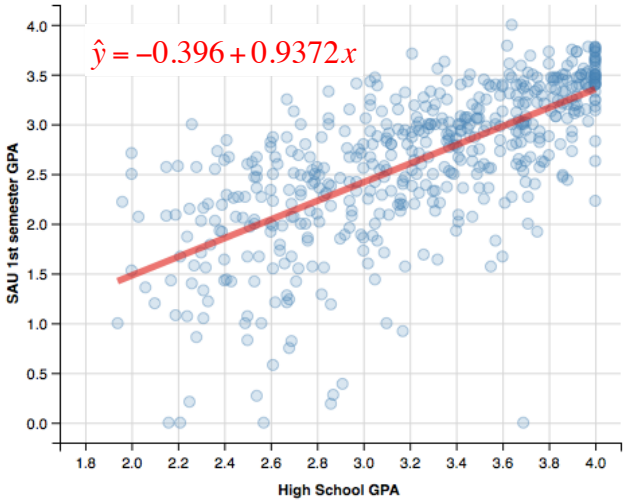
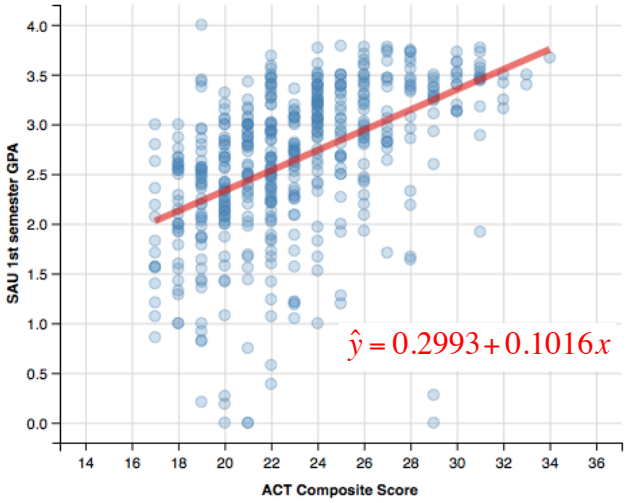
Scenario: Can you predict the success of SAU students? How would you define success? What variables would predict success?



$\bar{X} = 3.2354$ $\bar{Y} = 2.6362$

$s_x = 0.5435$ $s_y = 0.7510$

$n = 508$ $r_{xy} = 0.6784$



$\bar{X} = 23.002$ $\bar{Y} = 2.6362$

$s_x = 3.6176$ $s_y = 0.7510$

$n = 508$ $r_{xy} = 0.4894$

source: 2013-14 MAP-Works data

1. On each scatterplot, I plotted the line that **best fits** the data. Evaluate how well each line describes the relationship between X and Y. In which scatterplot does the line better fit the data?

2. Interpret the slope and y-intercept of the scatterplot on the top-left.

   slope: y-intercept:

3. Predict the average first-semester GPA of students with average high school GPAs of 3.50 and 1.50.

   Predicted first-semester GPA for hsGPA = 3.50: _____.

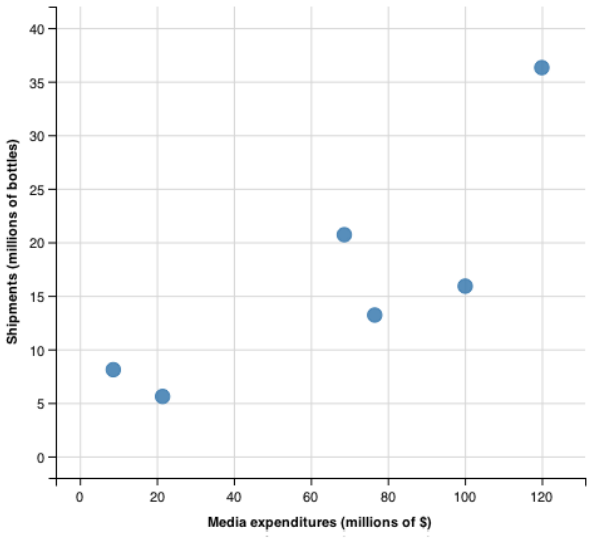
   Predicted first-semester GPA for hsGPA = 1.50: _____.

   I am more confident in my prediction for students with an average high school GPA of: **1.50** **3.50**

To learn how we can find the best-fitting line, let's use a small dataset of the relationship between **media expenditures** and **number of bottles shipped** for 6 beer brands:

Brand	Media Expenditures (millions of $)	Bottles Shipped (in millions)
Busch	8.7	8.1
Miller Genuine Draft	21.5	5.6
Bud Light	68.7	20.7
Coors Light	76.6	13.2
Miller Lite	100.1	15.9
Budweiser	120.0	36.3
mean	65.9333	16.6333
std. dev	43.5017	11.0471
correlation	**correlation: r = 0.8288**	

Source: Superbrands, 1998; 10/20/1997



With this data, we can construct the model: $\text{bottles shipped} = f(\text{media expenditures})$ or $y = f(x) + e$

The correlation indicates y is a linear function of x, so our function will have a slope (b_1) and y-intercept (b_0):

$$y = f(x \mid b_0, b_1) + e$$

or

$$y = b_0 + b_1 x + e$$

Our goal will be to estimate the parameters (slope and y-intercept) of this model to evaluate how well the model fits the data.

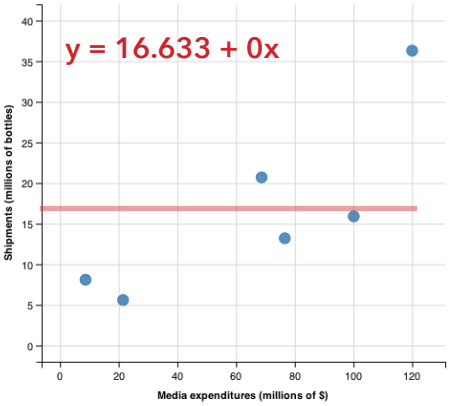
4. On the scatterplot displayed above, sketch the line you think best fits the data. Estimate the parameters of that line.

Y = _____ (x) + _____

(slope) (y-intercept)

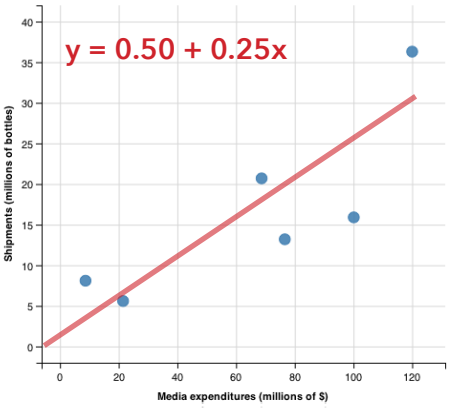
5. Each student has a different line with different values for the slope and y-intercept. How can we determine which student has the best line? How can we evaluate how well that "best" line fits the data?

Suppose 3 students sketched the following lines. Which line is best? What do the numbers in the tables represent?

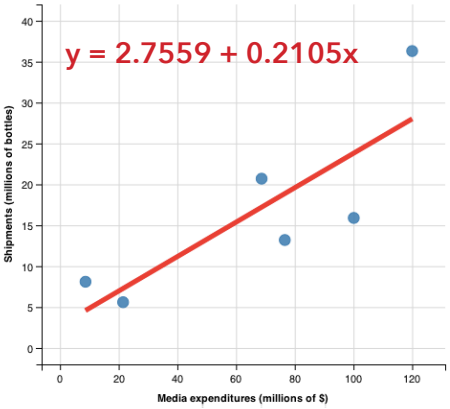


Note: This would be our best prediction if we didn't know anything about the relationship between media expenditures and bottles shipped.

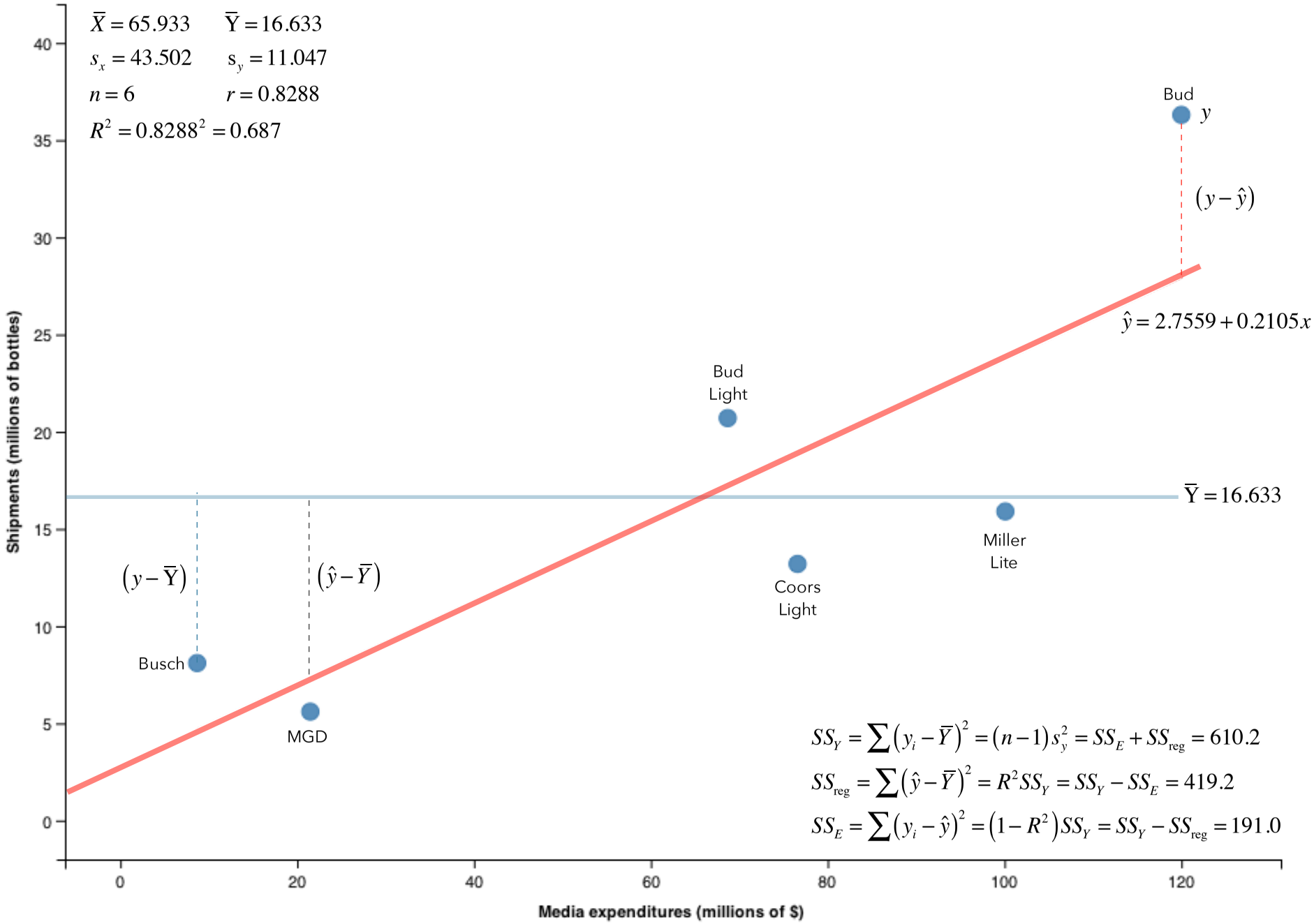
Observed		Predicted	error	error^2
Media (x)	Shipped (y)	$\hat{y}$	$(y-\hat{y})$	$(y-\hat{y})^2$
8.7	8.1	16.633	-8.53	72.761
21.5	5.6	16.633	-11.03	121.661
68.7	20.7	16.633	4.07	16.5649
76.6	13.2	16.633	-3.43	11.765
100.1	15.9	16.633	-0.73	0.533
120.0	36.3	16.633	19.67	386.909
		Sum =	0	**610.194**



