Linear Multiple Regression

School of Information Studies
Syracuse University

Animation Demo

install.packages("animation")

library(animation) # Load the animation library

par(mfrow=c(1,1)) # Configure the plot window as a single space

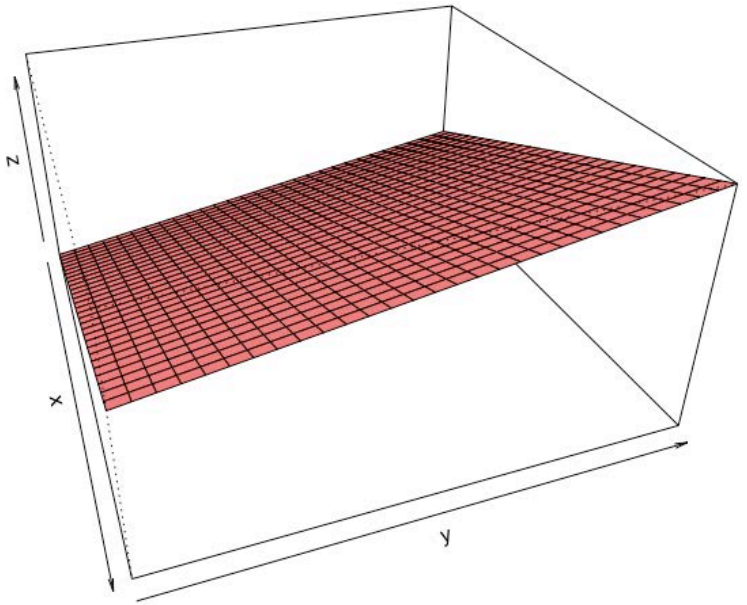
x <- seq(-10, 10, length= 30) # Create some X values

y <- x # Use the same for Y

f <- function(x,y) { z <- x*2 + y - 3 } # Z is a function of X and Y

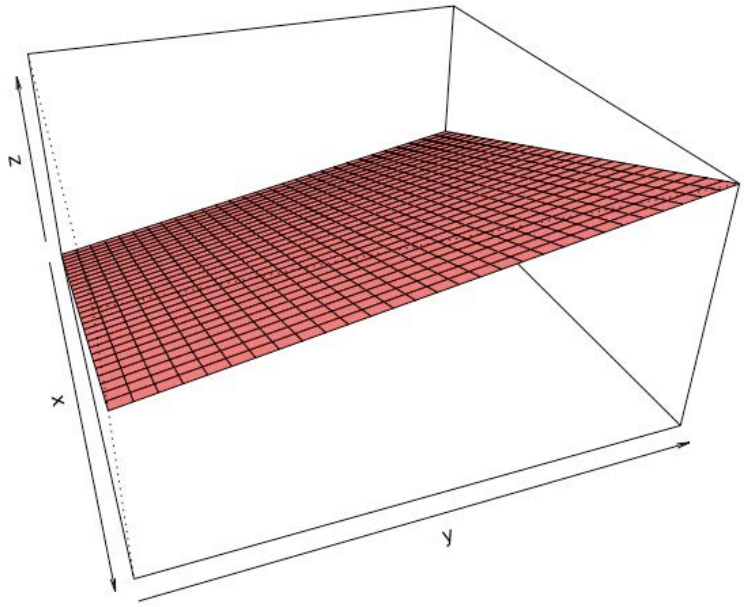
z <- outer(x,y,f) # Run the function on all X and Y

ani.record(reset = TRUE) # Empty the animation buffer



Animation Demo

```
# Change the perspective many times;
theta is the azimuth direction

for (i in 1:70) { persp(x, y, z, theta = i, phi =
30, expand = 0.5, col = “lightcoral”)
ani.record() # record the current frame}

oopts = ani.options(interval = 0.1) # Set
the delay between frames

ani.replay() # Replay the stored frames
```



