

Starting R: Session II

Danny Kaplan and Bob delMas and Andy Zieffler

February 26, 2009

Welcome Back!

Agenda for Session II

- Review of fundamental commands
- Brief introduction to modeling/regression notation
- Group work on Basic Skills Tests exercises
- Dinner!
- Hypothesis tests
- If time available: Start on how to organize/document your command.

Requested Topics for Session III

- Panel Data
- Bootstrapping
- Logistic Regression
- Structural Equation Modeling
- Regression Discontinuity
- Custom Graphics
- Hierarchical Linear Models
- Item Response Theory
- Qualitative Data Analysis
- Time Series Analysis

For Fun: Making Maps in R

R is structured to be extensible and to communicate with other systems. This let's interested people develop tools for other people to use. Here's an example for making maps, drawing on a Google resource.

http://www.iq.harvard.edu/blog/sss/archives/2008/04/google_charts_f_1.shtml

Basic Skills Test

- Basic Skills Tests (BST) were reading, mathematics, and writing tests that students who entered grade 8 in 2004-05 or earlier had to pass to receive a diploma from a public high school.
- Students first took the reading and mathematics tests in grade 8 and the writing test in grade 10. Each row represents a school district. All statistics are based on all students who took the BST tests on Feb. 3, 2005. **Data file:** 2005BSTpublic.csv

Starting with a New Data Set

- To keep things organized ... create a directory to store your work. Two steps.
 - Use your computer's file browser to create the directory.
 - Use the **change working directory** tool to tell R which directory to use.

- Copy the table of data into your R session

```
> bst = read.csv( file.choose() )
```

- Get basic information about the data frame:

```
> names(bst)
```

```
[1] "district"      "county"        "population"
[4] "mtested"       "rtested"       "wtested"
[7] "population_qrtl" "mtested_qrtl"  "rtested_qrtl"
[10] "wtested_qrtl"  "pct600m"       "mathavg"
[13] "readavg"       "writeavg"      "mavnolep"
[16] "ravnolep"      "wavnolep"      "speceduc"
[19] "lep"           "freelunch"     "sei_high"
[22] "pctfemale"     "pctmigrant"    "pctnonwhite"
```

```
> nrow(bst)
```

```
[1] 290
```

- Take a look at a few cases to make sure you know what a case is:

```
> head(bst, 4)
```

```

  district    county population mtested rtested wtested
1         1   Carver   22467     583     585     584
2         2 Crow Wing   2836      98     102     104
3         3   Winona   3536      93      94      87
4         4    Rice   18961     314     315     320
  population_qrtl mtested_qrtl rtested_qrtl wtested_qrtl
1              4TH              4TH              4TH              4TH
2              3RD              3RD              3RD              3RD
3              3RD              3RD              3RD              2ND
4              4TH              4TH              4TH              4TH
  pct600m mathavg readavg writeavg mavnolep ravnolep
1    85.8   646.5   659.5     3.3   649.0   661.7
2    72.4   623.2   643.3     3.0   624.1   643.7
3    74.2   633.8   644.0     3.3   633.8   644.0
4    79.3   645.6   647.7     3.3   649.8   651.6
  wavnolep speceduc lep freelunch sei_high pctfemale
1     3.4     11.1 3.2     12.6     YES     47.2
2     3.0     17.1 0.0     39.5     NO     60.0
3     3.3     11.0 0.8     23.7     YES     52.1
4     3.3     10.5 5.2     12.4     YES     48.0
  pctmigrant pctnonwhite
1         0.0         10.4
2         0.0          6.9
3         0.0          8.4
4         0.2          9.0

```

Variables in the BST dataset

Demographics

district Unique ID number for each school district

county Primary Minnesota County for the school district

population 2005 Population of largest cities in the school district

pctfemale Percent of students who took the test that are female

pctmigrant Percent of students who took the test who are from families of migrant workers

speceduc Percent of students who took the test that are in special education

lep Percent of students who took the test identified as Limited English Proficient (LEP)

freelunch Percent of students who took the test receiving free or reduced lunch

freelunch_low YES = $\text{freelunch} < \text{median}$, otherwise NO. [This variable appears to be missing!]

