Lecture 1. Regression and Anova refresher.

Some preliminaries

These notes will cover the same material I talk about during class. Supplementary readings will be posted on Laulima. These will be book chapters on similar topics, as well as more info on how to use these methods in R. The lectures notes + readings will cover a lot of ground, but if you want to get book(s) on this material, these are my favorites (and most of the supplementary reading will be pulled from these):

Gelman and Hill. *Data analysis using regression and multilevel/hierarchical models*.
Bolker. *Ecological models and data in R*.
Fox. *An R companion to applied regression*.

R and other resources can be found here http://cran.us.r-project.org/. R Studio (http://www.rstudio.com/) is free software that makes the R GUI at little more user friendly, at least if you're using a PC (the basic Mac GUI is better than the basic PC GUI).

Deterministic and stochastic models

What do we mean by thinking about statistics as model building? Let's start with an example of a *deterministic* model of how something works in nature. Ecology does not have a lot of simple theory derived from first principles (because nature is complicated), but an intriguing example comes from work on how metabolic rate (i.e. energy consumption) on an individual organism scales as a function of its body size. In general larger organisms consume more energy (not surprising), but they actually consume less on a *mass-specific* basis. For example, an elephant weighs ~10^5 times more than a mouse, but it's metabolic rate is only about ~10^4 times more, or about ten times less on a mass-specific basis. If metabolic rate had a 1:1 relationship with body mass, then we would expect a relationship like this:

$$W = a*M$$

where W is basal metabolic rate, M is the mass of the individual, and a is a constant that converts between the two. In reality the relationship looks like a power law:

$$W = a*M^{c} \qquad (1)$$

where c is an exponent that defines the extent to which the relationship deviates from a 1:1 line. An influential paper by West et al. (1997, *Science*, "A general model for the origin of allometric scaling laws in biology") used the physics of transporting materials within organisms' circulatory systems to predict that $c = ¾$. This inspired a lot of analyses testing whether this was actually true; Figure 1 shows an example for mammals. Statistically, it is easier to work with this relationship if we log transform both sides:

$$\log(W) = \log(a) + c*\log(M) \qquad (2)$$

If you don't understand how I went from (1) to (2), please go to Wikipedia or elsewhere and review the rules for logarithms, because they will come up frequently in this course.

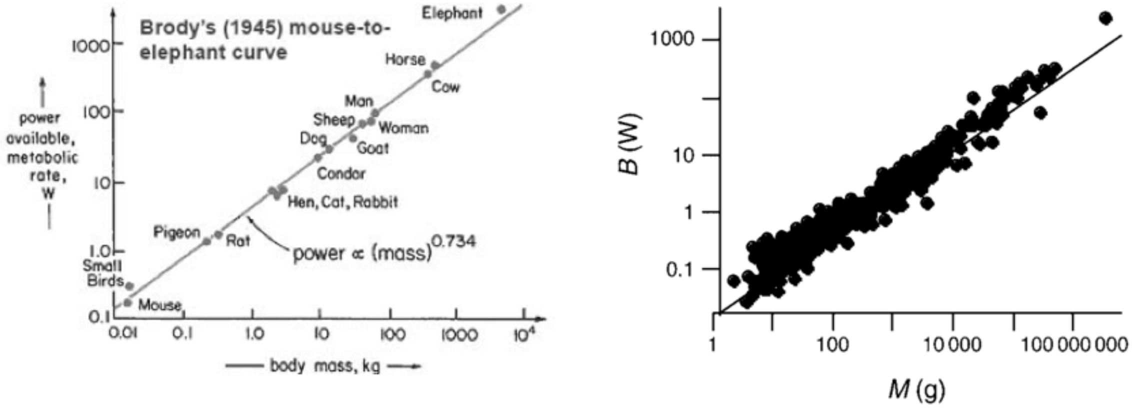


Figure 1. Basal metabolic rate (watts = joules per second) vs. individual body mass. On the left, a classic plot by Brody going from mice to elephants. One the right, a modern compilation of a wide variety of mammals. The estimate slope from a regression on the log-transformed variables is 0.724 (95% CI: [0.706, 0.742]). From Duncan et al. 2007, *Ecology*, "Testing the metabolic theory of ecology: allometric scaling exponents in mammals".

Equation (1) is an example of a *deterministic model*, because there is an exact, noiseless relationship between two variables (metabolic rate and mass). Most analyses you will do in biology do not have theoretical *a priori* predictions like this allometric scaling relationship. Nonetheless, thinking about the deterministic model that underlies your statistical analysis is going to be important, because analyzing biological complexity often involves a complex deterministic model.

The data in Figure 1 show a fairly tight relationship, but the measurements don't all fall on a line. This is presumably due to the fact that there are other factors besides body mass that affect metabolic rate, as well as the fact that the variables are measured with some error. Therefore, if we want to use this data to ask "What is the empirical value of *c*? Is it consistent with the theoretical prediction?", we need to estimate the parameters of a deterministic model that has the same shape as eqn (2), supplemented with a *stochastic* part that represents measurement error:

$$\mu_i = \beta_0 + \beta_1 * \log(M_i) \qquad (2)$$

$$\log(W_i) \sim \mathrm{Normal}(\mu_i, \sigma). \qquad (3)$$

Equation (2) is the equation for a line predicting μ_i, which is the expected value for log metabolic rate for the organism with mass M_i. β_0 is the intercept for the line (this will be an estimate of log(*a*), but we don't have a theoretical prediction for *a*), and β_1 is the slope of the line (the theoretical prediction is ¾). Because there are other sources of variation for metabolic rate, to fully model the observed data (W_i) we need to add some noise. We'll assume that noise is normally distributed with a standard deviation of σ. So formula (3) is saying that the observed metabolic rate $\log(W_i)$ is drawn from a normal distribution with a mean equal to the expected value μ_i, and with a standard deviation of σ. The tilde, '~' is often used to denote a random draw from a probability distribution.