Learning Topics for This Week

Review of Pearson Product Moment Correlation

Introducing regression

- Dependent and independent variables; criteria and predictors

Point clouds and best-fitting lines/planes

Least-squares criterion for fitting a line; sum of squared errors of prediction

Interpreting regression results

- B-weights (coefficients); R-Squared; Adjusted R-Squared; significance test

Learning Topics for This Week

Multicollinearity

Bayesian inference on coefficients and R-squared

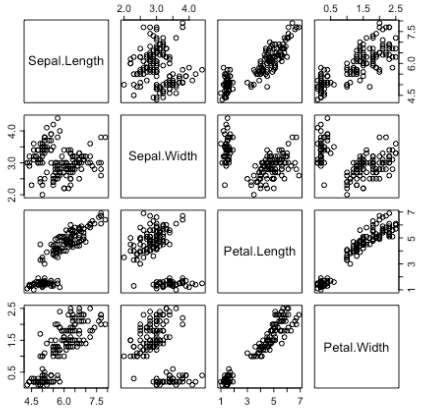
Learning goals: By the end of this class/week, students should be able to describe the use of multiple regression to find a best fitting line or plane in situations where one or more predictors predict a single metric criterion. Students should be able to conduct and interpret hypothesis tests on coefficients (B-weights) and overall R-squared.

Review of PPMC

Review of Correlation

The Pearson Product-Moment Correlation, aka "r", expresses the association between two metric variables on a scale of -1 to +1

Values of r near -1 or +1 are strong; values near 0 are weak



cor(iris[,1:4])

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
Sepal.Length	1.0000000	-0.1175698	0.8717538	0.8179411
Sepal.Width	-0.1175698	1.0000000	-0.4284401	-0.3661259
Petal.Length	0.8717538	-0.4284401	1.0000000	0.9628654
Petal.Width	0.8179411	-0.3661259	0.9628654	1.0000000

Introducing Regression

A Simple Model to Predict GPA

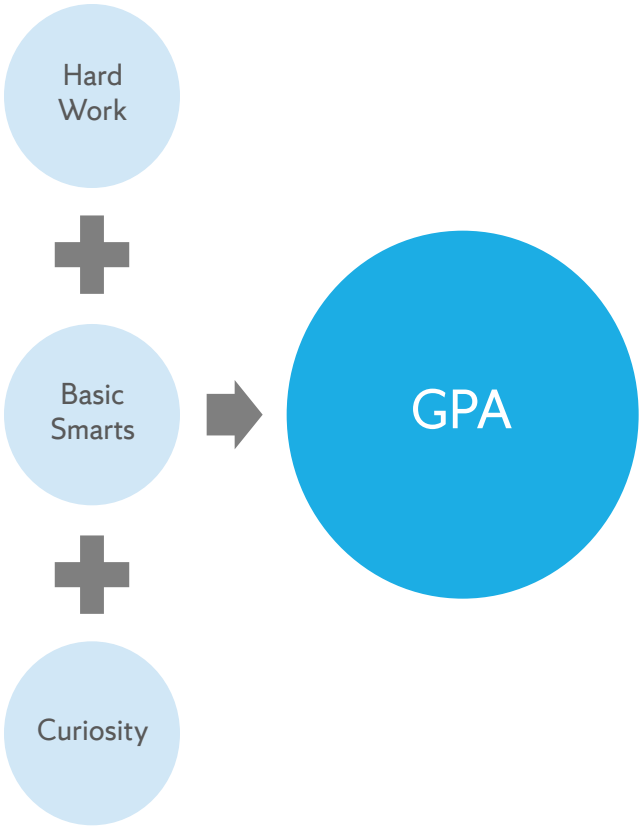
How could we predict a student's semester GPA using three pieces of information?

We could measure three predictors in advance, using tests or surveys with multiple students (e.g., n=120).

Then at the end of the semester we would have four pieces of information: the criterion (GPA) plus three predictors (hard work, basic smarts, & curiosity).

Using linear regression, we can calculate coefficients (B-weights) for each predictor to make an equation:

GPA = (B1 * HardWork) + (B2 * BasicSmarts) + (B3 * Curiosity)



Regression Terminology

Criterion/dependent variable: what we are trying to predict

Predictor/independent variable: one of the variables we use to predict the criterion; there are usually multiple predictors

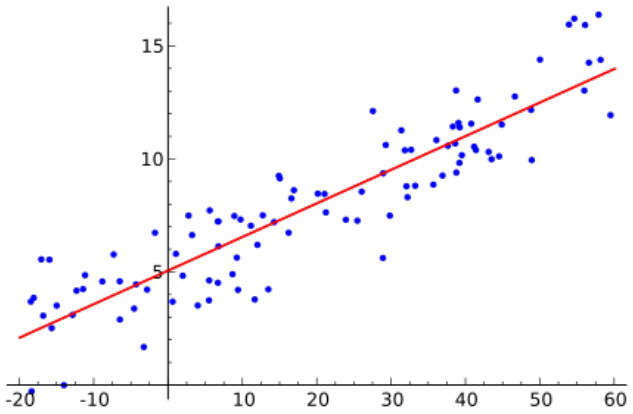
Coefficients/weights: the strength of prediction for each predictor; sometimes also called B-weights (or the standardized version is called a beta-weight)