Observed		Predicted	error	error^2
Media (x)	Shipped (y)	$\hat{y}$	$(y-\hat{y})$	$(y-\hat{y})^2$
8.7	8.1	2.675	5.43	29.485
21.5	5.6	5.875	-0.28	0.078
68.7	20.7	17.675	3.02	9.12
76.6	13.2	19.65	-6.45	41.603
100.1	15.9	25.525	-9.62	92.544
120.0	36.3	30.5	5.8	33.64
		Sum =	-2.1	**206.47**



Observed		Predicted	error	error^2
Media (x)	Shipped (y)	$\hat{y}$	$(y-\hat{y})$	$(y-\hat{y})^2$
8.7	8.1	4.58725	3.51	12.32
21.5	5.6	7.28165	-1.68	2.822
68.7	20.7	17.21725	3.48	12.11
76.6	13.2	18.8802	-5.68	32.262
100.1	15.9	23.82695	-7.93	62.885
120.0	36.3	28.0	8.28	68.558
		Sum =	-0.02	**190.957**


$\bar{X} = 65.933$ $\bar{Y} = 16.633$
$s_x = 43.502$ $s_y = 11.047$
$n = 6$ $r = 0.8288$
$R^2 = 0.8288^2 = 0.687$
Bud
y
$(y - \hat{y})$
$\hat{y} = 2.7559 + 0.2105x$
Bud Light
$\bar{Y} = 16.633$
Miller Lite
Coors Light
$(y - \bar{Y})$
$(\hat{y} - \bar{Y})$
Busch
MGD
$SS_Y = \sum (y_i - \bar{Y})^2 = (n-1)s_y^2 = SS_E + SS_{reg} = 610.2$
$SS_{reg} = \sum (\hat{y} - \bar{Y})^2 = R^2 SS_Y = SS_Y - SS_E = 419.2$
$SS_E = \sum (y_i - \hat{y})^2 = (1 - R^2) SS_Y = SS_Y - SS_{reg} = 191.0$
Shipments (millions of bottles)
0
5
10
15
20
25
30
35
40
0
20
40
60
80
100
120
Media expenditures (millions of $)

6. The best-fitting line in this scenario is **y = 2.7559 + 0.2105x**. Use that line to "predict" the number of bottles shipped by a company that spent $76.6 million on advertising.

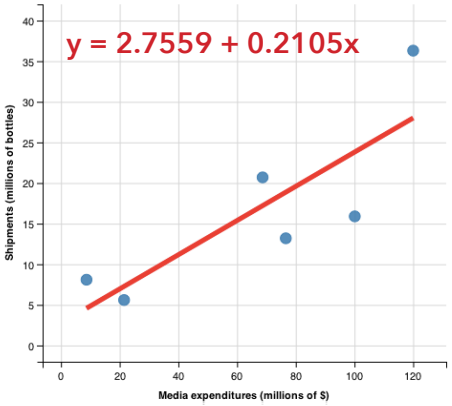
   Predicted number of bottles shipped = f(76.6) = _____

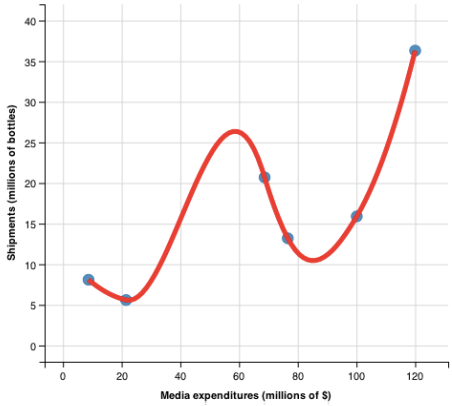
   Coors Light spent $76.6 million on advertising and shipped **13.2** million bottles. How far off was your prediction?

   Prediction error = _____

7. No line – not even the best line – will fit the data with perfect accuracy. There's always some amount of random error (and, probably, measurement error).

   Why, then, wouldn't we connect-the-dots to create our prediction model? Why might we prefer an imperfect line to a perfect connect-the-dots model?

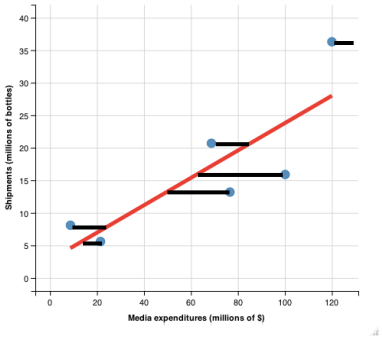




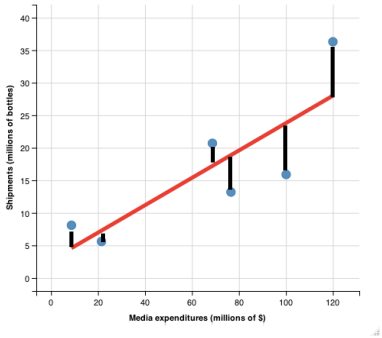

```
# Input data
brand <- c("Busch", "MGD", "Bud Light", "Coors Light", "Miller Lite", "Budweiser")
media <- c(8.7, 21.5, 68.7, 76.6, 100.1, 120.0)
ship <- c(8.1, 5.6, 20.7, 13.2, 15.9, 36.3)

# Store all variables in beer data.frame and load ggvis package for scatterplot
beer <- data.frame(brand, media, ship)
library(ggvis)

# Scatterplot with best-fitting line and lowess curve
beer %>%
  ggvis(x=~media, y=~ship,
        fill := "steelblue", stroke:="steelblue", fillOpacity:=.9, strokeOpacity:=1) %>%
  layer_points(size:=100) %>%
  layer_model_predictions(model = "lm", strokeWidth:=3, stroke:="red") %>%
  layer_smooths(strokeWidth:=5, stroke:="red") %>%
  add_axis("x", title = "Media expenditures (millions of $)", grid=F) %>%
  scale_numeric("x", domain=c(0, 125), nice=FALSE, clamp=TRUE)  %>%
  add_axis("y", title = "Shipments (millions of bottles)", grid=F) %>%
  scale_numeric("y", domain=c(0, 40), nice=FALSE, clamp=TRUE)

# Construct linear model
model <- lm(ship~media, data=beer) # Construct linear model
model                              # Get slope and y-intercept
summary(model)                     # Print additional information about model
plot(model)                        # Plots to evaluate conditions/assumptions
mplot(model)                       # Plot confidence interval for parameter estimates
predict(model)                     # Predict values of Y from the model
```

8. One criterion for the best line would be the line that minimizes the total amount of prediction error (the sum of the distances between the points and the line).

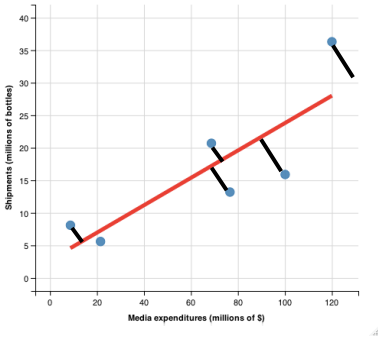
   If distances between points and the prediction line represent error, which distances (errors) are we interested in minimizing? Do we want to minimize the horizontal, vertical, or perpendicular distances? Why?



horizontal errors

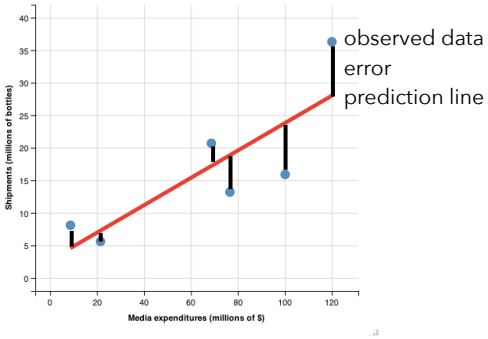


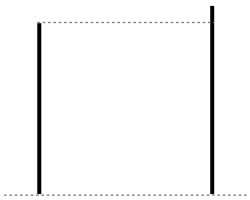
vertical errors
least squares
assumes x values are "good" measures
or that we chose the x values



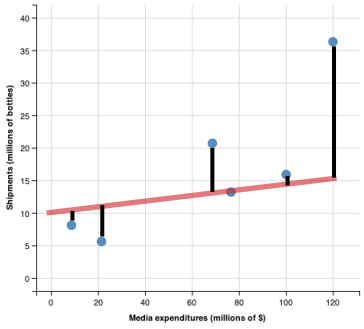
perpendicular errors
error-in-both-variables regression
orthogonal (Deming) regression
"perpendicular" changes as units change

9. If we want to predict Y for given values of X, we want to minimize the vertical errors. Below, I drew these vertical errors for two potential best-fit lines. I then calculated the total length of the lines to find the sum of these errors:





total error (sum of
all line lengths)



Since the sum of the errors is smaller for the line on the left, that line better fits our data. To find the absolute best line, all we need to do is find the sum of the errors for every possible line we could draw for our data.

That could take forever, so let's use some math to find the formula for the line that best fits a given dataset.

To do this, let's establish some notation: $(x_i, y_i) \leftarrow$ the coordinate of a data point

$$\hat{y}_i = b_0 + b_1 x_i + e \leftarrow \text{our linear model}$$

$$e = y_i - (\hat{y}_i) = y_i - (b_0 + b_1 x_i) = y_i - b_0 - b_1 x_i \leftarrow \text{error}$$

We want to find the line that minimizes the sum of those errors. We want to minimize: $\sum_{i=1}^{n} y_i - b_0 - b_1 x_i$

The problem is that the positive errors will cancel out the negative errors. If we find the sum of these values, the positive and negative errors will cancel each other out. Also, the minimum of our criterion would approach -∞.

How can we deal with this issue? How can we ensure all the errors are positive?

