Lab 2

Simple Linear Regression

Equations and Interpretations of Linear Regression

First let's load some data to work with: We'll use the "gifted" dataset from the openintro package.The description for this dataset says:

An investigator is interested in understanding the relationship, if any, between the analytical skills of young gifted children and the following variables: father's IQ, mother's IQ, age in month when the child first said 'mummy' or 'daddy', age in month when the child first counted to 10 successfully, average number of hours per week the child's mother or father reads to the child, average number of hours per week the child watched an educational program on TV during the past three months, average number of hours per week the child watched cartoons on TV during the past three months. The analytical skills are evaluated using a standard testing procedure, and the score on this test is used as the response variable.

Data were collected from schools in a large city on a set of thirty-six children who were identified as gifted children soon after they reached the age of four.

```r
# load the package
require(openintro)

# load the data
data("gifted")

# look at the first few rows
head(gifted)
```

```
##   score fatheriq motheriq speak count read edutv cartoons
## 1   159      115      117    18    26  1.9  3.00     2.00
## 2   164      117      113    20    37  2.5  1.75     3.25
## 3   154      115      118    20    32  2.2  2.75     2.50
## 4   157      113      131    12    24  1.7  2.75     2.25
## 5   156      110      109    17    34  2.2  2.25     2.50
## 6   150      113      109    13    28  1.9  1.25     3.75
```

```r
# Run and save simple linear regression using lm()
im <- lm(score ~ motheriq, data=gifted)
```

Population Regression Model

Conditional mean
$\mu_Y = E[Y|X] = \beta_0 + \beta_1 X$
Individual
$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$, for $i = 1, 2, ..., N$ with $\epsilon \sim N(0, \sigma^2)$

Since we do not know the true values of β_0 and β_1, we estimate them from a sample and call our estimates $\hat{\beta}_0$ and $\hat{\beta}_1$. We compute these estimates using the ordinary least squares method. Using these estimated coefficients, we can also compute other estimates.

Estimated Regression Model from Sample

Estimated conditional mean
$\hat{Y} = estimated\ E[Y|X] = \hat{\beta}_0 + \hat{\beta}_1 X$

This is "the expectation of Y given X" or "the expectation of Y conditional on X." When we evaluate this model for a particular value of X, we are estimating the **average** value of dependent variable Y for individuals with a certain value of independent variable X in the population. Let's look at an example using last week's dataset.

```
summary(im)
```

```
##
## Call:
## lm(formula = score ~ motheriq, data = gifted)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.3569 -2.7497  0.1157  2.8794  8.7091
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 111.0930    11.8567   9.370 6.02e-11 ***
## motheriq      0.4066     0.1002   4.058 0.000274 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.856 on 34 degrees of freedom
## Multiple R-squared:  0.3263, Adjusted R-squared:  0.3065
## F-statistic: 16.47 on 1 and 34 DF,  p-value: 0.000274
```

Questions

1) What is our estimate of β_0 (that is, what value did we obtain for $\hat{\beta}_0$)? How do we interpret this value in terms of our variables of test score and mother's IQ?

```
im$coefficients[1]
```

```
## (Intercept)
##     111.093
```

We expect mothers with an IQ of 0 to have children who score an **average** of 111 points on this test. The interpretation of the intercept coefficient often doesn't make much practical or realistic sense in context (e.g., age of 0, weight of 0, test score of 0).

2) What is our estimate of β_1 (that is, what value did we obtain for $\hat{\beta}_1$)? How do we interpret this value in terms of our variables of test score and mother's IQ?

```
im$coefficients[2]
```

```
##  motheriq
## 0.4065946
```

For two groups of mothers whose average IQ differs by 1 point, we expect the mothers with higher IQs to have children whose **average** score on this test is about 0.4 points greater.