Equations (2-3) are an example of *stochastic model*. All that means is that the relationship we are modeling includes a random component. In principle if we were omniscient beings who knew everything about how all organisms worked, and everything about how they were measured, then we would be able to perfectly predict their metabolic rate with no extra noise. Because we don't know everything, we treat the parts we don't understand as a random process, and part of the challenge of modeling data is modeling this random process in a reasonable way. What makes (2-3) a *statistical model* is that the parameters of the stochastic model (β_0, β_1, σ) are going to be estimated from the data. In this case, we can estimate these parameters with a simple linear regression. In general, statistical analyses involve some sort of model that describes how a response variable is related to one or more predictor variables, and the statistical model can be written in terms of a deterministic part and a stochastic part:

Deterministic part

$$\mu_i = \text{Function}(X_{1i}, X_{2i}, \ldots) \tag{4}$$

Stochastic part

$$Y_i \sim \text{Distribution}(\mu_i) \tag{5}$$

where Y_i is the *i*th value of the response variable, μ_i is the mean or expected value of Y_i, and X_{ji} is value *i* for predictor *j*. These two parts are saying that "the expected value for the response variable is a function of some predictors, and the observed value for the response variable is a random draw from some probability distribution with a mean equal to μ_i". Eqsn (4-5) are just a generalization of (2-3). To emphasize the point I'm getting at here, *when you fit a statistical model you are assuming that the data are generated by some stochastic process, and the goal of your analysis is to estimate the parameters that define that process*. The value of this framework is that it encompasses a wide variety of statistical analyses, but it is a fairly abstract way of thinking about things. Let's get more concrete by looking in depth at simple linear regression from this perspective.

Linear regression as a stochastic model

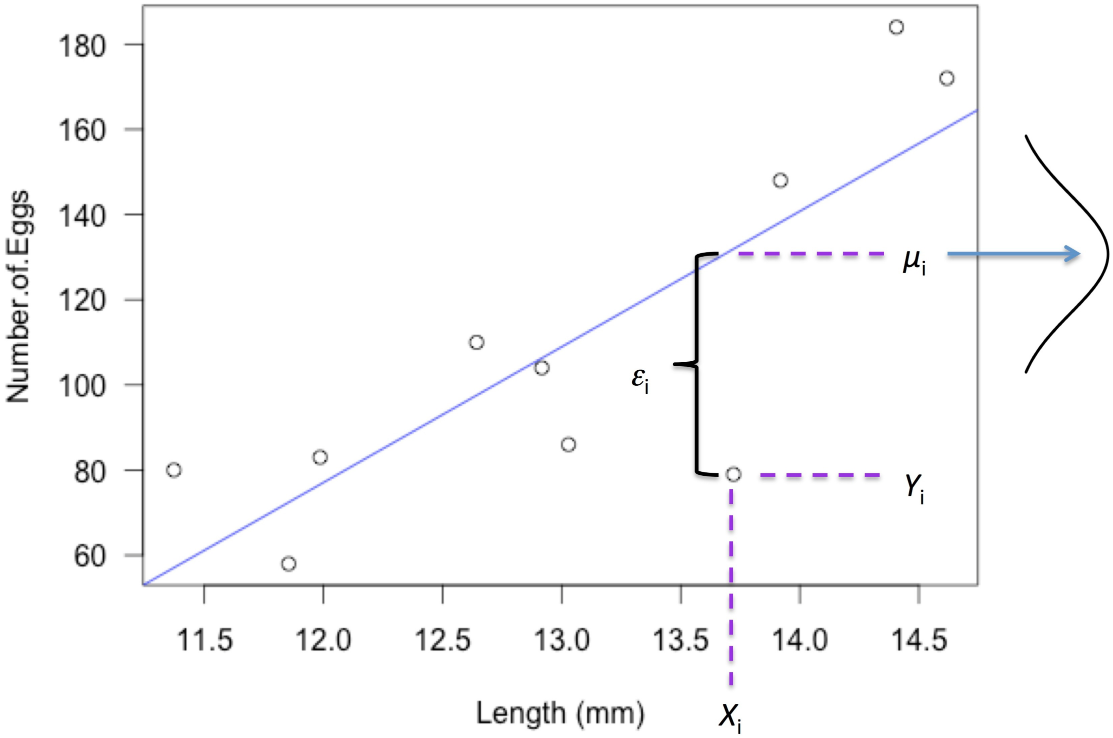


Figure 2. Number of eggs vs. length for females of the isopod *Idotea balthica*, from Manyak-Davis et al. 2013. For the point (Y_i, X_i), the expected value is μ_i, which means that the residual ε_i is assumed to be drawn from a normal distribution with mean μ_i and standard deviation σ.

Figure 2 shows an example of a standard linear regression. The authors measured fecundity (number of eggs) as well as body length for individuals of a common isopod. Part of their analysis was to compare the relationship between fecundity and size for different populations, but here we'll just focus on the relationship for these ten individuals. I've plotted these two variables for ten individuals, and the fitted regression is shown in blue. To stick with the notation defined above, Y_i and X_i are respectively # eggs and length, for individual *i*. The equation for a linear regression is usually written like

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \,, \qquad (6)$$

where β_0 is the intercept, β_1 is the slope, and ε_i is the residual error for individual *i*. This line is fit using least squares estimation, which means that the fitted intercept and slope coefficients are those that minimize the "sum of the squares": the sum of the squared vertical distances between the observations Y_i and the the line (i.e., minimizing $\sum_i \varepsilon_i^2$). The residuals ε_i are assumed to be normally distributed.

To make eqn (6) look more like (4) and (5), we can write it like this:

$$\mu_i = \beta_0 + \beta_1 X_i \tag{7}$$

$$Y_i \sim \text{Normal}(\mu_i, \sigma) \tag{8}$$

Which is the same format I used for (4) and (5). Equation (7) is the deterministic part, and formula (8) is the stochastic part. Here μ_i is the expected value for Y_i. How can we interpret this model biologically? It says that the number of eggs laid by a female is random but constrained, such that the mean number of eggs laid by a female of length X is $\beta_0 + \beta_1 X$, but the actual number of eggs laid is normally distributed around $\mu_i = \beta_0 + \beta_1 X_i$, with standard deviation σ.

Clarifying statistical assumptions. Why am I spending all this time belaboring linear regression and writing the equations in a slightly different way? There are several inter-related reasons. First, writing the regression with eqns (7) and (8) helps clarify what you are assuming when you fit a linear regression. Two of the important assumptions are homoscedasticity and independence of the residuals. Homoscedasticity means that the variance of the residual error is constant across the whole range of the predictor X. Independence means that each residual is drawn separately from the normal distribution. Examples of what it looks like to break these assumptions are shown in Figure 3. The important connection to (7) and (8) is that *you can't generate the patterns in Figure 2 using the model described by (7) and (8)*. Strictly speaking, it is possible, but very unlikely, that the patterns in Figure 3 could be generated by drawing each Y_i independently from a normal distribution with mean μ_i and a standard deviation σ that is constant across all Y_i. An important part of statistical analysis is figuring out whether the data violates the assumptions of the analysis, and *another way of looking at this is to ask whether the model you are using can plausibly generate the data you observe*.