Regression equation: the result of the regression analysis in the form of an algebraic equation

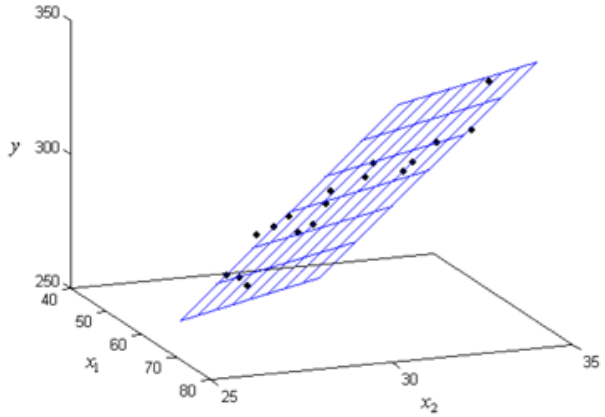
- $\hat{Y} = B_1X_1 + B_2X_2 + B_3X_3 + \ ...$
- Y-hat is the predicted Y, subscripts on Bs and Xs refer to predictor number

Point Clouds and Best-Fitting Lines

Visualizing Multiple Dimensions



Two dimensions: easy



Three dimensions: still good

?

Four dimensions?

Image credit: Sewaqu - Public Domain, https://commons.wikimedia.org/w/index.php?curid=11967659

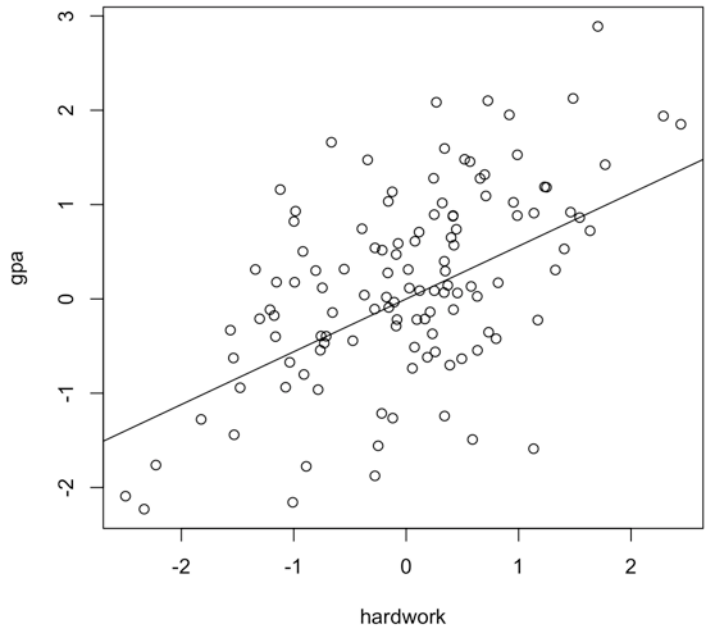
Finding the Best Fitting Line

The diagram at left visualizes a cloud of points representing scores on our "hardwork" predictor and our GPA criterion/outcome.

The other two predictors are still in the data, we're just ignoring them for the moment.

We can take a guess at a straight line that we can draw through the points.

How can we decide the best slope and intercept for the line? What does it mean to have a good fit to these points?