pctnonwhite Percent of students who took the test that are NOT White

mtested Number of students in the district who took the MST Mathematics test

rtested Number of students in the district who took the MST Reading test

Test Results

mathavg Average score on the Mathematics Test for the school district

readavg Average score on the Reading Test for the school district

writeavg Average score on the Writing Test for the school district

mavnolep Average Math score of students NOT designated as Limited English Proficient (LEP)

ravnolep Average Reading score of students NOT designated as Limited English Proficient (LEP)

wavnolep Average Writing score of students NOT designated as Limited English Proficient (LEP)

pct600m Percent of students in the district who scored 600 or greater on the Mathematics Test

Derived Size Results

wtested Number of students in the district who took the MST Writing test

population_qrtl Population is in the 1ST, 2ND, 3RD, or 4TH quartile

mtested_qrtl Number taking math test is in the 1ST, 2ND, 3RD, or 4TH quartile

rtested_qrtl Number taking reading test is in the 1ST, 2ND, 3RD, or 4TH quartile

wtested_qrtl Number taking writing test is in the 1ST, 2ND, 3RD, or 4TH quartile

Commentary on the BST data

There are several ways in which this data set seems to be influenced by limited abilities of computation. (Of course, they are also shaped by the bureaucratic imperatives of the educational system).

- The choice of case. The case is a school district. But it's likely we are interested in studying school **children**, not school districts.

There may be legitimate privacy concerns, but these can be dealt with in a sensible way that doesn't obscure the lessons to be learned from the data.

I often hear people say, "That would be too much data." But that's really a matter of whether it's easy or hard to handle data.

Question: How many students are involved in the BST data?

```
> sum(bst$mtested)
```

```
[1] 57821
```

```
> sum(bst$rtested)
```

```
[1] 57847
```

```
> sum(bst$wttested)
```

```
[1] 59281
```

Not so many!

- The lack of longitudinal data. Are we interested in how children change over time?
- The inclusion of 4 variables that reduce others to quartiles.
 - Why turn a continuous variable (quantile) into a categorical one?
 - Why not rely on people to **compute** the quantities of interest?

Computing the missing Low Free Lunch variable

One of the variables is missing in my data set: `freelunch_low`: a dummy variable to indicate which districts are below the median.

- Access a variable from a data frame with the `$` notation.
- Compute the median with `median` or, more generally, `quantile`.
- Use the boolean comparison operator `<` to check which cases are below the median.
- Store categorical variables as the "factor" type: `factor`

- Create a new variable in a data frame by assignment using the \$ notation.

Putting this together gives:

```
> bst$lowFreeLunch = factor(bst$freelunch < median(bst$freelunch),
  labels = c("LowFL", "HighFL"))
```

Or, for a tighter definition of "low," perhaps the lowest 15 percent:

```
> bst$veryLowFreeLunch = factor(bst$freelunch <
  quantile(bst$freelunch, 0.15), labels = c("veryLowFL",
  "other"))
```

Basic Descriptions of Variables

- **Categorical Variables** Examine the levels and the counts in each level.

```
> table(bst$mtested_qrtl)
```

```
1ST 2ND 3RD 4TH
 73  75  69  73
```

Strange! Why aren't they all the same?

- **Quantitative Variables** Examine the distribution

```
> quantile(bst$population)
```

```
      0%      25%      50%      75%     100%
 8.00   517.00  1830.50  5463.75 387711.00
```

```
> mean(bst$population)
```

```
[1] 8973.045
```

```
> sd(bst$population)
```

```
[1] 30992.12
```