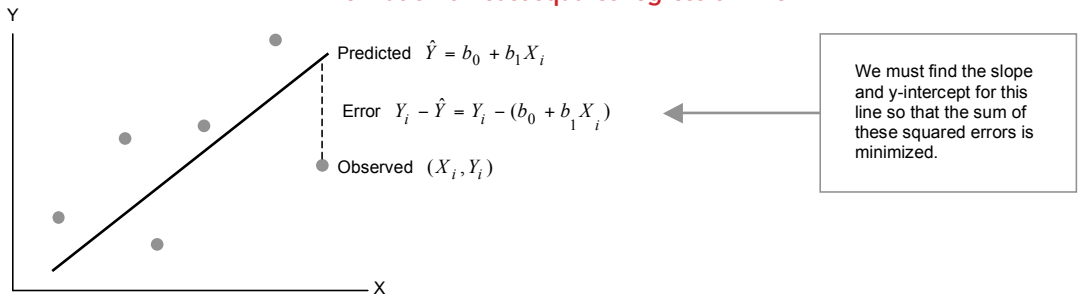
We could take the absolute value of our errors. We could minimize: $\sum_{i=1}^{n} |y_i - b_0 - b_1 x_i|$

This is the approach used in *quantile regression* (which we'll learn about later in the semester). One of the problems with absolute values is that they're difficult to work with algebraically.

Another approach would be to square each error. We'd then want to minimize: $\sum_{i=1}^{n} (y_i - b_0 - b_1 x_i)^2$

This gives us a function we can minimize using Calculus. It also has the bonus of magnifying outliers. When we square large errors (outliers), those squared errors get huge.

Derivation of least squares regression line



Let Q represent the sum of squared errors: $Q = \sum_{i=1}^{N} (Y_i - b_0 - b_1 X_i)^2$

We need to find values for b_0 and b_1 that will minimize Q. We know that to minimize a function, we must set its first derivative equal to zero and solve. Because we have two variables in this function, we'll need to take partial derivatives of Q with respect to b_0 and b_1.

Partial derivative of Q with respect to b_0: (we treat b_0 as a variable and all other terms as constants)

$$\frac{\partial Q}{\partial b_0} = \frac{\partial \sum_{i=1}^{N} (Y_i - b_0 - b_1 X_i)^2}{\partial b_0} = 2\sum (Y_i - b_0 - b_1 X_i) \frac{\partial (Y_i - b_0 - b_1 X_i)}{\partial b_0} = -2\sum (Y_i - b_0 - b_1 X_i)$$

(Chain Rule)

We set this partial derivative equal to zero: $-2\sum (Y_i - b_0 - b_1 X_i) = 0$

$$\sum (Y_i - b_0 - b_1 X_i) = 0$$

$$\sum Y_i = n b_0 + b_1 \sum X_i$$

Partial derivative of Q with respect to b_1: (we treat b_1 as a variable and all other terms as constants)

$$\frac{\partial Q}{\partial b_1} = \frac{\partial \sum_{i=1}^{N} (Y_i - b_0 - b_1 X_i)^2}{\partial b_1} = 2\sum (Y_i - b_0 - b_1 X_i) \frac{\partial (Y_i - b_0 - b_1 X_i)}{\partial b_1} = -2\sum X_i (Y_i - b_0 - b_1 X_i)$$

(Chain Rule)

Set the partial derivative equal to zero: $-2\sum X_i (Y_i - b_0 - b_1 X_i) = 0$

$$\sum X_i (Y_i - b_0 - b_1 X_i) = 0$$

$$\sum X_i Y_i - b_0 \sum X_i - b_1 \sum X_i^2 = 0$$

$$\sum X_i Y_i = b_0 \sum X_i + b_1 \sum X_i^2$$

Now we must solve this system of two *normal* equations…

System of normal equations:

$$\sum Y_i = nb_0 + b_1 \sum X_i$$
$$\sum X_i Y_i = b_0 \sum X_i + b_1 \sum X_i^2$$

This system can be solved to get: $b_1 = \dfrac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum x_i^2 - \left(\sum x_i\right)^2}$

and

$$b_0 = \frac{\sum x_i^2 \sum y_i - \sum x_i \sum x_i y_i}{n\sum x_i^2 - \left(\sum x_i\right)^2} = \frac{\sum y_i}{n} - b_1 \frac{\sum x_i}{n} = \overline{Y} - b_1 \overline{X}$$

We can rewrite b_1 given the following information: $S_{xy} = \sum (x_i - \overline{X})(y_i - \overline{Y}) = \sum x_i y_i - \frac{1}{n}\sum x_i \sum y_i$

$$S_{xx} = \sum (x_i - \overline{X})^2 = \sum x_i^2 - \frac{1}{n}\left(\sum x_i\right)^2$$

$$S_{yy} = \sum (y_i - \overline{Y})^2 = \sum y_i^2 - \frac{1}{n}\left(\sum y_i\right)^2$$

Therefore, $b_1 = \dfrac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum x_i^2 - \left(\sum x_i\right)^2} = \dfrac{S_{xy}}{S_{xx}} = r\dfrac{S_y}{S_x}$

So, the line that minimizes the sum of squared errors has the following slope and y-intercept parameters:

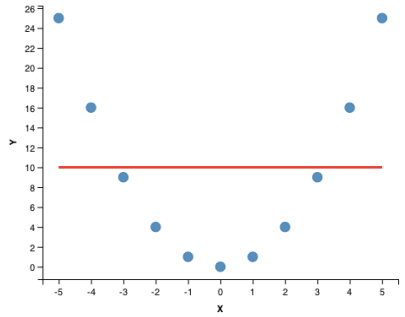
$$b_0 = \overline{Y} - b_1 \overline{X} \quad \text{and} \quad b_1 = r\frac{S_y}{S_x}$$

In our example, r = 0.829; s_y = 43.5017; s_x = 11.0471. Using the mean values of X and Y, we can compute:

$$\hat{\beta}_1 = r\frac{s_y}{s_x} = (0.829)\left(\frac{11.0471}{43.5017}\right) = 0.21 \qquad \hat{\beta}_0 = \overline{Y} - \hat{\beta}_1 \overline{X} = 16.633 - (0.21)(65.933) = 2.76$$

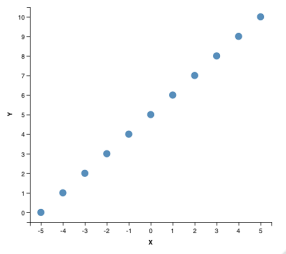
Now that we have formulas to calculate the slope and y-intercept of our least-squares regression line (the line of "best" fit), we need to find some way to determine if that line is any good. We can find the least-squares regression line for any dataset, but it doesn't mean the line is a good fit (or meaningful).

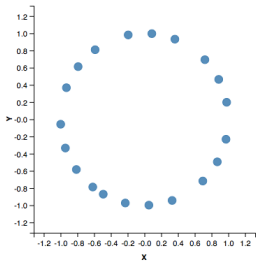
See the example to the right:



10. We're going to try out several measures of how well our regression line fits the data. Let's see if we can figure out the value of each measure under two situations: (a) a model that fits perfectly, and (b) a model that doesn't fit at all.

Eventually, we'll want to fill-in this table:





Measure / Index	Value for perfect fit	Value for no fit
$SS_E = \sum (y_i - \hat{y})^2$	_____	$\sum (y_i - \bar{Y})^2 = SS_y = (n-1)s_y^2$
$s_{y\|x}^2 = \frac{SS_E}{df_E} = \frac{\sum (y_i - \hat{y})^2}{n-2}$	_____	$\left(\frac{n-1}{n-2}\right) s_y^2 = \frac{SS_Y}{n-2}$
$s_{y\|x} = \sqrt{\frac{SS_E}{df_E}} = \sqrt{\frac{\sum (y_i - \hat{y})^2}{n-2}}$	_____	$s_y \sqrt{\frac{n-1}{n-2}} = \sqrt{\frac{SS_Y}{n-2}}$
$1 - R^2 = \frac{SS_E}{SS_Y} = \frac{\sum (y_i - \hat{y})^2}{\sum (y_i - \bar{Y})^2}$	_____	_____
$R^2 = \frac{SS_{\text{reg}}}{SS_Y} = \frac{\sum (\hat{y} - \bar{Y})^2}{\sum (y_i - \bar{Y})^2}$	_____	_____

Later, we'll investigate other measures of model fit, such as log-likelihood and Akaike's AIC (an information criterion)

11. One possible measure of how well a regression line fits the data is SSE (the sum of the squared vertical errors). What would SSE equal if the line fit the data perfectly?

Now suppose we have uncorrelated variables – knowing the value of X would not tell us anything about the value of Y. What would the least-squares regression line look like in this case?

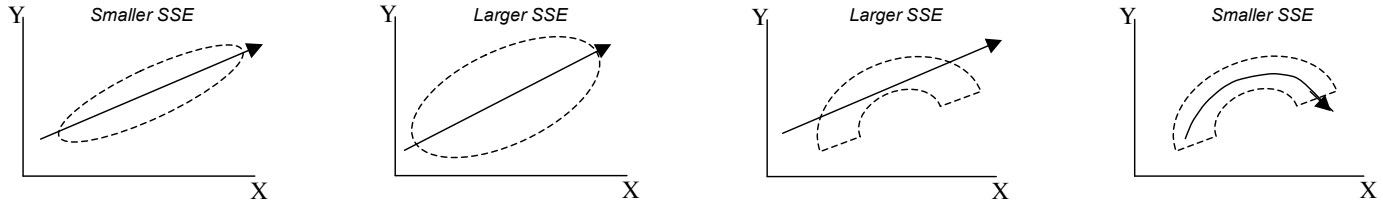
We'd need to find the value of *m* that would minimize the following: $\sum (y_i - M)^2$

If you remember from a previous statistics class, this value is minimized when M equals the sample mean.

Therefore, our best prediction for uncorrelated variables would be: $\sum (y_i - \bar{Y})^2$

That formula should look familiar. That's **SStotal** from ANOVA (or SSy, as we'll refer to it in regression). What's the largest value we could possible get for SSE?

12. The size of SSE depends on a few factors, such as the amount of variation in our data, the number of observations we have in our data, and the degree to which a line fits the data.



It seems problematic that adding data would necessarily increase the size of SSE. That would imply our line fits worse when we have a larger sample of data. Perhaps it would be better to calculate the <u>average</u> squared error (or mean squared error). This would give us the *variance of the estimate*:

$$s^2_{y|x} = \frac{SS_E}{df_E} = \frac{\sum\left(y_i - \hat{y}\right)^2}{n-2}$$

This variance of the estimate represents the <u>average squared distance from each observation to the prediction line</u>. In a situation with perfect fit, what would this measure equal? Write that in the table on the previous page.

With uncorrelated variables, what would be the maximum value of the variance of the estimate?

$$\max\left\{s^2_{y|x}\right\} = \frac{\sum\left(y_i - \overline{Y}\right)^2}{n-2} = \left(\frac{n-1}{n-2}\right)\left(\frac{\sum\left(y_i - \overline{Y}\right)^2}{n-1}\right) = \left(\frac{n-1}{n-2}\right)s^2_y = \frac{SS_Y}{n-2}$$

13. Perhaps it would be better if our measure of good-fit was <u>not</u> in squared units. We can fix that easily enough:

$$s_{y|x} = \sqrt{s^2_{y|x}} = \sqrt{\frac{SS_E}{df_E}} = \sqrt{\frac{\sum\left(y_i - \hat{y}\right)^2}{n-2}}$$

Since this is simply the square root of the variance of the estimate, it's easy to find the values under situations with perfect and no fit. These values have been filled-in the table on the previous page.