3) According to our model, what is the average score we estimate for mothers with an IQ of 100 (that is, what is the estimate of $E[Y|X = 100]$)?

```
# Method 1
im$coefficients[1] + im$coefficients[2]*100
```

```
## (Intercept)
##    151.7524
```

```
# Method 2
predict(im, data.frame(motheriq=100))
```

```
##        1
## 151.7524
```

151.75 points

Confidence Intervals and Predictions in R

Confidence Interval of Regression Coefficients

```
confint(im, level = .99)
```

```
##                  0.5 %      99.5 %
## (Intercept) 78.7432212 143.4426892
## motheriq     0.1332334   0.6799559
```

```
# 95% is the default. Otherwise, adjust the level argument
```

Confidence Interval of an Estimated Conditional Average, $\hat{Y}$

```
predict(im, data.frame(motheriq=100), interval = "confidence")
```

```
##        fit      lwr      upr
## 1 151.7524 147.8297 155.6752
```

Prediction Interval (for a new individual data point)

```
predict(im, data.frame(motheriq=100), interval = "prediction")
```

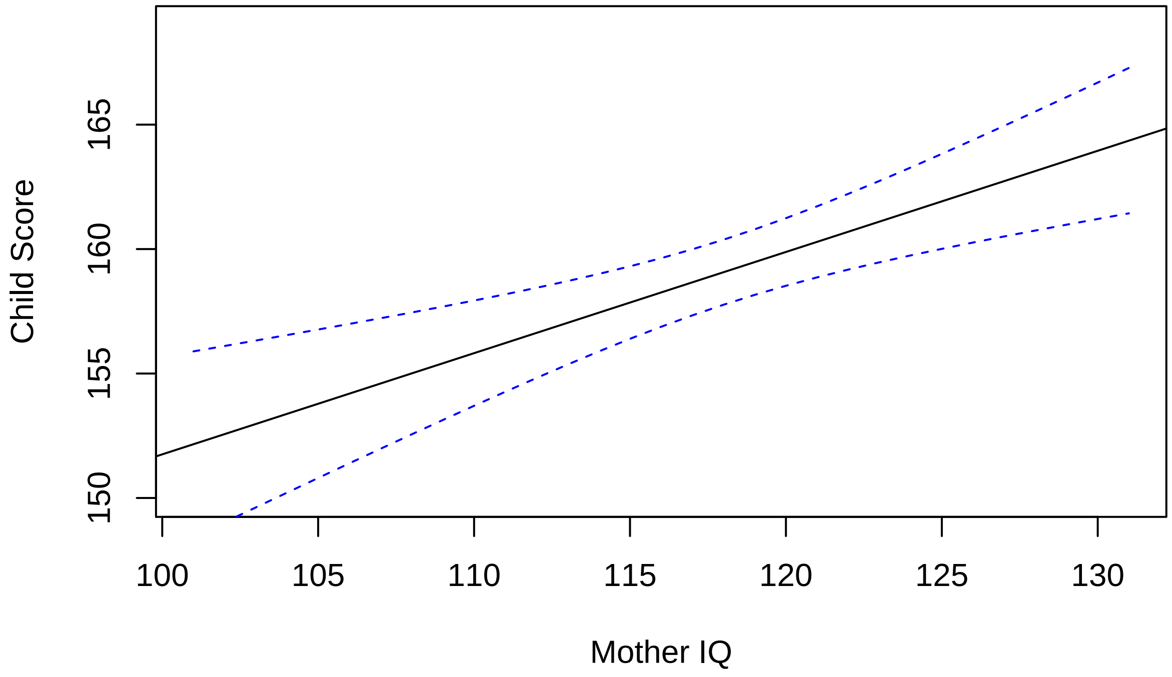
```
##        fit      lwr      upr
## 1 151.7524 142.9896 160.5153
```

Think about: Why is the prediction interval larger than the confidence interval?