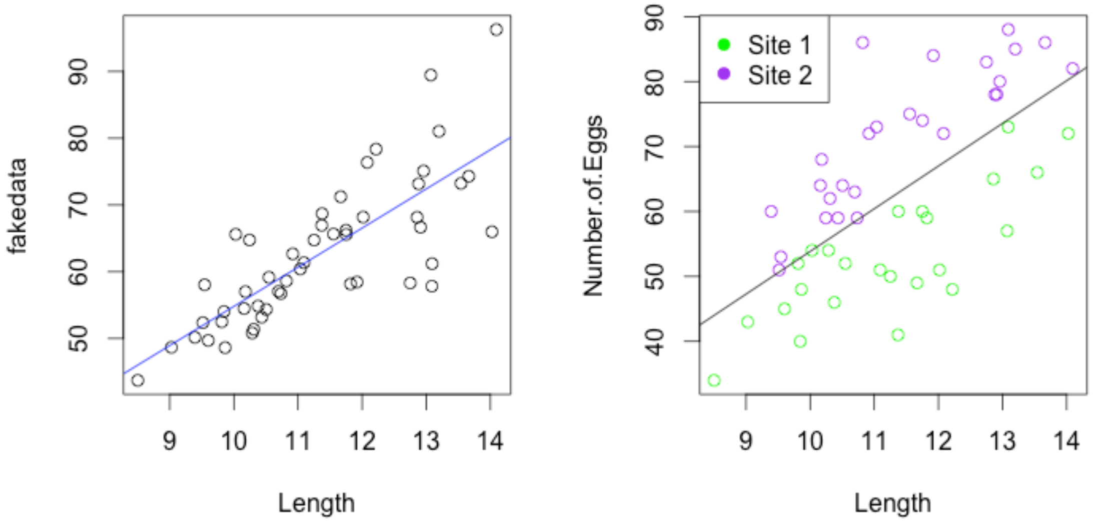


Figure 3. Examples of breaking the assumptions of the linear regression model. In the plot on the left, the residual variance is not constant (it increases as Length increases). In the plot on the right, the residuals are not independent of one another (residuals from the two Sites are clustered, such that Site 2 residuals are larger than Site 1 residuals).

Stochastic models and simulating data. A related point is one that I will emphasize throughout the course: *thinking about a statistical analysis in terms of stochastic models clarifies how you would simulate fake data that are generated by the same process as the original data*. Simulating fake data is very helpful in designing experiments, figuring out what model to use for your analysis, and performing hypothesis tests on the results. We will discuss this thoroughly in future lectures, but for now I'll show a simple example for linear regression.

```
library(gdata)

#the gdata library lets you load a worksheet from an MS Excel file
fecund.all = read.xls("ManyakBellSotka_AmNat_AllData.xlsx", sheet = 3)
#take a subset to just look at one location for now
fecund = subset(fecund.all, Population == "VIMS")
#rename the length variable
names(fecund)[3] = "Length"
#set up to put 10 plots on one image
par(mfrow = c(2,5))
#plot the data
with(fecund, plot(Number.of.Eggs ~ Length, main = 'Original data'))
```

```
model = lm(Number.of.Eggs ~ Length, data = fecund)
#plot the fitted line
abline(model, col = 'blue')

#use the fitted model to simulate fake data 9 times
for (i in 1:9) {
  fakedata.expected = coef(model)[1] + coef(model)[2]*fecund$Length
  fakedata.observed = rnorm(length(fakedata.expected), fakedata.expecte
d, summary(model)$sigma)
  plot(fakedata.observed ~ fecund$Length, main = paste('Simulated data
', i), xlab = 'Length', ylab = 'Simulated Number of Eggs', ylab = c(20,
110))
  abline(lm(fakedata.observed ~ fecund$Length), col = 'blue')
}
```

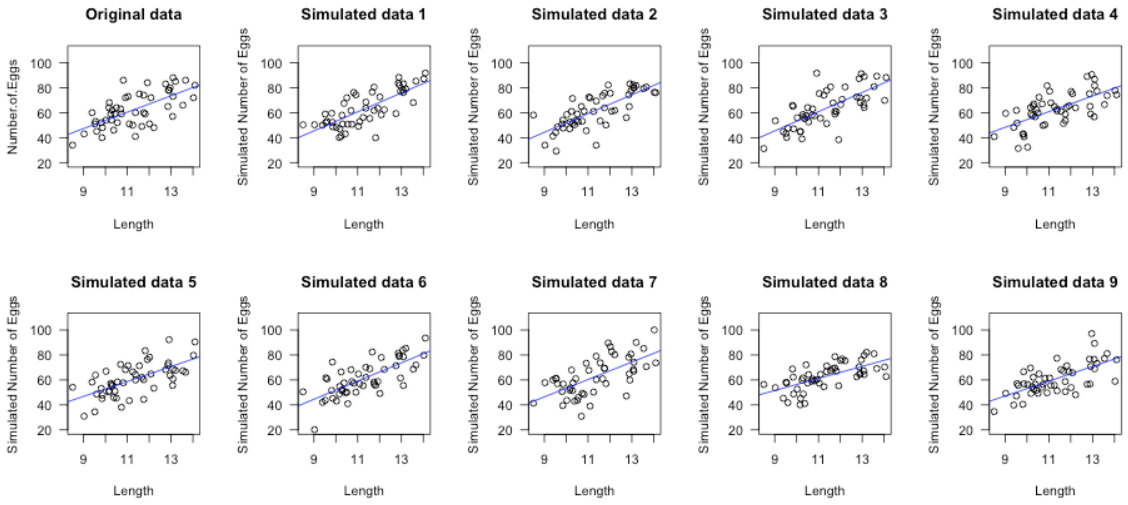


Figure 4. Example of simulating data from a linear regression. The top-left plot is the original data. The nine other plots use the fitted intercept, slope, and residual variance to simulate data from the stochastic model that the regression represents.

The main point to take from Fig. 4 is that the pattern of residuals changes somewhat as you make new random draws from the normal distribution, but none of these plots have an extreme pattern like in Fig. 3 where the residual variance strongly increases with Length. Of course there are ways to make this comparison quantitative, but for now we'll stick with visual inspection.

Linear regression in R

Fitting a regression in R is straightforward. The lm() function (for "linear model") is used, and a formula is specified with the response variable on the left and the predictor(s) on the right (see ?formula in R for a good explanation of R formula specification).

```
model = lm(Number.of.Eggs ~ Length, data = fecund)
```

We can inspect the fitted model with summary():

```
summary(model)

##
## Call:
## lm(formula = Number.of.Eggs ~ Length, data = fecund)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.841  -7.404   0.339   7.778  26.778
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -11.84      12.18   -0.97     0.34
## Length          6.57       1.07    6.15  1.5e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.6 on 48 degrees of freedom
## Multiple R-squared:  0.441,  Adjusted R-squared:  0.429
## F-statistic: 37.9 on 1 and 48 DF,  p-value: 1.46e-07
```

The summary() function outputs a lot of info, perhaps too much. Under Coefficients are the parameters estimates for the intercept, “(Intercept)”, and slope, “Length”. It also returns standard errors, t-statistics, and a p-value for a t-test that asks whether the estimated coefficient is different from zero. Usually we care whether the slope is different from zero (indicating a significant effect of the predictor), but not whether the intercept is different from zero.