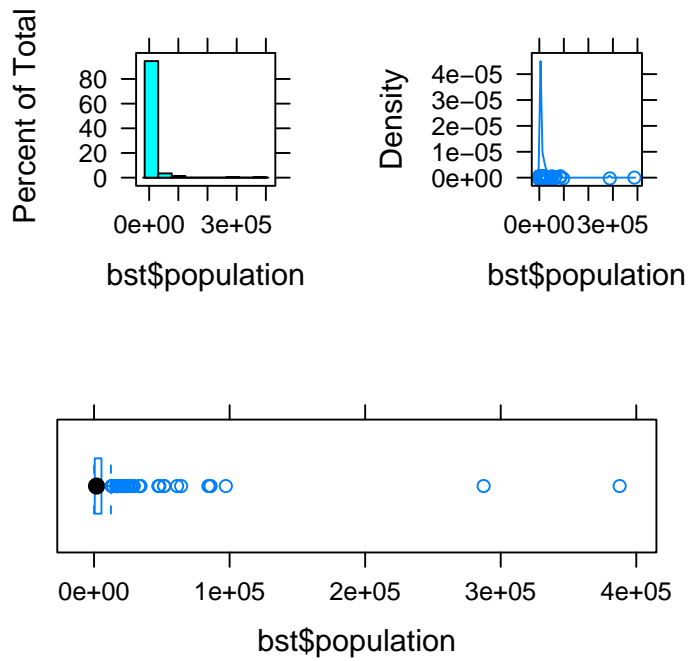
Displaying Distributions

Histograms, densities, boxplots.

```
> plot1 = histogram(bst$population)
```

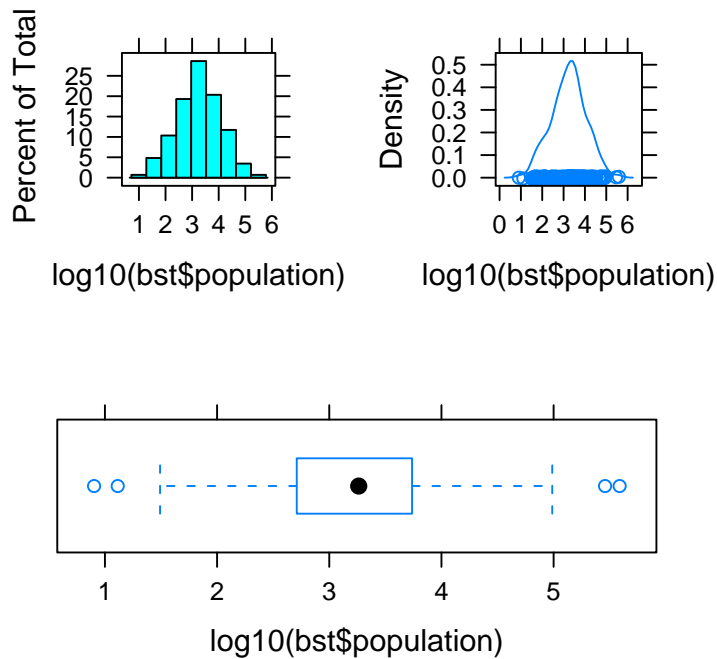
```
> plot2 = densityplot(bst$population)
```

```
> plot3 = bwplot(bst$population)
```



Maybe better to show the logarithm of population

```
> plot1 = histogram(log10(bst$population))
> plot2 = densityplot(log10(bst$population))
> plot3 = bwplot(log10(bst$population))
```

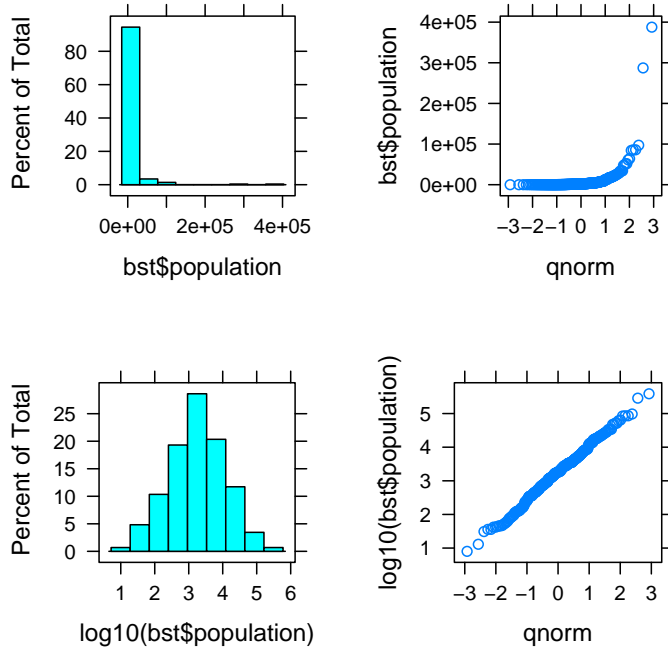


A Graphic for Checking for Normality

There are many specialized forms of graphic, for instance the QQ-plot to display how well a distribution matches a normal distribution.