Least Squares Criterion

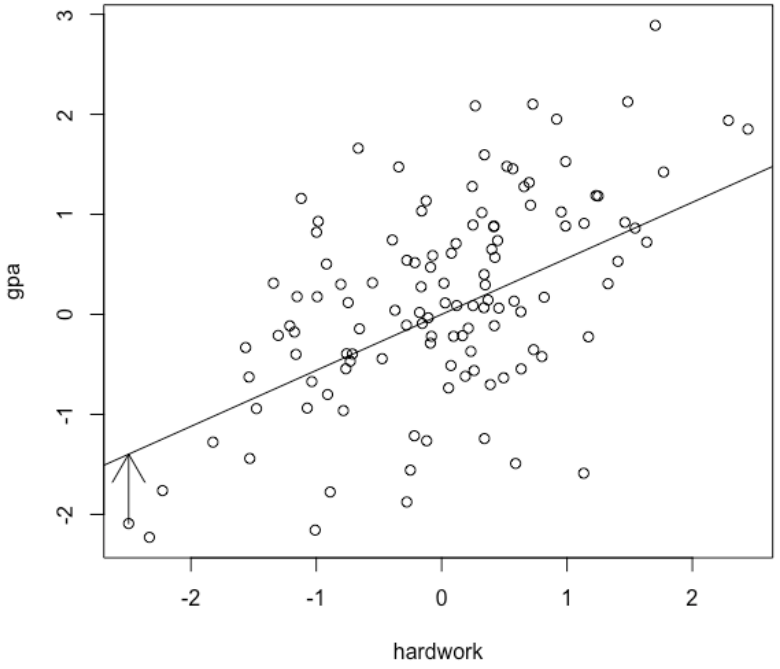
Prediction Errors: The Least Squares Criterion

Each point on the line represents our prediction of Y given a certain value of X.

To the extent that an observed value does not fall on the line, we have a prediction error: the vertical (Y-axis) distance between the point and the line. See the arrow at the lower left.

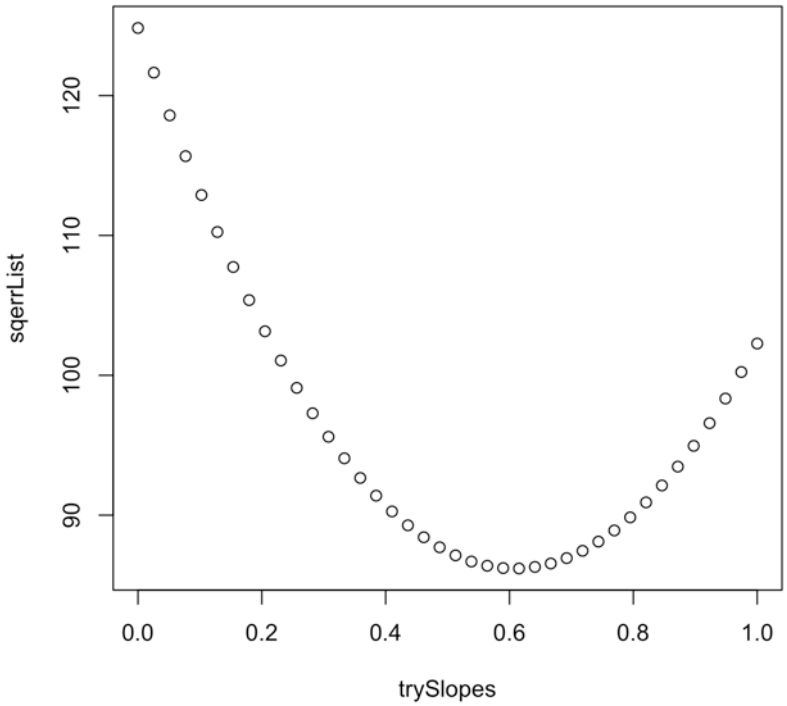
For the best fitting line, all of the positive errors (too high) and all of the negative errors (too low) will sum to zero.

As a result, it is also true that the sum of the squared errors of prediction from the best fitting line will be the smallest possible value.



Finding the Best Fitting Line (Minimizing the Sum of Squared Errors) With 40 Guesses

```
calcSQERR <- function(dv, iv, slope)
{ (dv - (iv*slope))^2 }

sumSQERR <- function(slope)
{ sum(calcSQERR(gpa, hardwork, slope)) }

trySlopes <- seq(from=0, to=1, length.out=40)

sqerrList <- sapply(trySlopes, sumSQERR)

plot(trySlopes, sqerrList)
```



Interpreting Regression Results

Interpreting Regression Results From lm()