From linear regression to nonlinear and/or non-normal situations

The final reason I am introducing linear regression as a stochastic model is because this framework generalizes to many other analyses. For a nonlinear relationship, such as a saturated consumption rate that is common in biology, the deterministic part is altered:

$$\mu_i = \frac{V_{\max} X_i}{K + X_i}$$
$$Y_i \sim \text{Normal}(\mu_i, \sigma)$$

where $V_{\max}$ is maximum consumption rate, and K is the half-saturation constant. A common model we will discuss later is the log-linear model for count data, where the stochastic part is a Poisson distribution and the deterministic part is a linear model wrapped inside an exponential function:

$$\mu_i = exp(\beta_0 + \beta_1 X_i + \ldots)$$
$$Y_i \sim \text{Poisson}(\mu_i)\ .$$

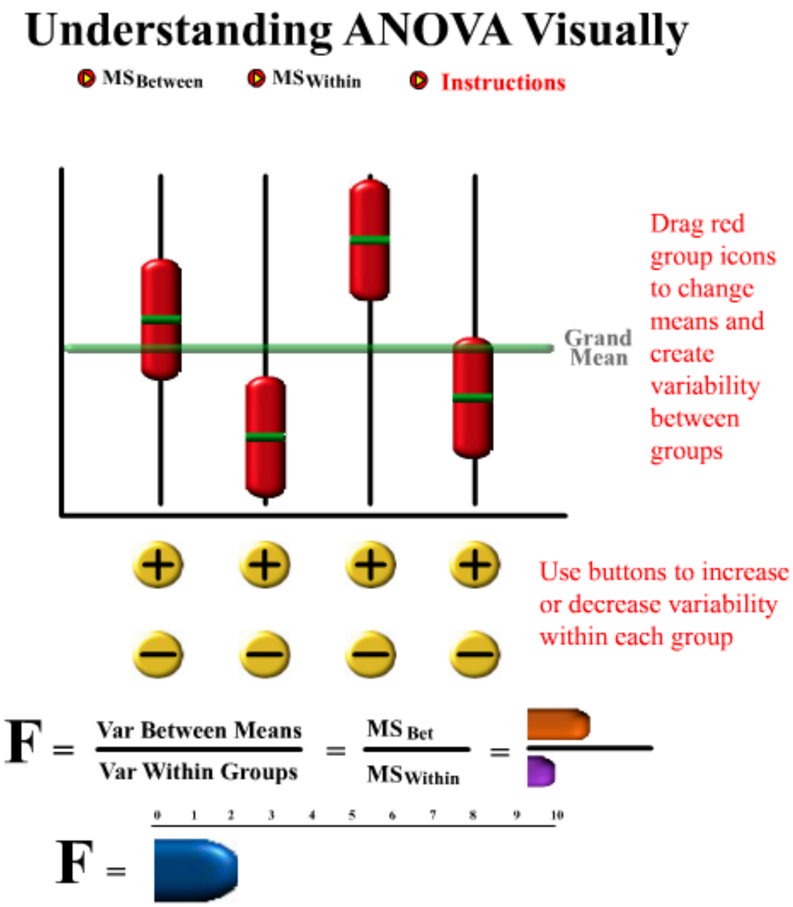
When data have some kind of grouping structure, such as spatial sites or temporal sampling periods, a mixed model (aka hierarchical or random effects model) is appropriate. The novel thing about mixed models is that some of the parameters are themselves given a stochastic model, so that there are multiple levels of variability in the overall model:

$$\mu_{ij} = \beta_{0j} + \beta_1 X_{ij} + \ldots$$
$$\beta_{0j} \sim \text{Normal}(0, \sigma_b)$$
$$Y_{ij} \sim \text{Normal}(\mu_{ij}, \sigma_r)$$

where Y_{ij} is data point *i* in group *j*, and β_{0j} is the group-specific intercept for group *j*, and the group-specific intercepts are normally distributed with standard deviation σ_b. Most of the course will be spend on learning the models I've sketched here, and I am mentioning them now to show how we'll use a common framework in which to think about these various methods.

Linear models and Anova

Analysis of variance (Anova) is the fundamental method for testing whether a set of groups (e.g. different experimental treatments) have different means. It is usually presented in terms of comparing the amount of variance between groups to the amount of variance within groups (Fig. 4).



Drag red group icons to change means and create variability between groups

Use buttons to increase or decrease variability within each group

Figure 4. Visual representation of Anova, taken from http://www.psych.utah.edu/stat/introstats/Anovaflash.html.

The creation of Anova by R. A. Fisher (and his intellectual antecedents) is an example of the kind of brilliant solution developed by statisticians 100 years ago when all the computation had to be done by hand. Partitioning the variability in the data among and between groups is straightforward if laborious, and the F statistic provides an exact (analytical) formula for testing whether there is significant variation among groups. However, in many cases the application of Anova breaks down, either because the experimental design is complex or unbalanced, making it difficult to define the appropriate sums of squares and F statistic, or because the residual variation is not normally distributed and can't be easily transformed to normality. Much of this course will focus on the methods one can use when standard Anova breaks down, but to begin with we need to review Anova and look at it from a stochastic model point of view. This will give us a unified framework in which we can think about both classical and more modern approaches to the analysis of experiments as well as observational data.