A normal distribution shows up as a straight line.

```
> plot1 = histogram(bst$population)
> plot2 = qqmath(bst$population)
> plot3 = histogram(log10(bst$population))
> plot4 = qqmath(log10(bst$population))
```



Checking Your Computations

An important aspect of scientific computing is being able to confirm that your computations do what you intended.

Example: Is our new `lowFreeLunch` variable correct. Some ways to check:

- About half the cases should be low:

```
> table(bst$lowFreeLunch)
```

```
LowFL HighFL
 145    145
```

```
> table(bst$veryLowFreeLunch)
```

```
veryLowFL    other
    246         44
```

- The maximum of the low cases should be the median, and the minimum of the high cases should be the median:

```
> median(bst$freelunch)
```

```
[1] 27.75
```

```

> max(subset(bst$freelunch, bst$lowFreeLunch ==
            "LowFL"))

[1] 71.6

> min(subset(bst$freelunch, !bst$lowFreeLunch ==
            "LowFL"))

[1] 3

```

Remember, the point is not to find the mistakes that the computer makes, it's to find the mistakes that YOU make!

Relationships between Two Variables

Is there a relationship between the population of a district and the fraction of students on free and reduced lunch?

```

> cor(bst$population, bst$freelunch)

[1] 0.1136299

There appears to be a small positive correlation. BUT ...
Is this due to the non-normal distribution?

> cor(log10(bst$population), bst$freelunch)

[1] -0.2765097

> cor(bst$population, bst$freelunch, method = "spearman")

[1] -0.3492283

> cor(rank(bst$population), bst$freelunch)

[1] -0.3313075

> cor(rank(bst$population), rank(bst$freelunch))

[1] -0.3492283

```

Graphics for Relationships between Two Variables

- Quantitative vs Quantitative

```

> plot1 = xyplot(log10(population) ~ freelunch,
                data = bst)

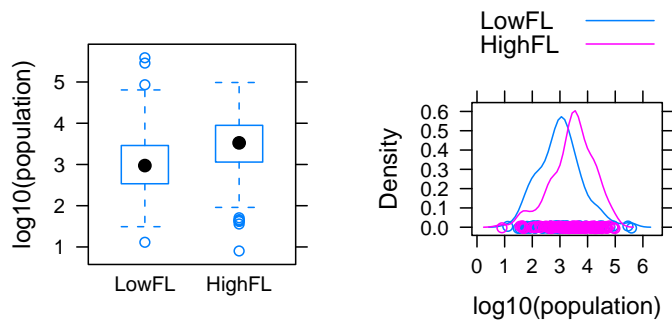
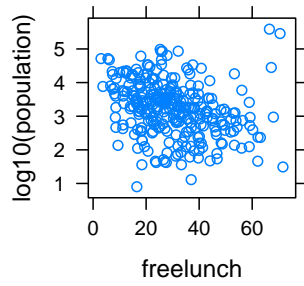
```

- Quantitative vs Categorical

```

> plot2 = bwplot(log10(population) ~ lowFreeLunch,
  data = bst)
> plot3 = densityplot(~log10(population), groups = lowFreeLunch,
  data = bst, auto.key = TRUE)

```

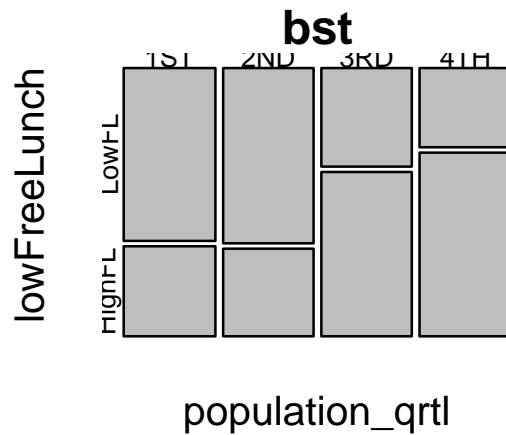


- Categorical vs Categorical

```

> plot4 = mosaicplot(population_qrtl ~ lowFreeLunch,
  data = bst)

```



Regression Operators

- Operators for fitting linear and generalized linear models.
- Use the modeling notation to set the model terms.
- Automatically create “dummy variables” out of categorical variables.

