1 - (6 * sum(dsquared)) / (N*(N^2-1))
Fstat1 <- (rho^2*(N-2)) / (1-(rho^2))
# Area under F-Distribution, using pf() function
pval1 <- 1 - pf(Fstat1, 1, N-2) # Fstat, parameters (k - 1), df (N - k)
rho
```

```
## [1] -0.1757576
```

```
pval1
```

```
## [1] 0.6271883
```

IQ and TV hours are not correlated



# Nonparametric - Spearman's

Using the Built in R function

```
cor.test(IQ, TVhours, method = "spearman")$estimate # rho
```

```
##          rho
```

```
## -0.1757576
```

```
cor.test(IQ, TVhours, method = "spearman")$p.value #p-value
```

```
## [1] 0.6319674
```

note: p-value calculated with a nonparametric approximation method.

# Next Week

Analysis of Variance (ANOVA) - between (one-way)