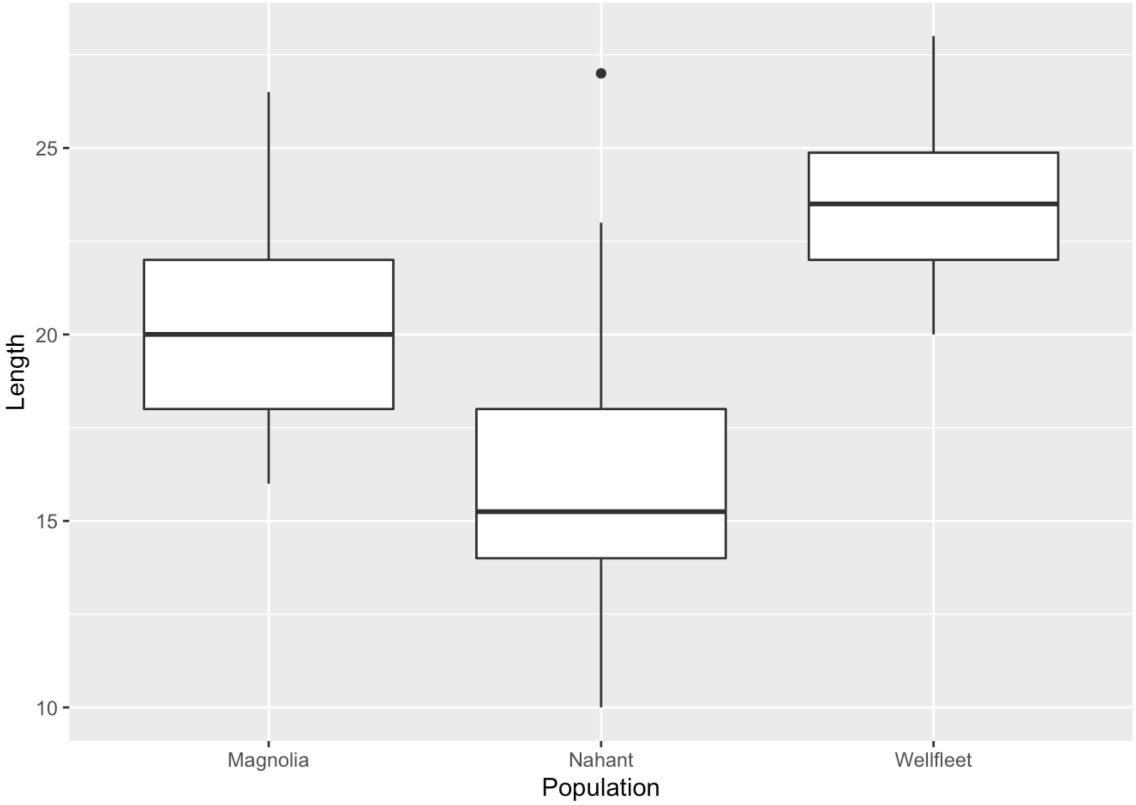


Figure 5. Boxplot of lengths of male *Idotea* from three populations, from Manyak-Davis et al. 2013.

Going back to the isopod study used in Fig. 1, let's compare the length of male isopods collected from three sites in Massachusetts (Figure 5). Based on a boxplot, it's pretty clear the isopods from the different sites have different lengths (on average). To test whether this is the case, we'll create a model along the lines of the linear regression we looked at earlier. How do we take a factor with multiple levels (in this case, Site has 3 levels), and give that a quantitative representation in a stochastic model? The trick is to use "dummy" or "indicator" variables that, when combined, code the factor level for each entry. The default coding scheme for a 3-level factor in R works like this (Fig. 6):

$\mu_i = \beta_0 X_{0i} + \beta_1 X_{1i} + \beta_2 X_{2i}$

Table 1. Using indicator variables to define factors in an Anova.

	X_0	X_1	X_2
Magnolia	1	0	0
Nahant	1	1	0
Wellfleet	1	0	1

What this means is that the expected length for isopod *i* is μ_i, and this expected length depends on which group the isopod is in. If the isopod is from site Magnolia, then the expected length (i.e. the mean length at Magnolia) is equal to

$\beta_0*1 + \beta_1*0 + \beta_2*0 = \beta_0$

In other words, the parameter β_0 is the group mean for Magnolia. If the isopod is from site Nahant, then the expected length is

$\beta_0*1 + \beta_1*1 + \beta_2*0 = \beta_0 + \beta_1$

So this means that the parameter β_1 quantifies the difference between Nahant and Magnolia. Finally, if the isopod is from site Wellfleet, the expected length is

$\beta_0*1 + \beta_1*0 + \beta_2*1 = \beta_0 + \beta_2$

In other words, the parameter β_2 quantifies the difference between Wellfleet and Magnolia. To summarize*, the way the coding scheme works is that it uses one parameter to define the Magnolia group as a baseline, and uses two other parameters to define the two other groups relative to Magnolia.* This may all seem a bit convoluted just for doing a routine Anova, but it is essential for interpreting and plotting the output of any model in R that has a factor as a predictor. The full stochastic model for this data is

$$\mu_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} \tag{9}$$

$$Y_i \sim \text{Normal}(\mu_i, \sigma) \tag{10}$$

(note I removed X_0 for simplicity, because it is equal to 1 for all *i*).

Eqns (9-10) look very similar to (7-8), which define a linear regression. This is intentional and also important, because linear regression and Anova are both special cases of the more general category, *linear models*. We will explore a wide variety of linear models throughout the course, but for now just notice that by using indicator variables to code factors, you can include factors (as in Anova) and continuous predictors (as in linear regression) in the same model framework. Also, as discussed above for linear regression, writing the Anova as a stochastic model with (9) and (10) helps to clarify the standard Anova assumptions. The residuals are assumed to be normally distributed, which is obvious, and are assumed to be independently distributed, as described above. In the context of Anova, homoscedasticity means that the different groups (sites) have the same residual variance. This is implicit in (10) because there is a single distribution from which all the residuals are drawn.

Anova as a linear model in R

How do we perform an Anova as a linear model in R? Building the model with the lm() function is quite easy (end of this code, which also loads the data and plots it):

```
#the gdata library lets you load a worksheet from an MS Excel file
survey.all = read.xls("/Users/orbistertius/Documents/teaching/OCN 683/l
ecture1/ManyakBellSotka_AmNat_AllData.xlsx", sheet = 1)
#rename the length variable
survey.all = rename(survey.all, Length = Length..mm.)
#take a subset to just look at males from MA
survey.north.m = survey.all %>% filter(Region == "North" & Sex == 'M')
 %>% select(Length, Population)

#plot the data - the populations look pretty different based on a boxpl
ot
ggplot(survey.north.m, aes(x = Population, y = Length)) + geom_boxplot
()

#compare the population means using an ANOVA, with the 'lm' function.
model = lm(Length ~ Population, data = survey.north.m)
#take a look at the model
summary(model)

##
## Call:
```

```
## lm(formula = Length ~ Population, data = survey.north.m)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.170 -2.087 -0.340  1.660 10.830
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          20.3400     0.3818  53.278  < 2e-16 ***
## PopulationNahant     -4.1700     0.5399  -7.724 1.61e-12 ***
## PopulationWellfleet   3.0000     0.5399   5.557 1.26e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.7 on 147 degrees of freedom
## Multiple R-squared:  0.5476, Adjusted R-squared:  0.5414
## F-statistic: 88.96 on 2 and 147 DF,  p-value: < 2.2e-16
```

The response variable (Length) is one the left side of the formula, and the factor (Population) is on the right side of the formula. We can use summary() to get basic info on the fitted parameters of the linear model. The listed coefficients are (Intercept), PopulationNahant, and PopulationWellfleet. How do these correspond to the indicator variables described above? The (Intercept) coefficient is the baseline against which the other coefficients are defined. This corresponds to β_0 in eqn (9), i.e. the group mean for Magnolia. PopulationNahant corresponds to β_1, i.e. the difference between Nahant and Magnolia; and PopulationWellFleet corresponds to β_2, i.e. the difference between Wellfleet and Magnolia. The naming of the coefficients is a bit confusing at first, but this is the default way that lm parses the data. The first level of the factor (first alphabetically) is treated as the baseline and called (Intercept), while the other levels combine the name of the factor ("Population") and the level of the factor that is being compare to (Intercept).