We’ll focus on fitting, the coefficients, the fitted and residual values, and the standard reports: regression and ANOVA.

The Modeling Notation

A special notation for identifying variables and terms.

- Basic structure:
Response Variable \sim Explanatory Variables or Terms
- + separates the explanatory terms.
- : signifies a pure interaction.
- * signifies main effects and interactions.

This notation is used in various model-fitting operators as well as the graphics operators.

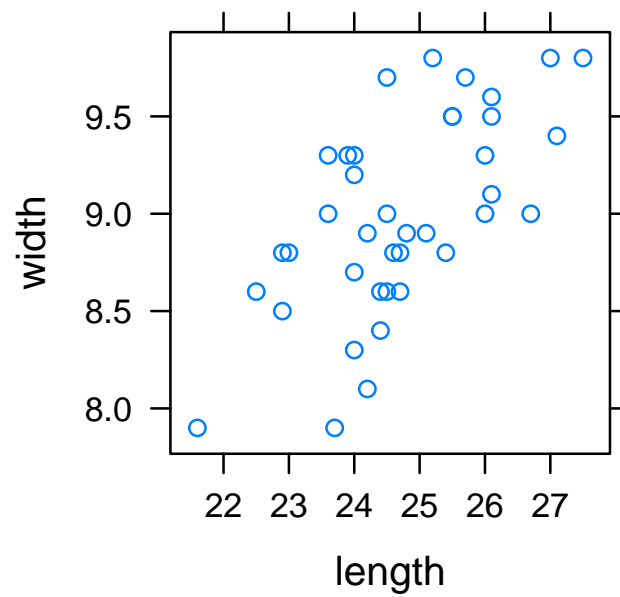
Some operators extend the notation, or interpret it in specialized ways. Example: Structural equation modeling operators.

Examples with Graphics

General rule: $Y \sim X$

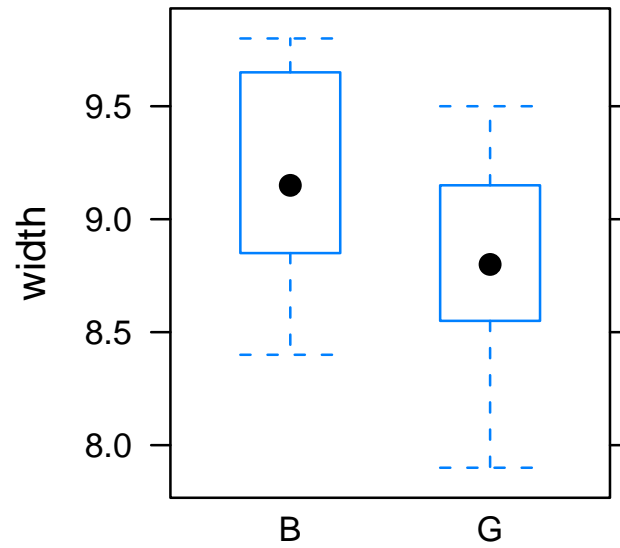
- Scatter plot

```
> print(xyplot(width ~ length, data = kids))
```



- Conditional box-and-whiskers plot

```
> print(bwplot(width ~ sex, data = kids))
```



Examples

Notation: Response variable ~ Response variables

```
> mod1 = lm(width ~ sex, data = kids)
> coef(mod1)
```

```
(Intercept)      sexG
  9.1900000  -0.4057895
```

```
> mod2 = lm(width ~ length, data = kids)
> coef(mod2)
```

```
(Intercept)      length
  2.8622761    0.2479478
```

```
> mod3 = lm(width ~ length + sex, data = kids)
> coef(mod3)
```

```
(Intercept)      length      sexG
  3.6411683    0.2210250  -0.2325175
```