This measure is called the **standard error of the estimate**. What does it represent? Sketch a scatterplot and show what the standard error of the estimate would be visually.

14. The maximum value of each of our measures is still unbounded. Ideally, the maximum value would be fixed.

    Suppose we took our total sums of squares (the variation in Y) and partitioned it.
    We know some of that variation is **unexplained** by our regression line, so we could calculate:

$$\frac{SS_E}{SS_Y} = \frac{\sum\left(y_i - \bar{Y}\right)^2}{\sum\left(y_i - \hat{y}\right)^2} = 1 - r^2$$

    Under the perfect-fit and no-fit scenarios, what would be the value of this ratio of error variance to total variance?
    Fill-in the table to show the values of this measure under perfect and no-fit situations.

15. That measure seems backwards – it equals zero when we have perfect fit and 1 when we have no fit.

    Let's invert that by taking: $r^2 = R^2 = \frac{SS_Y - SS_E}{SS_Y} = \frac{SS_{reg}}{SS_Y} = \frac{\sum\left(\hat{y} - \bar{Y}\right)^2}{\sum\left(y_i - \hat{y}\right)^2}$

    This is the **coefficient of determination** and it has the same interpretation as **eta-squared** in an ANOVA.
    Fill-in the table to show the values of this measure under perfect and no-fit situations.

16. Perhaps the most popular (basic) measures of how well a line models a dataset are the *coefficient of determination* and the standard error of the estimate (a.k.a. the RMSE, the *root mean squared error)*. Identify an advantage of each measure.

    Advantage of coefficient of determination:

    Advantage of standard error of estimate:

17. We've derived the least squares criterion and formulas to calculate the slope and y-intercept for that line of best fit. We've also derived some measures indicating how well that best-fitting line actually fits the data. We still need to:
    a) Practice using technology to estimate these regression lines
    b) Figure out how to determine if a regression line fits the data "good enough"
    c) Investigate the assumptions we're making when we estimate these least-squares regression lines.

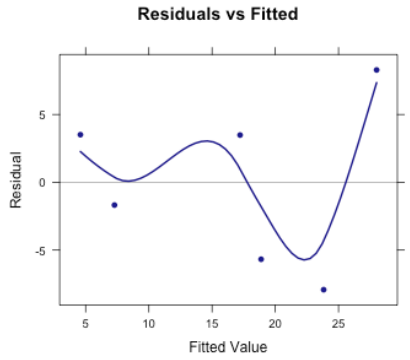
```r
# The mosaic, ggvis, and broom packages have been loaded using
library(mosaic)
library(ggvis)
library(broom)

The beer dataset has been loaded into memory (variables = brand, media, and ship)

# Scatterplot with best-fitting line (using ggvis package)
beer %>%
  ggvis(x=~media, y=~ship) %>%
  layer_points() %>%
  layer_model_predictions(model = "lm")

# To fit the linear model
model <- lm(ship~media, data=beer) # Construct linear model

# Get model coefficients
model                              # Get slope and y-intercept
```

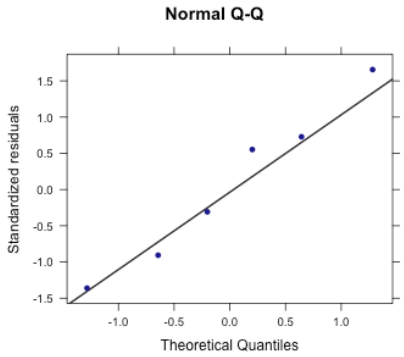
```
Call:
lm(formula = ship ~ media, data = beer)

Coefficients:
(Intercept)        media
     2.7559       0.2105
```

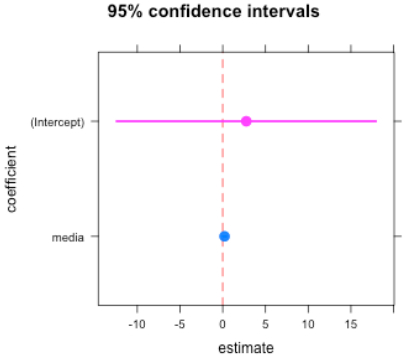
```r
summary(model)                     # Print additional information about model
```

```
Call:
lm(formula = ship ~ media, data = beer)

Residuals:
     1      2      3      4      5      6
 3.513 -1.681  3.484 -5.678 -7.925  8.287

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.75591    5.46812   0.504   0.6408
media        0.21048    0.07104   2.963   0.0414 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.911 on 4 degrees of freedom
Multiple R-squared:  0.6869,     Adjusted R-squared:  0.6087
F-statistic: 8.777 on 1 and 4 DF,  p-value: 0.04145
```

```r
mplot(model)                       # Plots of model parameters and conditions
```







```
predict(model)                    # Predict values of Y from the model
```

```
       1         2         3         4         5         6
 4.587060  7.281159 17.215652 18.878416 23.824615 28.013098
```

```
# Use the broom package to tidy up our model information
tidymodel <- tidy(model)             # Store results from model in data.frame
tidymodel                            # Display tidy model
```

```
        term  estimate  std.error statistic    p.value
1 (Intercept) 2.7559140 5.46812498 0.5039962 0.64075738
2       media 0.2104765 0.07104337 2.9626483 0.04144535
```

```
tidymodel$p.value[2]          # Access the p-value for the media coefficient
```

```
[1] 0.04144535
```

```
glance(model)                 # Glance at model summary statistics
```

```
  r.squared adj.r.squared    sigma statistic    p.value df    logLik      AIC      BIC deviance
1 0.6869445     0.6086806 6.910579  8.777285 0.04144535  2 -18.89556 43.79111 43.16639 191.0244
  df.residual
1           4
```

```
glance(model)$r.squared       # Access R-squared value
```

```
[1] 0.6869445
```

```
augment(model)                # Find predicted and residual values
```

```
  ship media   .fitted  .se.fit    .resid      .hat   .sigma    .cooksd .std.resid
1  8.1   8.7  4.587060 4.948950  3.512940 0.5128581 7.431721 0.27923444  0.7283307
2  5.6  21.5  7.281159 4.233682 -1.681159 0.3753252 7.884584 0.02846155 -0.3077992
3 20.7  68.7 17.215652 2.828071  3.484348 0.1674756 7.669016 0.03071440  0.5525972
4 13.2  76.6 18.878416 2.921233 -5.678416 0.1786914 7.112538 0.08943068 -0.9066916
5 15.9 100.1 23.824615 3.721721 -7.924615 0.2900406 5.847202 0.37834613 -1.3609651
6 36.3 120.0 28.013098 4.765840  8.286902 0.4756090 4.474642 1.24355572  1.6559613
```

```
anova(model)                  # ANOVA summary table for model
```

```
Analysis of Variance Table

Response: ship
          Df Sum Sq Mean Sq F value  Pr(>F)
media      1 419.17  419.17  8.7773 0.04145 *
Residuals  4 191.02   47.76
```

```
confint(model)                    # Confidence intervals for model coefficients
```

```
                  2.5 %     97.5 %
(Intercept) -12.42603485 17.9378628
media         0.01322851  0.4077246
```

18. The conditions of our linear regression model (in order from most to least important):

- Validity: The data you are analyzing maps to the research question you are trying to answer.
  Diagnosis: Take a careful look at the purpose of your study and the data you've collected
  How to fix: Get better data

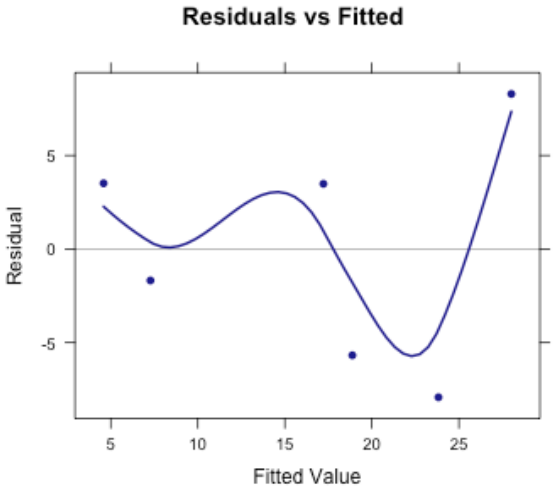
- Additivity and linearity: The deterministic component of the model is a linear function of the predictors.
  Diagnosis: Look at plots of observed vs predicted or residuals vs predicted values. The points should be symmetrically distributed around a diagonal line in the former plot or around horizontal line in the latter plot, with a roughly constant variance.
  How to fix: You could transform your data (if it seems appropriate) or add a nonlinear component

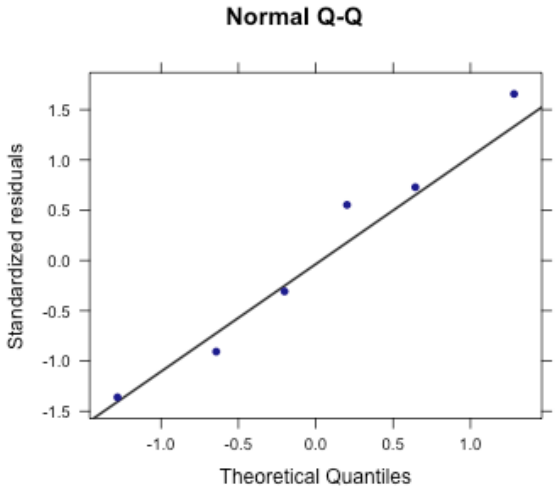
- Independent errors: No correlation among errors
  Diagnosis: If you have time series data, be careful that consecutive errors are not related.

- Equal variance of errors (*homoscedasticity*): The variance in the errors is the same across all levels of X.
  Diagnosis: Look at the plot of residuals vs predicted values. If the residuals grow larger as a function of X, you have a problem.

- Normality of errors
  Diagnosis: Look at a P-P or Q-Q plot of the residuals. The residuals should fall near the diagonal line. You could also run a test for normality, like the Shapiro-Wilk or Kologorov-Smirnov tests. Note that the dependent and independent variables in a regression model do not need to be normally distributed by themselves--only the prediction errors need to be normally distributed

Evaluate these conditions based on the following plots:

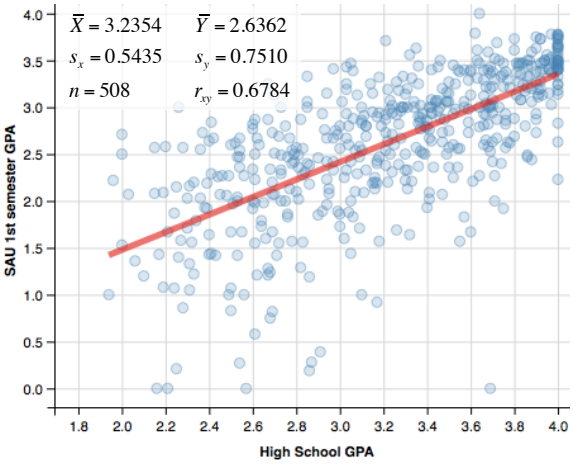




19. In the R output, you'll notice confidence intervals, p-values, and an ANOVA summary table. To learn what these are all about, let's go back to our scenario of predicting first semester GPAs based on high school GPAs.