If you are confused about how factors are being defined and modeled in your analysis, one way to see how the model works is with the model.matrix() function:

```
model.matrix(model)
```

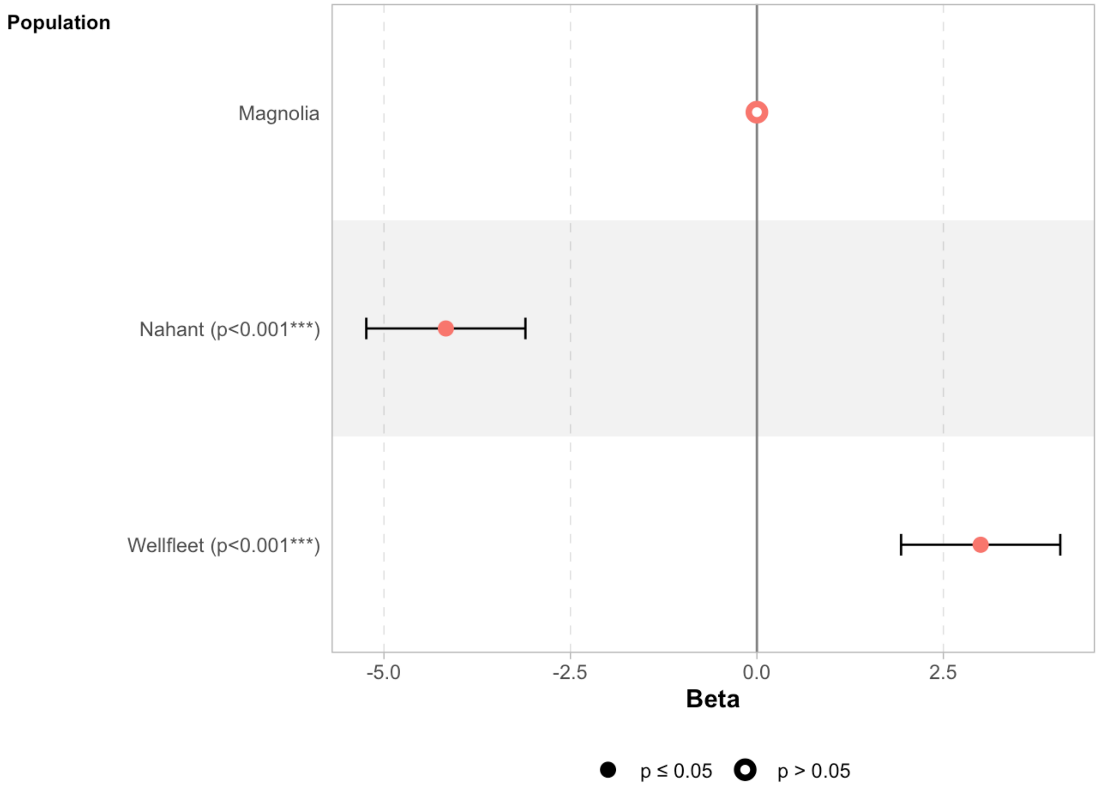
```
##     (Intercept) PopulationNahant PopulationWellfleet
## 151           1                0                   0
## 152           1                0                   0
## 153           1                0                   0
## 154           1                0                   0
## 155           1                0                   0
## 156           1                0                   0
## 157           1                0                   0
## 158           1                0                   0
## 159           1                0                   0
## 160           1                0                   0
## 161           1                0                   0
## 162           1                0                   0
## 163           1                0                   0
## 164           1                0                   0
## 165           1                0                   0
## 166           1                0                   0
## 167           1                0                   0
## 168           1                0                   0
## 169           1                0                   0
## 170           1                0                   0
## 171           1                0                   0
## 172           1                0                   0
## 173           1                0                   0
## 174           1                0                   0
## 175           1                0                   0
## 176           1                0                   0
## 177           1                0                   0
## 178           1                0                   0
## 179           1                0                   0
## 180           1                0                   0
## 181           1                0                   0
## 182           1                0                   0
## 183           1                0                   0
## 184           1                0                   0
## 185           1                0                   0
## 186           1                0                   0
## 187           1                0                   0
## 188           1                0                   0
## 189           1                0                   0
## 190           1                0                   0
## 191           1                0                   0
## 192           1                0                   0
## 193           1                0                   0
## 194           1                0                   0
## 195           1                0                   0
## 196           1                0                   0
## 197           1                0                   0
## 198           1                0                   0
## 199           1                0                   0
## 200           1                0                   0
## 251           1                1                   0
## 252           1                1                   0
## 253           1                1                   0
## 254           1                1                   0
## 255           1                1                   0
## 256           1                1                   0
## 257           1                1                   0
## 258           1                1                   0
## 259           1                1                   0
## 260           1                1                   0
## 261           1                1                   0
## 262           1                1                   0
## 263           1                1                   0
## 264           1                1                   0
## 265           1                1                   0
## 266           1                1                   0
## 267           1                1                   0
## 268           1                1                   0
## 269           1                1                   0
## 270           1                1                   0
## 271           1                1                   0
## 272           1                1                   0
## 273           1                1                   0
## 274           1                1                   0
## 275           1                1                   0
## 276           1                1                   0
## 277           1                1                   0
## 278           1                1                   0
## 279           1                1                   0
## 280           1                1                   0
## 281           1                1                   0
## 282           1                1                   0
## 283           1                1                   0
## 284           1                1                   0
## 285           1                1                   0
## 286           1                1                   0
## 287           1                1                   0
## 288           1                1                   0
## 289           1                1                   0
## 290           1                1                   0
## 291           1                1                   0
## 292           1                1                   0
## 293           1                1                   0
## 294           1                1                   0
## 295           1                1                   0
## 296           1                1                   0
## 297           1                1                   0
## 298           1                1                   0
## 299           1                1                   0
## 300           1                1                   0
## 401           1                0                   1
## 402           1                0                   1
## 403           1                0                   1
## 404           1                0                   1
## 405           1                0                   1
## 406           1                0                   1
## 407           1                0                   1
## 408           1                0                   1
## 409           1                0                   1
## 410           1                0                   1
## 411           1                0                   1
## 412           1                0                   1
## 413           1                0                   1
## 414           1                0                   1
## 415           1                0                   1
## 416           1                0                   1
## 417           1                0                   1
## 418           1                0                   1
## 419           1                0                   1
## 420           1                0                   1
## 421           1                0                   1
## 422           1                0                   1
## 423           1                0                   1
## 424           1                0                   1
## 425           1                0                   1
## 426           1                0                   1
## 427           1                0                   1
## 428           1                0                   1
## 429           1                0                   1
## 430           1                0                   1
## 431           1                0                   1
## 432           1                0                   1
## 433           1                0                   1
## 434           1                0                   1
## 435           1                0                   1
## 436           1                0                   1
## 437           1                0                   1
## 438           1                0                   1
## 439           1                0                   1
## 440           1                0                   1
## 441           1                0                   1
## 442           1                0                   1
## 443           1                0                   1
## 444           1                0                   1
## 445           1                0                   1
## 446           1                0                   1
## 447           1                0                   1
## 448           1                0                   1
## 449           1                0                   1
## 450           1                0                   1
## attr(,"assign")
## [1] 0 1 1
## attr(,"contrasts")
## attr(,"contrasts")$Population
## [1] "contr.treatment"
```