Finally, let's plot the regression line with lines on either side to show the 95% confidence interval around the regression line.

```
#plot a blank scatter plot with the correct dimensions
plot(gifted$motheriq, gifted$score, type="n", xlab="Mother IQ", ylab="Child Score")

#plot the regression line
abline(im)

#create a vector of iqs incrementing by 1, from the min to max motheriq value
new_iqs=seq(min(gifted$motheriq, na.rm=T), max(gifted$motheriq, na.rm=T), by=1)

#get a matrix of the bounds on a 95% confidence interval for predicted scores based on the new iqs vect
bounds <- predict(im, data.frame(motheriq=new_iqs), interval = "confidence")

#add lines that connect the points that we estimated using the confidence intervals
```

```r
lines(new_iqs, bounds[,2], lty=2, col=4)
lines(new_iqs, bounds[,3], lty=2, col=4)
```



Centering, Standardizing, and Re-scaling Variables

Sometimes, we may want to center, standardize, or re-scale a variable in order to make the regression results more interpretable. The process for running the regression is the same as before, but now we have to be careful about the scale we use when interpreting the results.

First, let's center the motheriq variable:

```r
# Center the motheriq variable, call this new variable motheriq0
gifted$motheriq0 <- gifted$motheriq-mean(gifted$motheriq)

# Re-run the regression with motheriq0
im2 <- lm(score ~ motheriq0, data=gifted)
summary(im2)
```

```
##
## Call:
## lm(formula = score ~ motheriq0, data = gifted)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.3569 -2.7497  0.1157  2.8794  8.7091
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 159.1389     0.6426 247.640  < 2e-16 ***
## motheriq0     0.4066     0.1002   4.058 0.000274 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 3.856 on 34 degrees of freedom
## Multiple R-squared:  0.3263, Adjusted R-squared:  0.3065
## F-statistic: 16.47 on 1 and 34 DF,  p-value: 0.000274
```

Now, the interpretation of the intercept is: We expect mothers with average IQ in this sample to have children who earn an average score of 159.1 points on this test. The estimate for beta1 didn't change because centering does not affect the impact of the dependent variable on the independent variable. So the interpretation of beta1 stays the same.

Next, let's standardize the motheriq variable so that every value of motheriq in the dataset is actually each mother's Z-score with respect to IQ (number of standard deviations that they are away from the mean).

```
# Standardize the motheriq variable, call this new variable motheriq_st
gifted$motheriq_st <- gifted$motheriq0/sd(gifted$motheriq0)

# Re-run the regression with motheriq_st
im3 <- lm(score ~ motheriq_st, data=gifted)
summary(im3)
```

```
##
## Call:
## lm(formula = score ~ motheriq_st, data = gifted)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.3569 -2.7497  0.1157  2.8794  8.7091
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 159.1389     0.6426 247.640  < 2e-16 ***
## motheriq_st   2.6449     0.6517   4.058 0.000274 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.856 on 34 degrees of freedom
## Multiple R-squared:  0.3263, Adjusted R-squared:  0.3065
## F-statistic: 16.47 on 1 and 34 DF,  p-value: 0.000274
```

This doesn't change the interpretation of the intercept, but now we interpret beta1 as representing an expected change in *standard deviations*, not IQ points: For two groups of mothers whose average IQ differs by 1 *standard deviation*, we expect the mothers with higher IQs to have children whose **average** score on this test is about 2.6 points greater.

Estimating beta0 and beta1 using matrices

Reminders about creating/manipulating matrices in R

```
# R wants the data to be entered by columns starting with column one
# 1st arg: c(2,3,-2,1,2,2) the values of the elements filling the columns
# 2nd arg: 3 the number of rows
# 3rd arg: 2 the number of columns

A <- matrix(c(2,3,-2,1,2,2), 3, 2)
A
```

```
##      [,1] [,2]
## [1,]    2    1
## [2,]    3    2
## [3,]   -2    2
```

```
B <- matrix(c(2,-2,1,2,3,1), 2, 3)

# Matrix multiplication
A %*% B
```

```
##      [,1] [,2] [,3]
## [1,]    2    4    7
## [2,]    2    7   11
## [3,]   -8    2   -4
```

```
# Transpose of a Matrix
# Sometimes written as A' ("A prime")
AT <- t(A)
AT
```

```
##      [,1] [,2] [,3]
## [1,]    2    3   -2
## [2,]    1    2    2
```

```
# Multiplying a matrix by its transpose produces a square matrix
AAT <- A %*% AT
dim(AAT)
```

```
## [1] 3 3
```

```
# Matrix Inverses
C <- matrix(c(4,4,-2,2,6,2,2,8,4), 3, 3)
CI <- solve(C)
```