You can also have interaction terms

```
> mod4 = lm(width ~ length + sex + length:sex, data = kids)
> coef(mod4)
```

```
(Intercept)      length      sexG length:sexG
 3.85208056  0.21262376 -0.62387903  0.01582067
```

Models are Objects

As with other things in R, the result of fitting a model is an object. This object can be used as the input to other computations:

- Extracting the fitted values and residuals: `fitted` and `resid`.
- Using the model to make predictions on new input values.
- Standard reports, such as regression reports or ANOVA.
- Comparing two or more models, as in ANOVA.
- Collecting outputs for bootstrapping. (Session III)

Simple Examples of Computing on Models

Teaching About Models The response variable, the fitted values, and the residuals follow the pythagorean relationship: $A^2 + B^2 = C^2$. Example: `mod4` on the kids feed data.

```
> sum(resid(mod4)^2)
```

```
[1] 5.329327
```

```
> sum(fitted(mod4)^2)
```

```
[1] 3158.141
```

```
> sum(kids$width^2)
```

```
[1] 3163.47
```

The Coefficient of Determination This is the fraction of the variance in the response "accounted for" by the model:

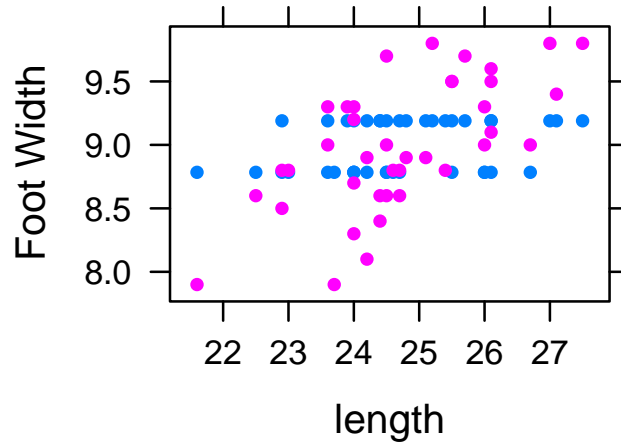
```
> var(fitted(mod4))/var(kids$width)
```

```
[1] 0.4599217
```

This is also a part of standard regression reports.

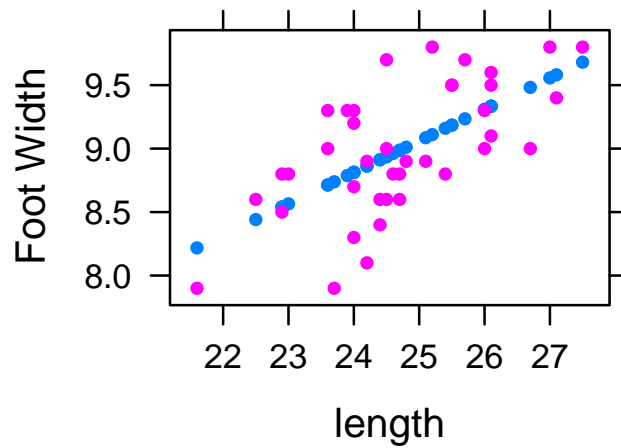
Plotting Fitted Values: mod1

```
> print(xyplot(fitted(mod1) + width ~ length, data = kids,  
  pch = 20, ylab = "Foot Width"))
```



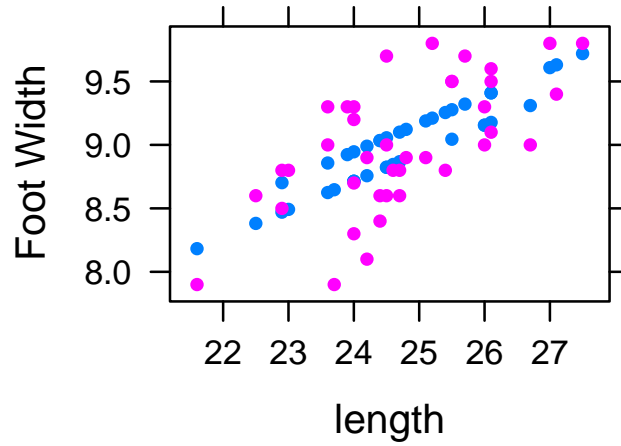
Plotting Fitted Values: mod2

```
> print(xyplot(fitted(mod2) + width ~ length, data = kids,  
  pch = 20, ylab = "Foot Width"))
```