This is a lot of output, but it is helpful. The columns correspond to the coefficients of the model, and the rows correspond to each row of the dataframe. The column

"(Intercept)" has a '1' for each row, because the calculation for all factor levels includes the mean for Magnolia. The column "PopulationNahant" has a '1' only for those rows in the Nahant group, and the column "Population Wellfleet" has a '1' only for those rows in the Wellfleet group. In other words, the model estimates the mean Length for the Magnolia population as (Intercept), the mean Length for the Nahant population as (Intercept) + PopulationNahant; and the mean Length for the Wellfleet population as (Intercept) + PopulationWellfleet.

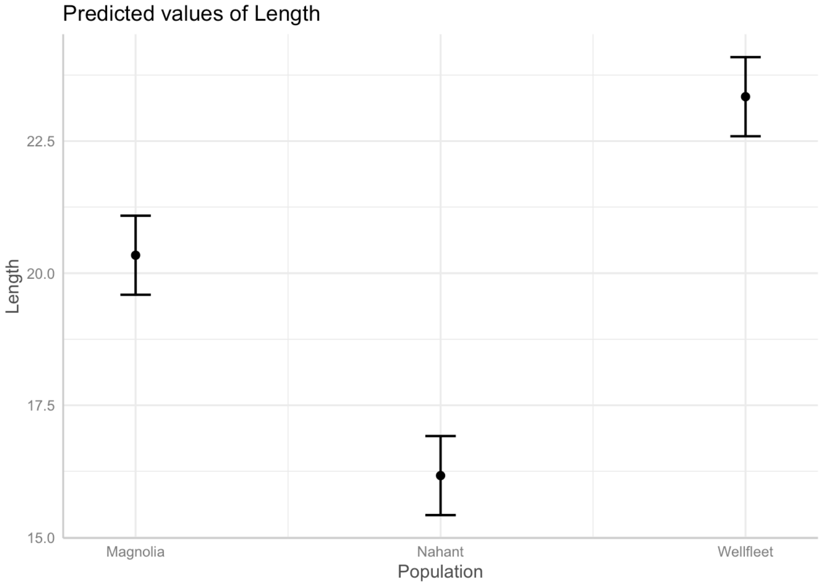
OK, now we have a sense for what lm() does when we fit an Anova. How do we interpret the results? It is usually a good idea to visualize the fitted model. A natural choice is to plot the parameter estimates, plus the standard error of the parameter estimates, which gives us a sense of the uncertainty in the parameter. The function ggcoef_model() in the package "GGally" is one of many ways to do this:

```r
library(GGally)

ggcoef_model(model)
```



This basically plots the same info returned by summary(). The (Intercept) coefficient is omitted by default (you can change this), which is why Magnolia has a 'Beta' of zero, and both of the other parameters appear to be strongly different from zero, indicating that Nahant is smaller than Magnolia, and Wellfleet is larger than Magnolia. Although this visualizes how the linear model works, for an ANOVA what we usually want to visualize is the mean of the different treatments. Plotting the model-estimated means (and uncertainty around those estimates) for each population is easily done with the package 'ggeffects':

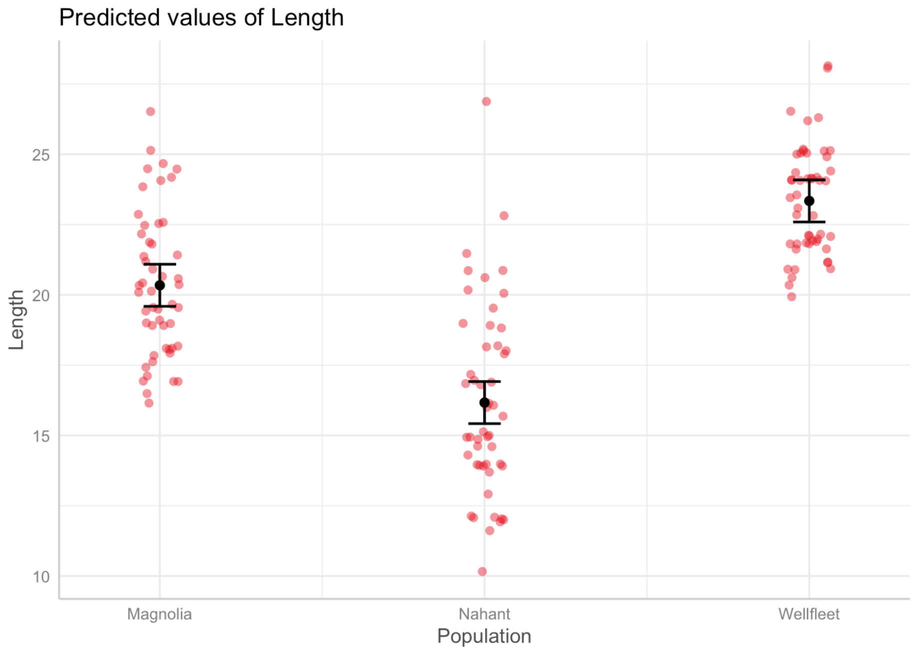
```
library(ggeffects)
ggeffect(model) %>% plot()
```



By default the ggpredict() function extracts model estimated means for each group, as well as standard errors or confidence intervals for those means (default is 95% CI). As we noted above, the model is not actually parameterized in terms of those means, but the group means and confidence intervals are easily calculated from the model-fitted coefficients.