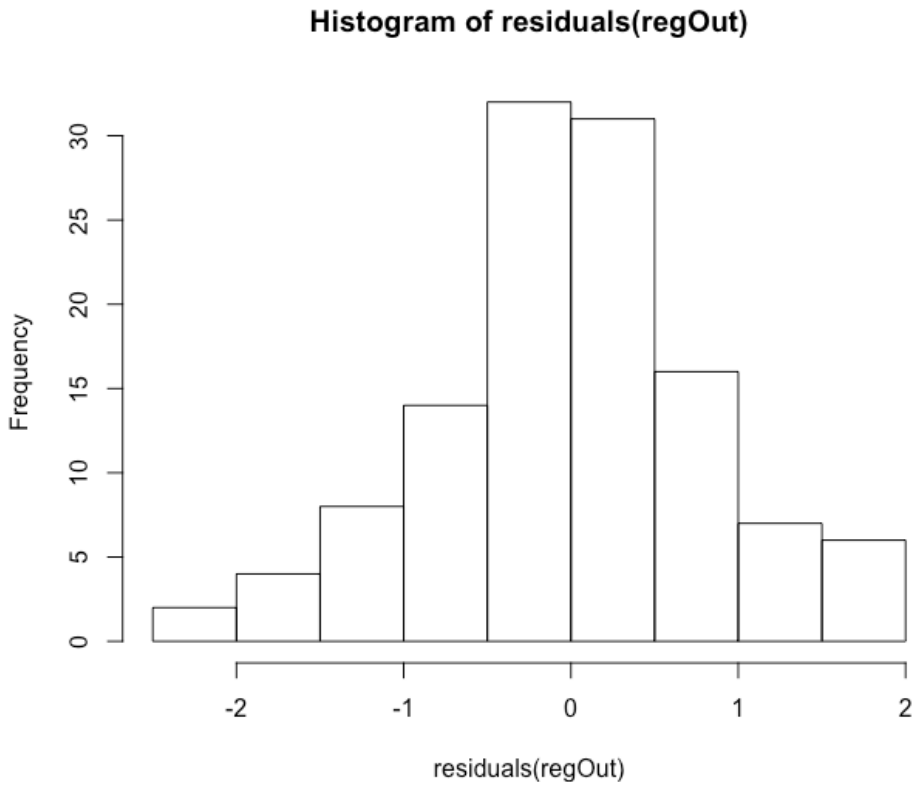
```
Residuals:
     Min       1Q   Median       3Q      Max
-2.43563 -0.47586  0.00028  0.48830  1.90546
Coefficients:
             Estimate Std. Error t value   Pr(>|t|)
(Intercept)  0.15920  0.07663   2.078     0.0399 *
hardwork     0.60700  0.08207   7.396     2.23e-11 ***
Residual standard error: 0.8394 on 118 degrees of freedom
Multiple R-squared: 0.3167, Adjusted R-squared: 0.311
F-statistic: 54.7 on 1 and 118 DF, p-value: 2.227e-11
```

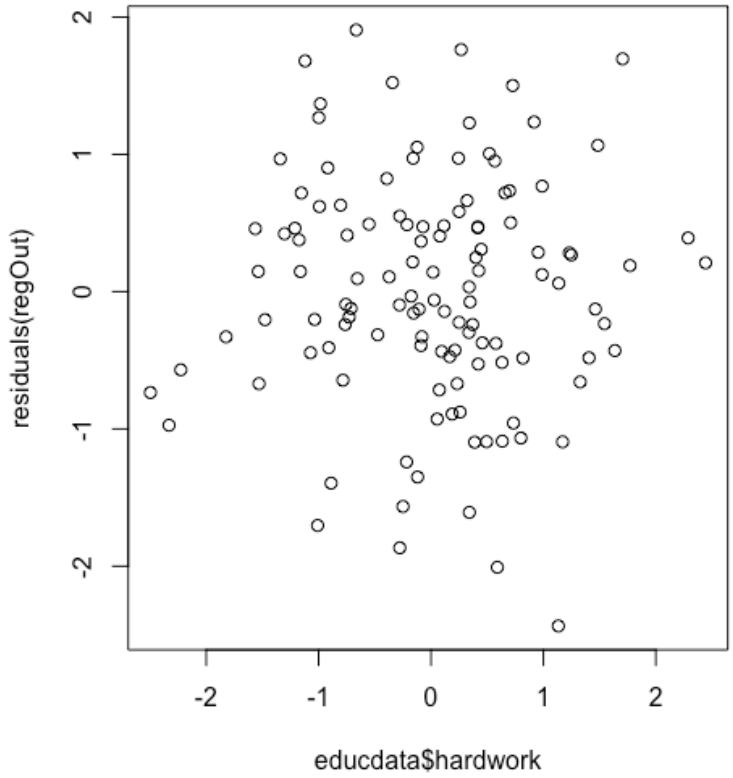
Residuals are the same thing as prediction errors. These first three lines summarize their distribution. They should be normally distributed with a median near zero.