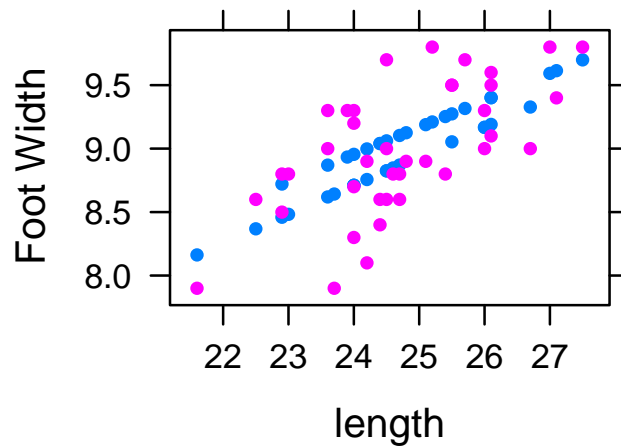
Plotting Fitted Values: mod3

```
> print(xyplot(fitted(mod3) + width ~ length, data = kids,  
  pch = 20, ylab = "Foot Width"))
```



Plotting Fitted Values: mod4

```
> print(xyplot(fitted(mod4) + width ~ length, data = kids,  
  pch = 20, ylab = "Foot Width"))
```



Slightly different slopes for the fitted lines.

Regression Reports

```
> summary(mod1)
```

Call:

```
lm(formula = width ~ sex, data = kids)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-0.88421	-0.29000	0.01579	0.46000	0.71579

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	9.1900	0.1057	86.97	<2e-16 ***
sexG	-0.4058	0.1514	-2.68	0.0109 *

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4726 on 37 degrees of freedom
Multiple R-squared: 0.1626, Adjusted R-squared: 0.14
F-statistic: 7.184 on 1 and 37 DF, p-value: 0.01092

```
> summary(mod4)
```

Call:

```
lm(formula = width ~ length + sex + length:sex, data = kids)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-0.74233	-0.27577	0.04042	0.23837	0.68051

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	3.85208	1.84912	2.083	0.04460 *
length	0.21262	0.07357	2.890	0.00657 **
sexG	-0.62388	2.50100	-0.249	0.80447
length:sexG	0.01582	0.10096	0.157	0.87638

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3902 on 35 degrees of freedom
Multiple R-squared: 0.4599, Adjusted R-squared: 0.4136
F-statistic: 9.935 on 3 and 35 DF, p-value: 7.009e-05

ANOVA — for after dinner

- Many people think of ANOVA as a test for the difference of multiple group means. Actually, it's a way of describing how much each additional term contributes to a model.
- So, it's really a description of a model, not a description of data.

```
> anova(mod1)
```

Analysis of Variance Table

Response: width

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
sex	1	1.6044	1.6044	7.1841	0.01092 *
Residuals	37	8.2633	0.2233		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> anova(mod4)
```

Analysis of Variance Table

Response: width

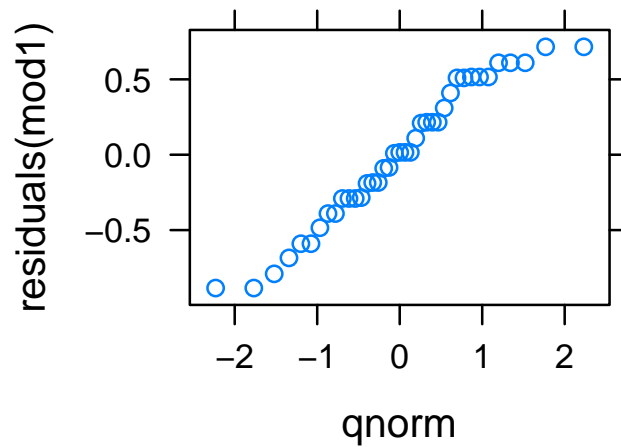
	Df	Sum Sq	Mean Sq	F value	Pr(>F)
length	1	4.0557	4.0557	26.6353	9.86e-06 ***
sex	1	0.4790	0.4790	3.1456	0.08484 .
length:sex	1	0.0037	0.0037	0.0246	0.87638
Residuals	35	5.3293	0.1523		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Regression Diagnostics (example)

Are the residuals normally distributed?