    The scatterplot is, once again, displayed to the right.
    You can download this data at:
    http://www.bradthiessen.com/html5/data/actgpa.csv

    From the output displayed below, write out and interpret the coefficients and R-squared value for the linear model.




```
gpa.model <- lm(fallGPA ~ hsGPA, data=actgpa)                    # Construct linear model
gpa.model
```

```
Call:
lm(formula = fallGPA ~ hsGPA, data = actgpa)

Residuals:
     Min       1Q   Median       3Q      Max
-3.06226 -0.27925  0.08677  0.33371  1.27793

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.39601    0.14801  -2.675   0.0077 **
hsGPA        0.93720    0.04512  20.773   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5523 on 506 degrees of freedom
Multiple R-squared:  0.4603,  Adjusted R-squared:  0.4592
F-statistic: 431.5 on 1 and 506 DF,  p-value: < 2.2e-16
```

20. In the last lesson, we saw that we always expect to get a non-zero correlation coefficient in a sample of data (even when we expect the variables to be uncorrelated in the population). We expect the same thing with slopes of regression lines.

    Even if first semester GPAs have no relationship with high school GPAs, we expect to calculate a non-zero slope for our best-fitting line. The slope of the regression line from our data was estimated to be 0.9372. Does this slope imply our variables have a relationship for our population of interest or does this slope arise from random variation in our sample data?

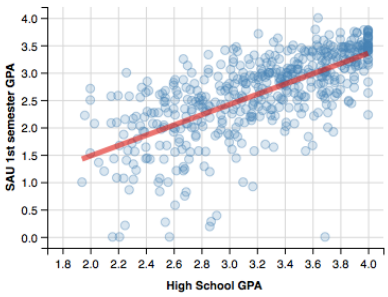
    Another way of asking this question is: How unlikely were we to observe a slope of 0.9372 or greater if the data were uncorrelated?

    Explain how we're going to estimate this likelihood using randomization-based methods.

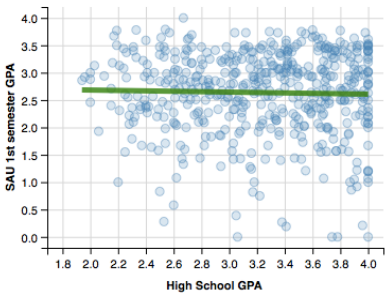
21. Let's assume there is no relationship between high school and first semester GPAs. If that was the case, then any high school GPA in our sample data could be associated with any first semester GPA. Someone with a 4.00 high school GPA would be just as likely to have a 3.50 first semester GPA as they would a 0.83 first semester GPA.

    With this logic, we can randomize the values of one variable (high school GPA) while holding the other variable constant. Then, with this randomized sample, we can calculate the slope of our regression line. We can repeat this process many times, yielding many randomization-based estimates of our slope parameter.

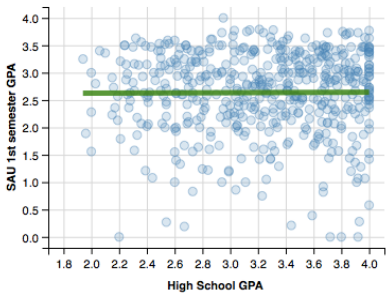
    Let's take a look at a few randomizations:



Actual data
slope = 0.9372



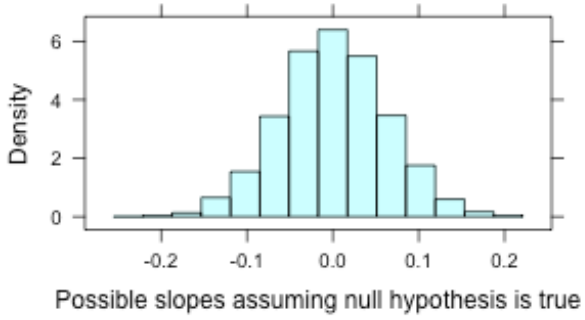
Randomization #1
slope = -0.03754



Randomization #2
slope = 0.008652

If we repeat this process 10,000 times, we can estimate our randomization distribution of slopes:

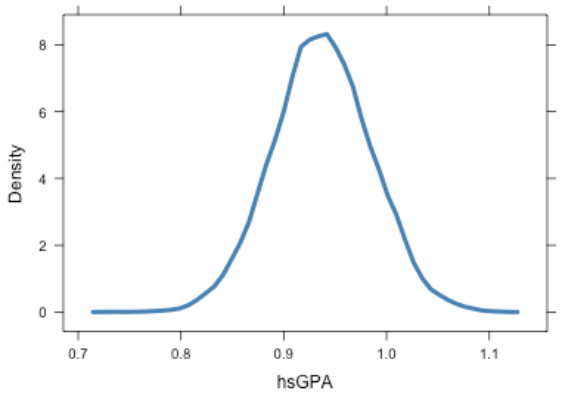
```
test.slope <- 0.9372                                    # Store our observed slope as "test.slope"
rand.slopes <- do(10000) * lm(fallGPA ~ shuffle(hsGPA), data=actgpa)       # 10,000 randomizations

# Histogram with p-value
histogram(~hsGPA., data=rand.slopes,
          xlab="Possible slopes assuming null hypothesis is true",
          groups=hsGPA. >= test.slope,   # Highlight values > test statistic
          main=paste("p-value = ", prop(~hsGPA. >= test.slope, data=rand.slopes)))
ladd(panel.abline(v=test.slope))   # Add vertical line at test statistic
```



Based on this histogram, estimate the p-value and state your conclusion.

22. We can also construct a bootstrap confidence interval for the slope of our regression line. Explain the bootstrap process and interpret this interval:

```
bstrap <- do(10000) * lm(fallGPA ~ hsGPA, data=resample(actgpa))    # 10,000 bootstrap samples

densityplot(~hsGPA, data=bstrap, plot.points = FALSE, col="steelblue", lwd=4)      # Plot distribution

cdata(0.95, hsGPA, data = bstrap)     # Get 95% CI
```




```
       low        hi central.p
0.8440269 1.0308139 0.9500000
```

23. You'll practice using randomization-based tests and constructing bootstrap confidence intervals for the slopes of linear models in the assignment associated with this lesson. For now, let's move on to theory-based tests we can use to evaluate the fit of a model to a given dataset.

    When fitting linear models, we may be interested in finding the most **parsimonious** model that can explain *enough* of the relationship between the variables. To find the best model, we might compare several different models, each increasing in complexity (*nested* models). For example:

    a) We could start with the most basic model that predicts the same value for Y regardless of X. All variation in observed Y values would be modeled by random error: $\hat{y}_i = b_0 + e_i$. What value would we choose for b_0?

    b) We could then add one predictor to the model to create: $\hat{y}_i = b_0 + b_1 x_1 + e_i$. We could compare the performance of this model to the previous model to determine if the improvement in prediction justified the additional complexity of adding the predictor.

    c) We could then add yet another predictor: $\hat{y}_i = b_0 + b_1 x_1 + b_2 x_2 + e_i$. Once again, we could compare this model to the previous model. If this new model provided a significantly better prediction (explained a significant amount of previously unexplained variance), then we could decide to keep this new model. If the model didn't improve our prediction by very much, we might decide to keep the previous, simpler model.

    At each stage in building our regression model, we can assess the value of adding predictors (complexity) through randomization-based or parametric hypothesis testing methods. These methods can help us determine which predictors to keep in our model.

    We could also work through this process backwards. We could start with a relatively complex model, take away the predictor that explains the lease amount of variance in Y, and determine if the simpler model was significantly worse than the more complex model.

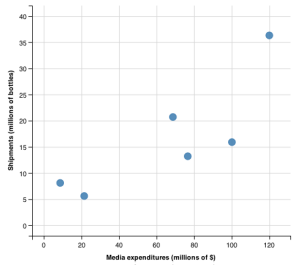
24. When comparing models, it's helpful to write out the **full model** (more complex model) and the **reduced model**. When you're analyzing your own data, you'll choose these models based on your experience with the data or area of study). For now, I'll force us to choose specific models.

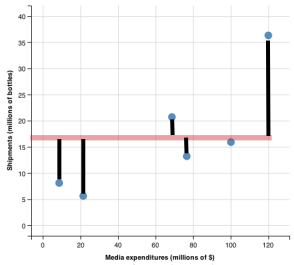
    We'll work one last time with the beer data set. I want to know if X (media expenditures) predicts Y (bottles shipped) better than a model with no predictors. Write out our full and reduced models:

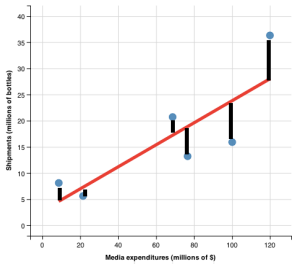
    Full model: _____ Reduced model: _____

25. As we've already seen, the sample mean minimizes the sum of squared errors (if we have no predictor variables). Therefore, what does SSE represent in our reduced model?

    Our full model is the least-squares regression line (using one predictor variable). As you can see below, the full model reduced our error variance by 610.193 – 191.025 = 419.168. What does this value represent?