Estimated regression equation: $Y = XB$, where Y is a nx1 matrix of all the Y values, X is a nx2 matrix of a column of 1's and the X values, and B is a 2x1 matrix of beta_0 and beta_1.

We can manipulate these matrices to have the equation solved for B: $B = (X'X)^{-1}(X'Y)$

See this link for algebra: http://faculty.cas.usf.edu/mbrannick/regression/regma.htm

```
Y <- as.matrix(gifted$score)
X <- matrix(c(rep(1, nrow(gifted)), gifted$motheriq), nrow(gifted), 2)

B <- solve(t(X) %*% X) %*% t(X) %*% Y
B
```

```
##             [,1]
## [1,] 111.0929552
## [2,]   0.4065946
```

```
lin_mod <- lm(score ~ motheriq, data = gifted)
lin_mod$coefficients
```

```
## (Intercept)    motheriq
## 111.0929552   0.4065946
```

Matrix algebra is extremely useful in solving for the coefficients in a multiple linear regression (more than one X variable).

More Useful Computational Functions in R

Mean, Variance, Covariance, and Correlation

```
# Basic computations in R
mean(gifted$motheriq)
```

```
## [1] 118.1667
```

```
var(gifted$motheriq)
```

```
## [1] 42.31429
```

```
cov(gifted$motheriq, gifted$score)
```

```
## [1] 17.20476
```

```
cor(gifted$motheriq, gifted$score)
```

```
## [1] 0.571242
```

Standard Error of Regression Coefficient

```
# Variance-Covariance Matrix
#variance of each and then covariance on diagonal
vcov(im)
```

```
##             (Intercept)    motheriq
## (Intercept)  140.581166 -1.18619070
## motheriq      -1.186191  0.01003829
```

```
# Standard error of beta_1
summary(im)
```

```
##
## Call:
## lm(formula = score ~ motheriq, data = gifted)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.3569 -2.7497  0.1157  2.8794  8.7091
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 111.0930    11.8567   9.370 6.02e-11 ***
## motheriq      0.4066     0.1002   4.058 0.000274 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.856 on 34 degrees of freedom
## Multiple R-squared:  0.3263, Adjusted R-squared:  0.3065
## F-statistic: 16.47 on 1 and 34 DF,  p-value: 0.000274
```

```
# Or
sqrt(vcov(im)[2,2])
```

```
## [1] 0.1001912
```

Practice

1. Download the “run10” dataset from the openintro package, which has info for participants in the 2012 Cherry Blosson run in DC. Run and inspect a summary of a linear regression of race time (DV) on age (IV).
2. Calculate 95% confidence intervals around each of the regression coefficients.
3. Write a one sentence interpretation of the intercept and slope of this regression.
4. Time is recorded in minutes in this dataset. Re-code the time variable so that it is in seconds and re-run the regression. Then re-interpret the coefficients.
5. Center the age variable and save this new variable as age0. Re-run the same regression and interpret the coefficients.
6. What is the expected average finishing time (in minutes) of a runner who is 30 years old? Provide a 95% confidence and prediction interval around your answer and provide an interpretation for each of those intervals (i.e., explain why they are different).
7. Show how you can use matrix multiplication to get the same result as you got from the linear model in question 1.
8. For the regression in question 1, plot the regression line, with additional lines around it showing a 95% confidence interval around the line.