```
> print(qqmath(residuals(mod1)))
```



Generalized Linear Models

Very much analogous to linear models.

Example: Logistic regression.

```
> mod5 = glm(sex == "G" ~ width + length + domhand,
             data = kids, family = "binomial")
```

The fitted model values reported from logistic regression are probabilities.
The “link” values are also available.

Classical Hypothesis Tests

Andy Zieffler will talk about these after dinner.

- t-tests (for differences in means)
- χ^2 -tests (for counts)
- p-tests (for proportions)
- ANOVA for regression
- Tukey’s HSD
- Homogeneity of variance

t-tests ... briefly

In analyzing the Basic Skills Test data, you may want to do a t-test.

The basic operator is called `t.test`. It can be used in several modes

- **One-sample t-test** to compare a mean to a null hypothesis value.

Example: Is the student body typically 50% female?

```
> t.test(bst$pctfemale, mu = 50)
```

One Sample t-test

```
data: bst$pctfemale
t = -5.9053, df = 289, p-value = 9.868e-09
alternative hypothesis: true mean is not equal to 50
95 percent confidence interval:
 47.80696 48.90339
sample estimates:
mean of x
 48.35517
```

- **Paired t-test** to compare the means of two different groups or variables when there is a natural pairing in the members of the group, e.g., before and after.

Example: Is there a difference between the overall average reading scores and the average reading scores for non-LEP students? Pair by school district.

```
> t.test(bst$writeavg, bst$wavnolep, paired = TRUE)
```

Paired t-test

```
data: bst$writeavg and bst$wavnolep
t = 0.0576, df = 289, p-value = 0.954
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.02286143  0.02424074
sample estimates:
mean of the differences
 0.0006896552
```

- **Two sample t-test** to compare the means of two different groups. This comes in two different flavors: equal and unequal variance.

Two different ways to do this:

– Comparing two variables:

```
> t.test(bst$writeavg, bst$wavnolep)
```

Welch Two Sample t-test

```
data: bst$writeavg and bst$wavnolep
t = 0.043, df = 470.655, p-value = 0.9657
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.03079461  0.03217392
sample estimates:
mean of x mean of y
 3.188276  3.187586
```

- Comparing one variable broken down into groups by another categorical variable.

```
> t.test(writeavg ~ lowFreeLunch, data = bst)
```

Welch Two Sample t-test

```
data: writeavg by lowFreeLunch
t = -7.1099, df = 287.801, p-value = 9.203e-12
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.13736937 -0.07780304
sample estimates:
mean in group LowFL mean in group HighFL
 3.134483                3.242069
```

Standing On My Soapbox: The t-test

t-tests are grossly over-emphasized in introductory statistics courses.

- They ask too simple a question. They don't leave room for covariates.
- They are mysterious: Why \sqrt{n} ? Where does the table come from?
- They force instructors to spend time explaining t^* rather than important concepts like covariates.
- They fit into a general framework, linear modeling, better to teach the framework rather than the special cases.
- The distinction between equal and unequal variance tests is a mathematical nicety of no practical significance. If the variances are so different that the unequal variance t-test is better, you should have been doing non-parametrics anyways.

Group Work for Exercises

Form into groups of two to four and work on the exercises based on the Basic Skills Test data.

- We haven't yet introduced the commands for parts 10 & 11 and 16 & 17.

- We'll review some of the classical hypothesis testing operators after dinner.
- As a simple way to document your work, you might want to open a word process and cut and paste your SUCCESSFUL commands and the related output.