Observed		**Reduced Model**			**Full Model**		
Media (x)	Shipped (y)	predicted	error	error^2	predicted	error	error^2
8.7	8.1	16.633	-8.533	72.812	4.587	3.513	12.339
21.5	5.6	16.633	-11.033	121.727	7.282	-1.682	2.828
68.7	20.7	16.633	4.067	16.541	17.217	3.483	12.130
76.6	13.2	16.633	-3.433	11.785	18.880	-5.680	32.265
100.1	15.9	16.633	-0.733	0.537	23.827	-7.927	62.837
120.0	36.3	16.633	19.667	386.791	28.016	8.284	68.626
			Sum	**610.193**		Sum	**191.025**

26. Fill-in these SSy and SSE values in the ANOVA summary table. How many degrees of freedom will we have?

Source of variation	**SS**	**df**	**MS**	**MSR (F)**
Regression (b_1 | b_0)	_____	_____	_____	_____
Error	_____	_____	_____	(blank)
Total	_____	_____	MS_{total}	R^2 = _____

Complete the ANOVA summary table and estimate a p-value. What conclusion could we make? Remember, you can always use the F-distribution applet at: http://lock5stat.com/statkey/theoretical_distribution/theoretical_distribution.html#F

27. The only difference between our full and reduced models is the b_1 coefficient (the slope). If $b_1 = 0$, the full model would be the same as our reduced model. Another way, then, to compare our full and reduced models would be to test the hypothesis: **H_0: $b_1 = 0$**. We can test this hypothesis with a t-test.

$$t_{n-2} = \frac{(\text{observed value}) - (\text{hypothesized value})}{\text{standard error}} = \frac{\hat{b}_1 - 0}{SE_{b_1}} =$$

$$t_{n-2} = \frac{\hat{b}_1 - 0}{SE_{b_1}} = \frac{\hat{b}_1}{\frac{s_{y|x}}{s_x\sqrt{n-1}}} = \frac{r_{xy}\frac{s_y}{s_x}}{\frac{s_y\sqrt{1-r^2}\sqrt{\frac{n-1}{n-2}}}{s_x\sqrt{n-1}}} = \frac{r_{xy}\frac{s_y}{s_x}s_x\sqrt{n-1}}{s_y\sqrt{1-r^2}\sqrt{\frac{n-1}{n-2}}} = \frac{r_{xy}\sqrt{n-1}}{\sqrt{1-r^2}\sqrt{\frac{n-1}{n-2}}}$$

$$t_{n-2} = \sqrt{\frac{r_{xy}^2(n-1)}{\left(1-r^2\right)\left(\frac{n-1}{n-2}\right)}} = \sqrt{\frac{r_{xy}^2(n-2)}{\left(1-r^2\right)}} = \frac{r_{xy}\sqrt{n-2}}{\sqrt{1-r^2}} = \frac{r_{xy}-0}{\sqrt{\frac{1-r^2}{n-2}}} = \frac{r_{xy}-0}{SE_{r_{xy}}}$$

What did that derivation just show? Conduct this t-test and state an appropriate conclusion. Compare the value of your t-statistic to the value of the MSE you calculated for the ANOVA.

28. This time, conduct a test of the hypothesis: **H_0: $r_{xy} = 0$**.

29. A test for the slope of a regression line is the same as a test for the correlation between x and y. So why do we use an ANOVA to compare our full and reduced models? Follow along:

$$F = MSR = \frac{MS_{reg}}{MS_E} = \frac{SS_{reg}/df_{reg}}{SS_E/df_E} = \frac{SS_{reg}\left(df_E\right)}{SS_E\left(df_{reg}\right)} = \frac{r^2SS_Y(n-2)}{\left(1-r^2\right)SS_Y(1)} = \frac{r^2(n-2)}{\left(1-r^2\right)} = t_{n-2}^2$$

30. It can also be shown that we can calculate our omnibus F-statistic with the following:

$$F = \frac{\left(R^2_{\text{full}} - R^2_{\text{reduced}}\right) \Big/ \left(k_{\text{full}} - k_{\text{reduced}}\right)}{\left(1 - R^2_{\text{full}}\right) \Big/ \left(N - k_{\text{full}} - 1\right)}$$

Verify this formula gives us the same value for our MSR (as the ANOVA table in question #26).

Notes: The p-value from this test should be similar to a p-value we could get via randomization-based methods. Notice that to compare two *nested* models, we're really just comparing their R-squared values.

To construct an ANOVA summary table to compare two competing models, we can use the **ANOVA()** command in R:

```
# Using the beer data.frame
fullmodel <- lm(ship ~ media, data=beer)            # Fit the full model with 1 predictor
anova(fullmodel)                                    # ANOVA summary table as displayed in question #26
```

```
Analysis of Variance Table
Response: ship
          Df Sum Sq Mean Sq F value  Pr(>F)
media      1 419.17  419.17  8.7773 0.04145 *
Residuals  4 191.02   47.76
```

```
# We could also compare competing models
reducedmodel <- lm(ship ~ 1, data=beer)             # Fit the reduced model with no predictors
anova(reducedmodel, fullmodel)                      # ANOVA summary table as displayed in question #26
```

```
Analysis of Variance Table
Model 1: ship ~ 1
Model 2: ship ~ media
  Res.Df    RSS Df Sum of Sq      F  Pr(>F)
1      5 610.19
2      4 191.02  1    419.17 8.7773 0.04145 *
```

31. There are other measures (beyond R-squared) that allow us to evaluate and compare regression models:

Likelihood: The likelihood of a model is the probability it produces our data given its parameter estimates. If we assume all our observations are independent, then we can write our likelihood function as:

$$L\left(Y \mid b_0, b_1, \sigma_{e_i}^2\right) = \prod_{i=1}^{n} P\left(y_i \mid b_0, b_1, \sigma_{e_i}^2\right) \propto \exp\left(\frac{-\sum_{i=1}^{n}\left(y_i - b_0 - b_1 x_i\right)^2}{2\sigma^2}\right) / \sigma^n$$

Oftentimes, the *natural log* of the likelihood is used (rather than the likelihood itself) because it's easier to work with. The log likelihood will always be negative, with values closer to zero indicating better model fit.

To compare a full and reduced model, we can calculate the log-likelihood of each model. We know the reduced model will fit worse, so its log-likelihood will be smaller. To compare the log-likelihoods, we can take the likelihood ratio:

$$LR = -2\ln\frac{L(\text{reduced})}{L(\text{full})} = 2\left[\ln\left(L_{\text{full}}\right) - \ln\left(L_{\text{reduced}}\right)\right]$$

If this likelihood ratio is large, it means the full model provides a much better fit than the reduced model.

To make this likelihood ratio useful, we only need to know its distribution. With large samples, the likelihood ratio is distributed as a chi-square distribution with $df = df_{\text{full}} - df_{\text{reduced}}$

```
# Using the beer data.frame
fullmodel <- lm(ship ~ media, data=beer)            # Fit the full model with 1 predictor
reducedmodel <- lm(ship ~ 1, data=beer)             # Fit the reduced model with no predictors

logLik(fullmodel)                                   # Calculate log-likelihood of full model (-18.89556)
logLik(reducedmodel)                                # Calculate log-likelihood of reduced model (-22.3797)

2 * (logLik(fullmodel) - logLik(reducedmodel))      # Calculate likelihood ratio (6.968 in this example)
pchisq(6.968, df=1, lower.tail=FALSE)               # Yields p-value

# We could do all of this with one command if we load the lmtest package
library(lmtest)                                     # Load the lmtest package
lrtest(reducedmodel, fullmodel)                     # Get likelihood ratio test with p-value (p=0.0083)
```

AIC: Akaike's **a**n **i**nformation **c**riterion (derived from *Kullback-Leibler* information theory) provides another measure that can help you select the "best" model from a set of competing models. It's a criterion that seeks to find the model that has a good fit to the data with relatively few parameters (predictors). It's defined as:

$$AIC = -2\ln\left(L_{\text{model}}\right) + 2\left(b+1\right) = n\left[\ln\left(2\pi\right) + 1 + \ln\left(\frac{SS_E}{n}\right)\right] + 2\left(b+1\right) \propto n\ln\left(\frac{SS_E}{n}\right) + 2p$$

where b = the number of coefficients estimated in our model (slope(s) and intercept) and p = the number of predictors in our model. When comparing models, we prefer the model that produces the **smaller** AIC value. To obtain the AIC values in R, we can use the **AIC(model)** command.

```
AIC(reducedmodel, fullmodel)          # Calculate AIC for each model
```

```
             df      AIC
reducedmodel  2 48.75936
fullmodel     3 43.79111
```

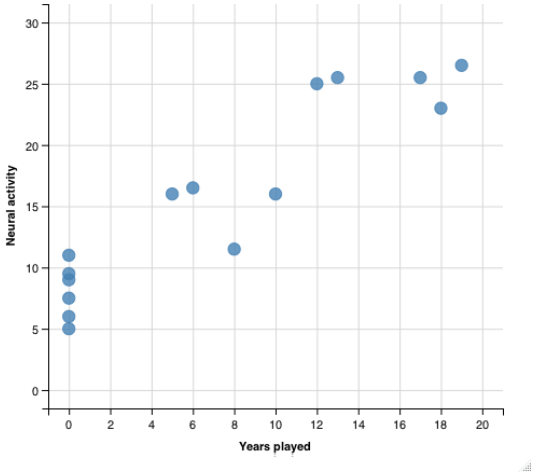
$$AIC = -2\left(-18.89556\right) + 2\left(2+1\right) = 6\left[\ln\left(2\pi\right) + 1 + \ln\left(\frac{191.0244}{6}\right)\right] + 2\left(2+1\right) = 43.79111$$

Scenario: Certain activities can affect the reorganization of the human central nervous system.

In one study, psychologists used magnetic source imaging (MSI) to measure neuronal activity in the brains of 9 violin players and 6 controls (those who have never played a stringed musical instrument) when the fingers on their left hands were exposed to mild stimulation. The researchers felt that stringed instrument players, who use the fingers on their left hand extensively, might show an increased amount of neuron activity. Shown below is a neuron activity index from the MSI along with the number of years each individual had been playing a stringed instrument:

Subject	Years played	Neural activity
1	0	5.0
2	0	6.0
3	0	7.5
4	0	9.0
5	0	9.5
6	0	11.0
7	5	16.0
8	6	16.5
9	8	11.5
10	10	16.0
11	12	25.0
12	13	25.5
13	17	25.5
14	18	23.0
15	19	26.5
mean	7.2	15.56667
std. dev	7.24273	7.782459
	correlation: r = 0.928	

Elbert, T., "Increased cortical representation of the fingers of the left hand in string players," Science, 270, 13 October, 305-307



Data: http://www.bradthiessen.com/html5/data/violin.csv

32. Our goal is to determine whether neural activity increases as the number of years playing the violin increases. Suppose we decide to conduct an ANOVA. How would we do this? What conclusions could we draw?

33. We want to determine if the years variable predicts or explains the neural variable. Write our competing models:

Full model: _____ Reduced model: _____

```
violin <- read.csv("http://www.bradthiessen.com/html5/data/violin.csv")          # Load data
model <- lm(neural ~ years, data=violin)            # Fit the linear model
summary(model)                                      # Summarize model
```

```
Residuals:
    Min      1Q  Median      3Q     Max
-4.8644 -2.3730  0.1614  2.3713  4.6471

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   8.3873     1.1149   7.523 4.35e-06 ***
years         0.9971     0.1110   8.980 6.18e-07 ***

Residual standard error: 3.009 on 13 degrees of freedom
Multiple R-squared:  0.8612,  Adjusted R-squared:  0.8505
F-statistic: 80.63 on 1 and 13 DF,  p-value: 6.178e-07
```

```
anova(model)          # ANOVA summary table
```

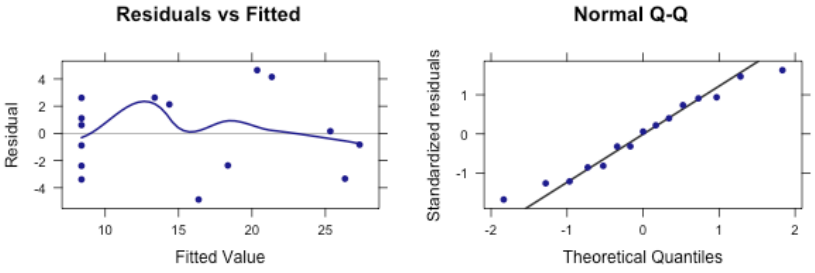
```
Analysis of Variance Table

Response: neural
          Df Sum Sq Mean Sq F value    Pr(>F)
years      1 730.21  730.21  80.633 6.178e-07 ***
Residuals 13 117.73    9.06
```

34. Interpret that output. Then, fill-in the following ANOVA summary table and verify the calculations. What conclusions can we make?