The two coefficients are the Y-intercept (first line) and the slope/B-weight on the predictor (second line). For each, the estimate is the statistical value of interest. The std. error estimates variability of the underlying sampling distribution. Together, these two values to calculate "t" and an associated null hypothesis test.

The final block shows summary statistics, including the R-squared and a null hypothesis test (using F).

Residuals Are Errors of Prediction





A Note on Standardized Coefficients

Throughout Chapter 8 and these slides we have worked with unstandardized "B" weights that are calibrated according to the original metric of the predictor.

A B-weight on a predictor called "length" will be very different depending upon whether length is calibrated in centimeters versus kilometers.

In social science research it is common to only report "standardized" weights, known as beta-weights, in order to be able to examine the relative strength of a predictor without reference to its original scale.

Beta weights are on a scale of -1 to +1 (analogously to values of r).

```
# Run any regression model in lm()

regOut <- lm(weight ~ length + height, data=testdata)

summary(regOut)

# Use the lm.beta() function in package QuantPsyc

# to produce beta weights

install.packages("QuantPsyc")

library("QuantPsyc")

lm.beta(regOut)
```

What's Adjusted R-Squared?

$$R^2_{adj} = 1 - \left(\frac{SS_{res}/(n-p-1)}{SS_{tot}/(n-1)}\right)$$

When we calculate *adjusted* R-squared we use *unbiased estimators* for the estimates of total variance and "error" variance.

The total variance in the dependent variable is the sum of squares divided by n-1 degrees of freedom in the denominator.

The degrees of freedom for the error (residual) variance is n-p-1, in other words, the sample size, minus the number of predictors, minus one.

A Model With All Three Predictors: regOut3 <- lm(gpa ~ hardwork + basicsmarts + curiosity, data=educdata)

```
summary(regOut3)

Call:

lm(formula = gpa ~ hardwork + basicsmarts + curiosity, data = educdata)

Residuals:

     Min       1Q   Median       3Q      Max
-1.02063 -0.37301  0.00361  0.31639  1.32679

Coefficients:

            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.08367    0.04575   1.829     0.07 .
hardwork     0.56935    0.05011  11.361  <2e-16 ***
basicsmarts  0.52791    0.04928  10.712  <2e-16 ***
curiosity    0.51119    0.04363  11.715  <2e-16 ***
---

Residual standard error: 0.4978 on 116 degrees of freedom

Multiple R-squared:  0.7637, Adjusted R-squared:
0.7576 F-statistic:  125 on 3 and 116 DF, p-value: < 2.2e-16
```

Residuals look good: median near zero and symmetric

F statistic tests the null hypothesis that R-squared == 0

Notice the small penalty for having 3 predictors

Degrees of Freedom in Regression

When examining regression output, the F-test provides a significance test that tests the null hypothesis that R-squared is 0

Sometimes people refer to this as the "omnibus" test, because it tests "all" of the regression results as a whole

Just as we saw before with ANOVA, F is a family of statistical distributions whose shape varies based on two different values for degrees of freedom; these are sometimes referred to as:

- Numerator and denominator (algebraic)
- Effect and error (generic)
- Between subjects and within subjects (ANOVA)
- Regression and residual (regression)

Degrees of Freedom in Regression

The first df in regression is one less than the number of parameters to be estimated (including the intercept); therefore, many people say that the first df is equal to the number of predictors

The second df is the total number of observations minus one and also minus the number of predictors

Significance Tests on Predictors

T-value is calculated from the B-weight and its standard error

Significance test of the null hypothesis that B == 0

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.08367	0.04575	1.829	0.07 .
hardwork	0.56935	0.05011	11.361	<2e-16 ***
basicsmarts	0.52791	0.04928	10.712	<2e-16 ***
curiosity	0.51119	0.04363	11.715	<2e-16 ***