Source of variation	SS	df	MS	MSR (F)
Regression ($b_1 \mid b_0$)	_____	_____	_____	_____
Error	_____	_____	_____	(blank)
Total	_____	_____	MS_{total}	R^2 = _____

35. Replicate that MSR by calculating the omnibus F-statistic.

36. Explain what the following plots indicate with regards to the assumptions underlying linear regression.



Scenario: Some occupations are more prestigious than others (inspiring more respect or admiration). For example, most people would agree that a heart surgeon has a more prestigious occupation than a waitress. We're going to examine some factors that may influence the prestige of various occupations.

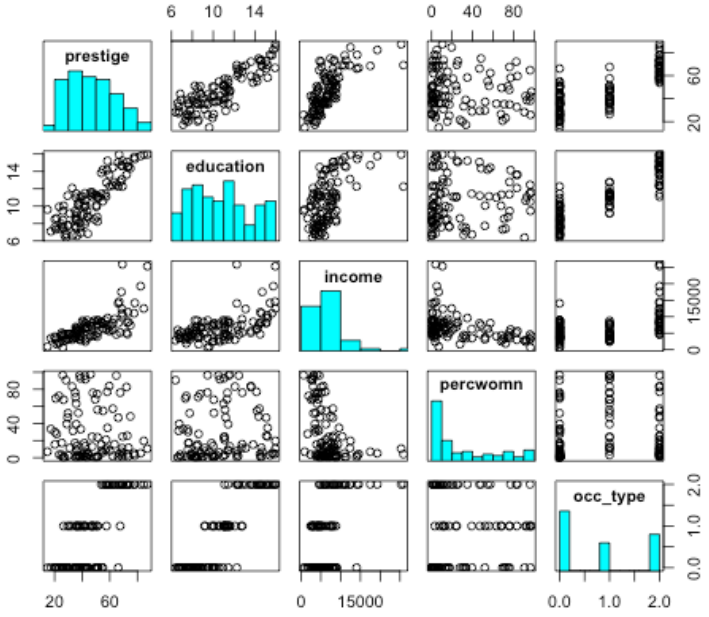
Data: http://www.bradthiessen.com/html5/data/prestige.csv
Source: Canada (1971). Census of Canada. Vol. 3, Part 6. Statistics Canada, 19-21.

prestige: Pineo-Porter Prestige score (a survey)
education: average years of education for people in that occupation
income: average income (1971 Canadian dollars) for people in that occupation
percwomn: % of workers in that occupation who are female
type: 0=blue collar, 1=white collar, 2=professional/technical/managerial

#	Title	Education	Income	%women	Type	Prestige
1	Physicians	15.96	25308	10.56	Professional	87.2
2	University Professors	15.97	12480	19.59	Professional	84.6
3	Lawyers	15.77	19263	5.13	Professional	82.3
4	Architects	15.44	14163	2.69	Professional	78.1
5	Physicists	15.64	11030	5.13	Professional	77.6
6	Psychologists	14.36	7405	48.28	Professional	74.9
7	Chemists	14.62	8403	11.68	Professional	73.5
8	Civil Engineer	14.52	11377	1.03	Professional	73.1
...	...	...	...	...	...	...
18	Medical Technicians	12.79	5180	76.04	White collar	67.5
19	Secondary Teachers	15.08	8034	46.8	Professional	66.1
...	...	...	...	...	...	...
26	Elementary Teachers	13.62	5648	83.78	Professional	59.6
...	...	...	...	...	...	...
98	Launderers	7.33	3000	69.31	Blue collar	20.8
99	Bartenders	8.5	3930	15.51	Blue collar	20.2
100	Elevator Operators	7.58	3582	30.08	Blue collar	20.1
101	Janitors	7.11	3472	33.57	Blue collar	17.3
102	Newsboys	9.62	918	7	(missing)	14.8
	Means	**10.738**	**6797.90**	**28.979**	**N/A**	**46.833**
	Std. Deviations	**2.7284**	**4245.92**	**31.725**	**N/A**	**17.204**

Correlations:

	education	income	%women	prestige
education	1.0000			
income	0.5776	1.0000		
%women	0.0619	-0.4411	1.0000	
prestige	0.8502	0.7149	-0.1183	1.0000



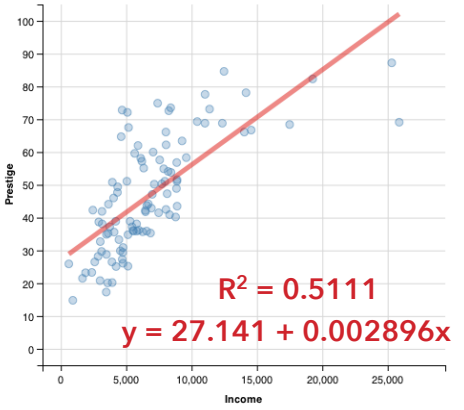
37. Before attempting to model prestige, I wanted to know if the 3 occupation types differed in prestige. Interpret these results:

```
  occ_type  n     mean       sd
1        0 49 36.08571 11.347320
2        1 23 42.24348  9.515816
3        2 30 67.90667  8.819255


Pairwise comparisons (Bonferroni)
  0       1
1 0.059   -
2 < 2e-16 4.4e-14
```

Source	SS	df	MS	MSR (F)
Type	17796	2	9733.576	92.40
Error	12100	99	105.336	$p = 2.2 \times 10^{-16}$
Total	29896	101	MS_{total}	$\eta^2 = 0.5953$

```
Bartlett test of homogeneity of variances
data:  prestige by occ_type
Bartlett's K-squared = 2.4469, df = 2, p= 0.2942
```

38. The relationship between prestige and income is displayed below. Interpret the coefficients of our model (which you could verify using the summary statistics on the previous page).



When I conducted this regression analysis in R, it gave me the following output:

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 27.141176  2.268e+00   11.97   <2e-16 ***
income       0.0028968  2.833e-04   10.22   <2e-16 ***
```

Interpret those p-values and the R-squared value.

39. Write out the full and reduced models. Complete the ANOVA summary table. How did we already know the MSR?

Full model: _____ Reduced model: _____

Source of variation	SS	df	MS	MSR (F)
Regression ($b_1 \mid b_0$)	15279	_____	15279	104.54
Error	14616	_____	146.16	(blank)
Total	29895	_____	MS_{total}	$R^2 = 0.5111$

40. Calculate and interpret the RMSE (root mean square error). What does it mean in this study?

41. Use the omnibus F-test to verify the F-statistic from the ANOVA summary table on the previous page.

42. With our least-squares regression line, we could predict the prestige of a job with an average income of $7000:

y = 27.141 + 0.002896(7000) = 47.41877

We know that prediction won't be perfectly accurate, so it might make sense to construct a confidence interval for our regression coefficients. Using R, I found the following confidence intervals:

```
                    2.5 %       97.5 %
(Intercept) 22.642116976 31.640235760
income       0.002334692  0.003458907
```

Interpret the 95% confidence interval for the slope of our regression line: (0.00233, 0.00345). Why does this <u>not</u> mean we're 95% confident that increasing an occupation's income by $1000 will be associated with a 2.33 – 3.45 increase in prestige.

We could use bootstrap methods or the following formula to construct a confidence interval for our regression line:

$$\hat{y} \pm \left(t_{n-2}^{\alpha/2}\right) s_{y|x} \sqrt{\frac{1}{n} + \frac{\left(x_0 - \bar{X}\right)^2}{(n-1)s_x^2}}$$

where

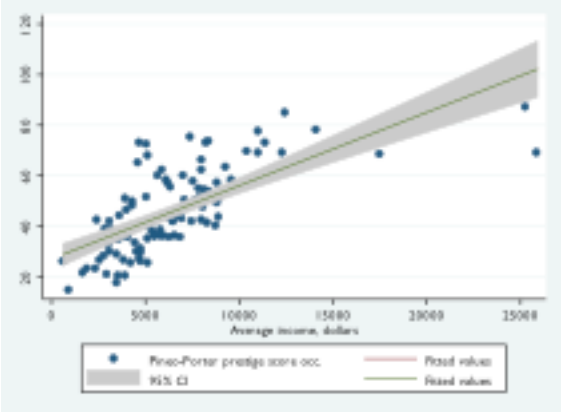
$$s_{y|x} = \sqrt{\frac{\left(y_i - \bar{Y}\right)^2}{(n-2)}} = \sqrt{\frac{SS_E}{n-2}} = \sqrt{\frac{\left(1-R^2\right)SS_Y}{n-2}} = \sqrt{\frac{\left(1-R^2\right)(n-1)s_y^2}{n-2}} = s_y\sqrt{1-R^2}\sqrt{\frac{n-1}{n-2}} = \sqrt{MSE}$$

A 95% confidence interval for the **<u>average prestige of all occupations with $7000 incomes</u>** is, then:

$$s_{y|x} = \sqrt{146.16} = 12.089$$

$$47.41877 \pm (1.984)(12.089)\sqrt{\frac{1}{102} + \frac{(7000 - 6797.90)^2}{(102-2)(4245.92)^2}} = 47.41877 \pm 2.38$$

43. Will this confidence interval have the same width (uncertainty) for all values of income? Explain.



The confidence interval is displayed on the plot to the right.

44. Based on our interpretation, this confidence interval didn't give us exactly what we wanted. We wanted an interval to predict the prestige of a single occupation that has a \$7000 income. The interval we calculated predicts the **average** prestige **all** occupations with \$7000 incomes.

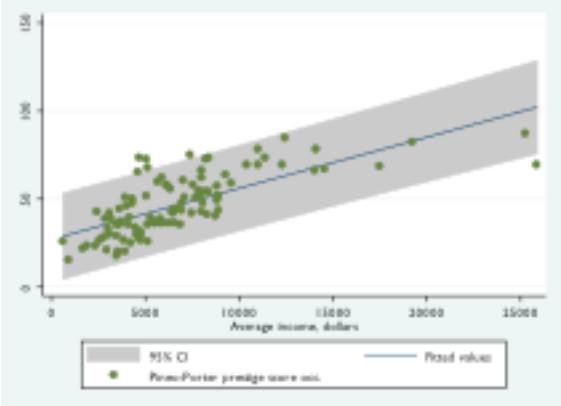
If we construct an interval to predict a single future observation, that interval must be **WIDER** **MORE NARROW** than our confidence interval.

To construct a prediction interval for our regression line, we use:

$$\hat{y}_i \pm \left(t_{n-2}^{\alpha/2}\right) s_{y|x} \sqrt{1+\frac{1}{n}+\frac{\left(x_0-\bar{X}\right)^2}{(n-1)s_x^2}}$$

$$47.41877 \pm (1.984)(12.09)\sqrt{1+\frac{1}{102}+\frac{(7000-6797.90)^2}{(102-1)(4245.92)^2}}$$

$$47.41877 \pm 24.10$$



The prediction interval is displayed to the right.

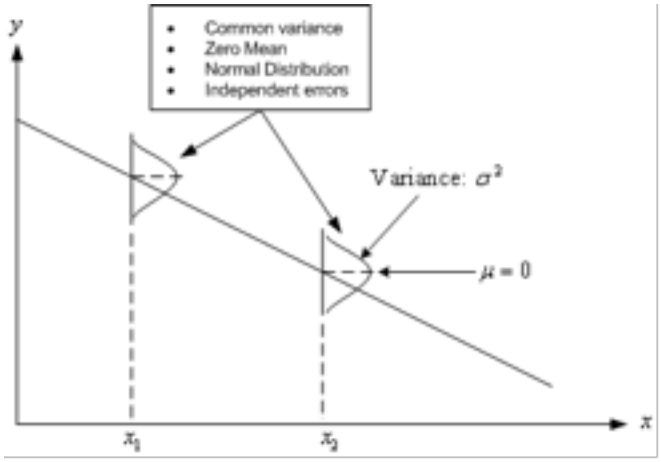
Obtaining confidence or prediction intervals in R is easy. Once you've specified your model, you apply the interval to new data using:

```
predict(model, newdata, interval="confidence")    predict(model, newdata, interval="predict")
```

The output, when our new data is a job with an income of \$7000, is:

```
     fit      lwr      upr                          fit      lwr      upr
47.41877 45.04112 49.79642                    47.41877 23.31552 71.52202
```

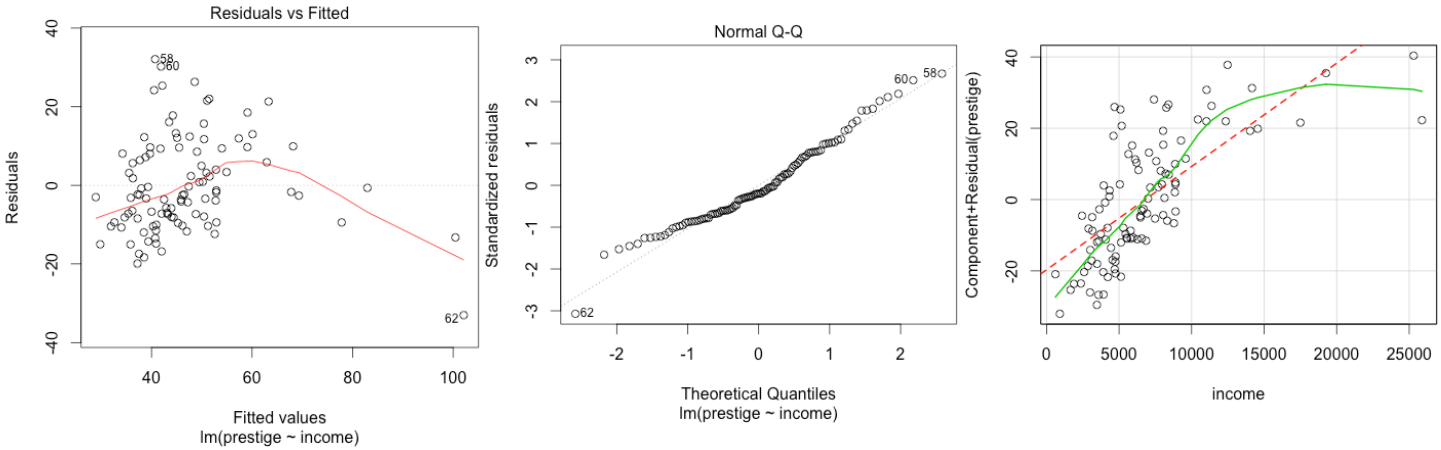
45. Recall the assumptions underlying regression. The diagram to the right attempts to display many of these assumptions.



Below, I've pasted output from the plots(model) command in R. Interpret these graphs and evaluate whether the assumptions appear to be satisfied in this case.

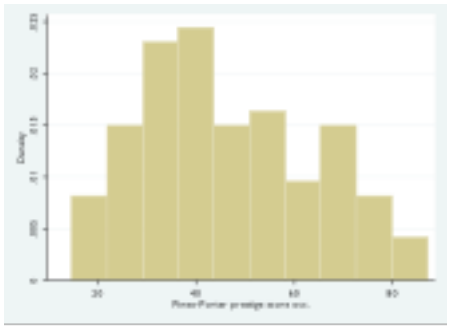
Non-constant error variance test:

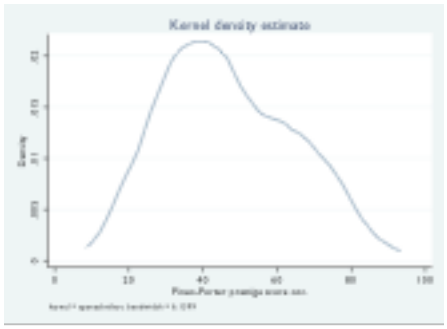
```
Variance formula: ~ fitted.values
Chisquare = 3.088455   Df = 1    p = 0.07885
```



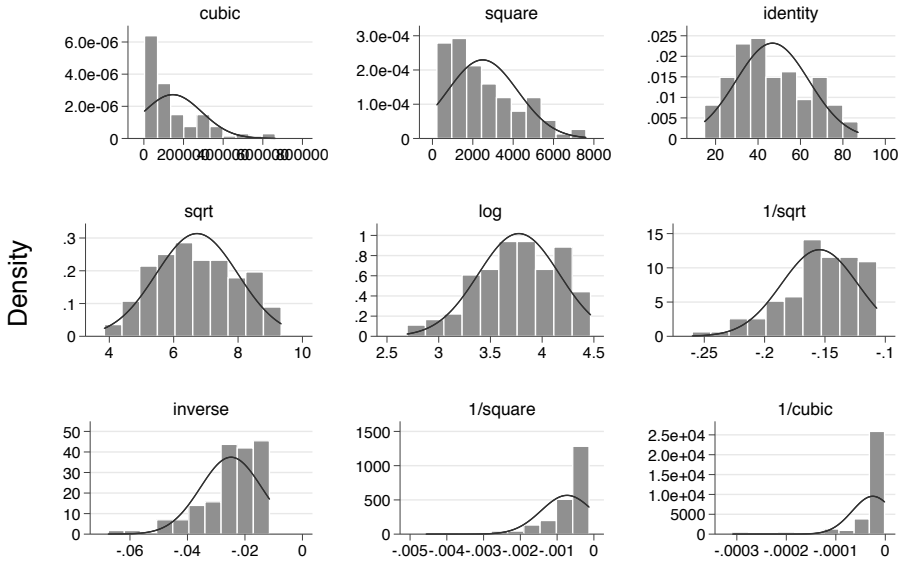
46. If we're worried about the normality and/or heteroscedasticity of our residuals, we have a few options.

a) We could transform our dependent variable to make it better approximate a normal distribution. Here's the distribution of our prestige data:





The figure on the next page displays the distributions we would get if we were to transform the prestige data using logarithms, exponents, or other transformations.



If a transformation makes the data better approximate a normal distribution, it may mean the residuals will better approximate a normal distribution. Be careful with this, though. Once you transform the data, your linear model may become much more difficult to interpret.

To learn more about transformations, check out http://onlinestatbook.com/2/transformations/tukey.html or http://onlinestatbook.com/2/transformations/box-cox.html

b) You could use *robust* regression methods (as we'll learn in a future activity). Interpret the following:

Ordinary least squares regression

```
                Estimate Std. Error t value Pr(>|t|)
(Intercept) 27.141176  2.268e+00   11.97   <2e-16 ***
income      0.0028968  2.833e-04   10.22   <2e-16 ***
```

Robust linear regression

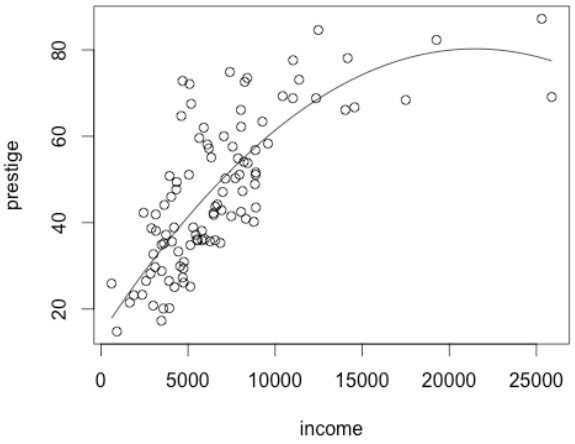
```
Robust linear regression                        Number of obs =     102
                                                F(  1,   100) =   48.28
                                                Prob > F      =  0.0000
                                                R-squared     =  0.5111
                                                Root MSE      =   12.09
------------------------------------------------------------------------------
             |               Robust
    prestige |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      income |   .0028968   .0004169     6.95   0.000     .0020697    .0037239
       _cons |   27.14118   2.886142     9.40   0.000     21.41515     32.8672
------------------------------------------------------------------------------
```

Quantile (median) regression

```
Quantile (Median) regression                    Number of obs =       102
  Raw sum of deviations     1447 (about 43.5)
  Min sum of deviations 954.6664                Pseudo R2     =    0.3402
------------------------------------------------------------------------------
    prestige |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      income |   .0030293   .0003073     9.86   0.000     .0024196    .0036391
       _cons |   23.94584   2.518318     9.51   0.000     18.94957    28.94211
------------------------------------------------------------------------------
```

c) You might also want to try to fit a model that isn't linear. We'll also learn some of these methods in the future.

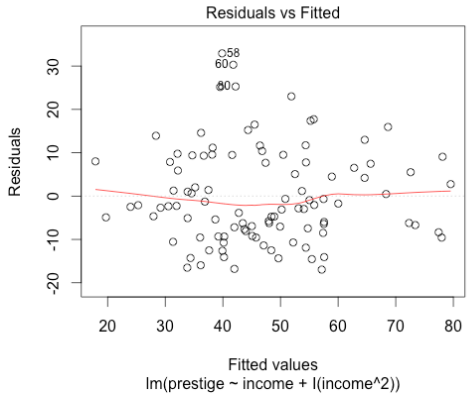
Model: $y = b_0 + b_1 x_1 + b_2 x_1^2 + e$



Best-fitting quadratic function:

$$y = 14.183 + 0.00615x - 0.000000143x^2$$

Below: Residuals vs. fitted plot:



Lowess (locally locally weighted scatterplot smoothing):