Residual standard error: 0.4978 on 116 degrees of freedom

Multiple R-squared: 0.7637; Adjusted R-squared: 0.7576

F-statistic: 125 on 3 and 116 DF, p-value: < 2.2e-16

Multicollinearity

Multicollinearity

```
> x <- rnorm(100)
> y <- rnorm(100)
> z <- x
> testDF <- data.frame(x,y,z)
> summary(lm(y ~ x + z, data=testDF))
Residuals:
     Min       1Q   Median       3Q      Max
-2.03056 -0.62523  0.05375  0.56941  2.45931
Coefficients: (1 not defined because of singularities)
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.02978    0.09536   0.312   0.7555
x           -0.24523    0.11049  -2.220   0.0288*
z                NA         NA      NA       NA
Residual standard error: 0.9491 on 98 degrees of freedom
Multiple R-squared:  0.04786, Adjusted R-squared:  0.03815
F-statistic: 4.926 on 1 and 98 DF,  p-value: 0.02876
```

lm() uses matrix algebra to calculate the coefficients for the regression equation, using the covariance matrix

The covariance matrix becomes "singular" (its determinant is 0) if one or more of the variables can be expressed as a linear combination of other variables in the matrix

Practically, this means that if two or more predictors are too highly correlated, it will be impossible to create the lm() model

In the model at the left, we try to use x and z together as predictors, but they are perfectly correlated, creating a singularity

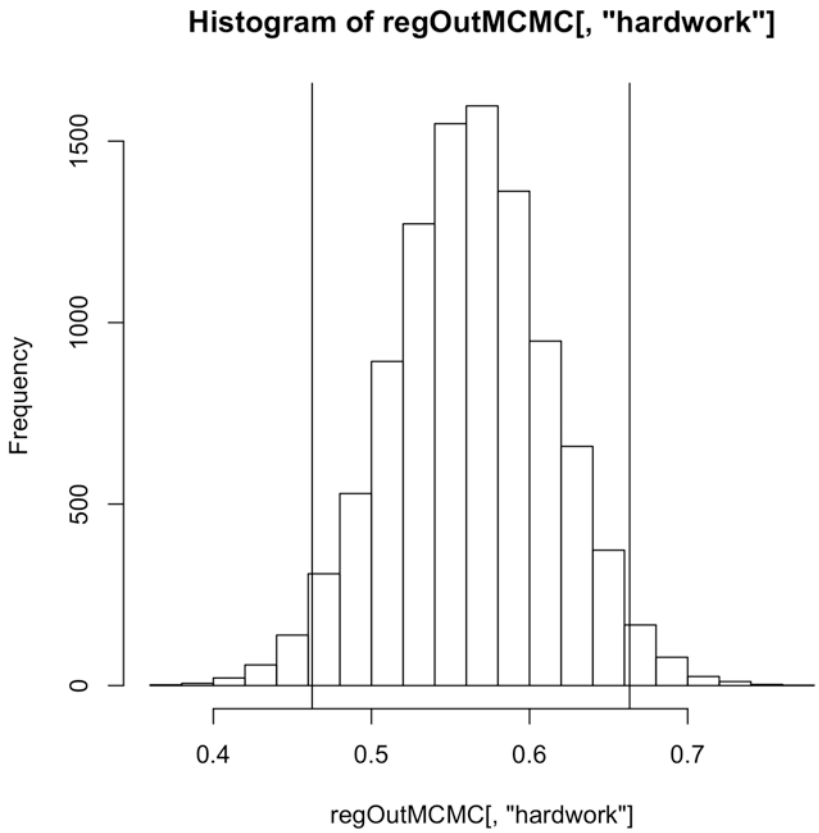
Bayesian Inference for Regression

Bayesian Approach to Regression

As with previous Bayesian tests, we use the Markov Chain Monte Carlo technique to generate many posterior estimates of each parameter (e.g., the coefficient/B-weight for hardwork).

The lmBF() procedure uses the same syntax as lm(), except for the addition of "posterior=TRUE, iterations=10000" to generate the MCMC output.

The histogram at right shows 10,000 MCMC estimates of the coefficient, with vertical lines marking the extent of the 95% Highest Density Interval (HDI).



Mean MCMC Estimates

1. Empirical mean and standard deviation for each variable, plus standard error of the mean:

Mean of Y, not the intercept

	Mean	SD	Naive SE	Time-series SE
mu	0.1621	0.04616	0.0004616	0.0004616
hardwork	0.5628	0.05073	0.0005073	0.0005161
basicsmarts	0.5220	0.05003	0.0005003	0.0004971
curiosity	0.5055	0.04384	0.0004384	0.0004478
sig2	0.2545	0.03427	0.0003427	0.0003616
g	1.6716	7.29839	0.0729839	0.0729839

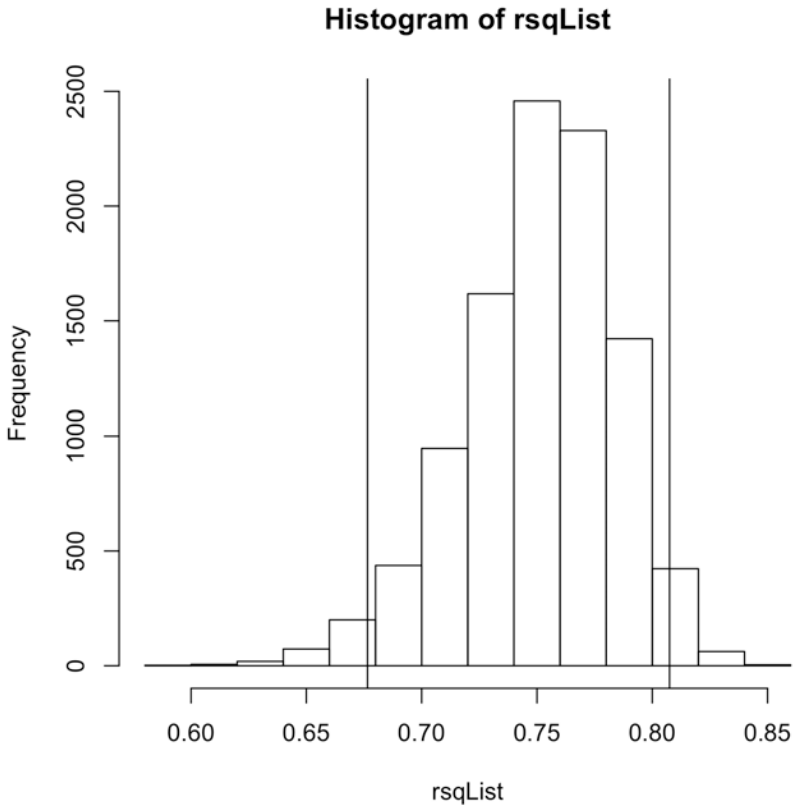
HDI Overview Table

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
mu	0.07282	0.1311	0.1618	0.1930	0.2542
hardwork	0.46172	0.5293	0.5627	0.5962	0.6641
basicsmarts	0.42375	0.4890	0.5224	0.5550	0.6198
curiosity	0.41853	0.4761	0.5053	0.5347	0.5909
sig2	0.19633	0.2304	0.2515	0.2753	0.3300
g	0.26580	0.5676	0.9322	1.6316	6.5155

HDI Display for R-Squared

```
# sig2 from lmBF() estimates error variance

# in the prediction model

rsqList <- 1 - (BFregOut2["sig2"] / var(gpa))

mean(rsqList) # Overall mean R-squared is 0.75

hist(rsqList) # Show a histogram

# Draw boundaries of the 95% HDI

abline(v=quantile(rsqList,c(0.025)), col="black")
abline(v=quantile(rsqList,c(0.975)), col="black")
```



Integrating the Bayes Factor

Bayes factor analysis

---

[1] hardwork + basicsmarts + curiosity : 7.885849e+32 ±0%

Against denominator:

Intercept only

Bayes factor type: BFlinearModel, JZS

Integrating three types of evidence for hypothesis testing:

1. Using the "frequentist" (traditional) null hypothesis test, we reject the null hypothesis for all three predictors as well as the overall R-squared
2. The Highest Density Intervals (HDIs) from the MCMC output show estimates for the coefficients and R-squared that concur with the frequentist model
3. The Bayes Factor overwhelmingly favors a model that includes the three predictors

Putting It All Together

We tested a model of academic achievement that used three variables to predict GPA: hard work, basic smarts, and curiosity. A Bayesian analysis of this model showed a mean posterior estimate for R-squared of 0.75, with the highest density interval ranging from roughly 0.68 to 0.82. The traditional analysis confirmed this result with a slightly more optimistic R-squared of 0.76. The F-test on this value was $F(3, 116)=125.0$, $p<.001$, so we reject the null hypothesis that R-squared was equal to zero. All three predictors were also significant with B-weights of 0.57 (hardwork), 0.53 (basicsmarts), and 0.51 (curiosity). The Bayes factor of 7.9e+32 was strongly in favor of the three predictor model (in comparison with an intercept-only model).