For a model this simple it can be nice to include the raw data as well:

```
ggeffect(model) %>% plot(add.data = TRUE)
```



Note that the 95% CI error bars are narrow compared to the range of the data, because they represent *confidence in the values of the means* (strictly speaking, 95% of the time the true value of the mean is within the interval). This is different than an interval containing 95% of the data.

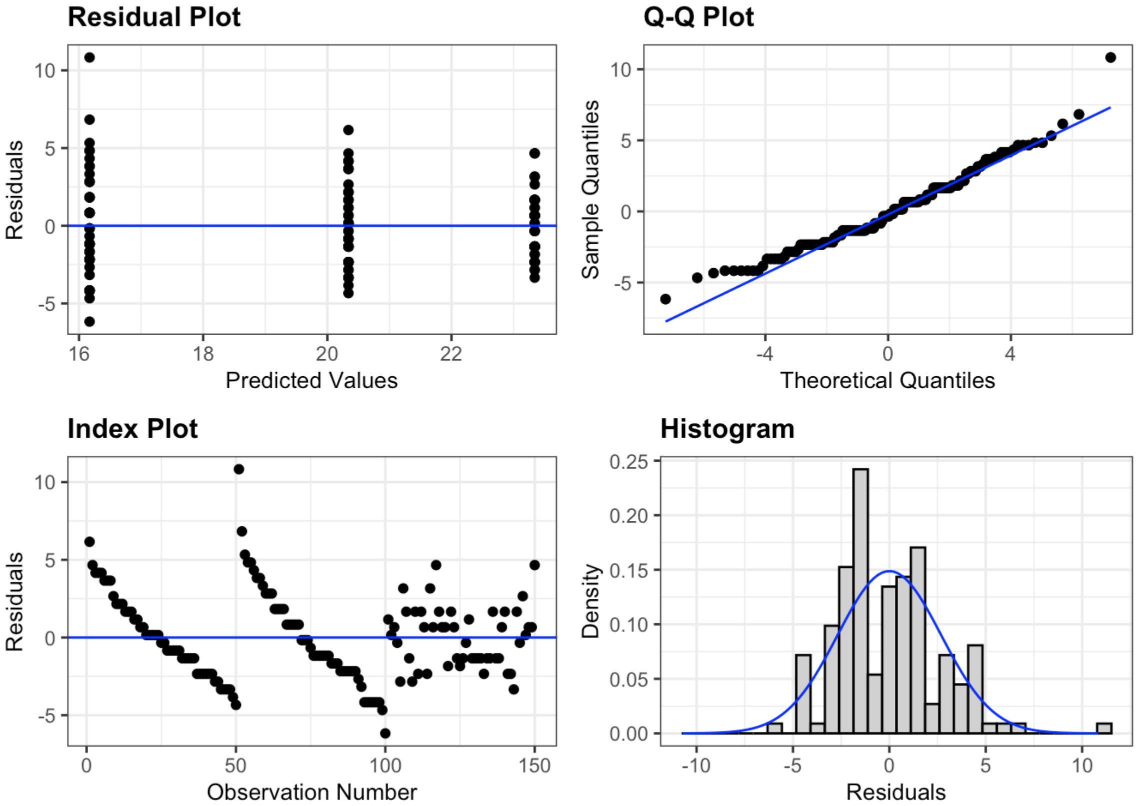
Next we want an appropriate test for whether there is significant variation between populations. The appropriate test is an F-test, and there are a couple different functions that can produce this. Here I will introduce the Anova() function from the "car" package, because it is useful for a variety of models.

```
#load the 'car' package
library(car)
Anova(model)

## Anova Table (Type II tests)
##
## Response: Length
##             Sum Sq  Df F value Pr(>F)
## Population   1297   2      89 <2e-16 ***
## Residuals    1071 147
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The F-test for Population is highly significant.

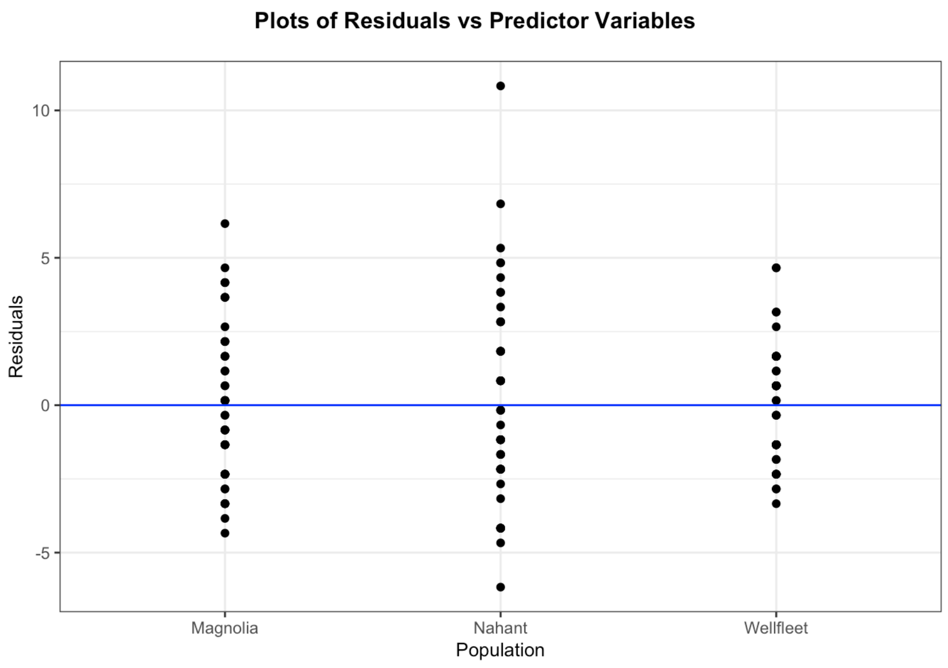
One last thing we should look at is model diagnostics, i.e. whether the data fit the assumptions of the analysis. There are many functions for doing this, the package ggResidpanel is particularly convenient:

```r
library(ggResidpanel)
resid_panel(model)
```



The plot on the top right is a quantile-quantile plot, which compares quantiles of standardized residuals to their expected value. Because the empirical quantiles for the residuals are close to the predicted line, the residuals are approximately normal. This can also be seen from the residuals histogram on the bottom right. For Anova the homoscedasticity assumption means that the residual variance should be similar across groups, which can be assessed by plotting residuals against the group predictor:

```r
resid_xpanel(model) + geom_boxplot()
```

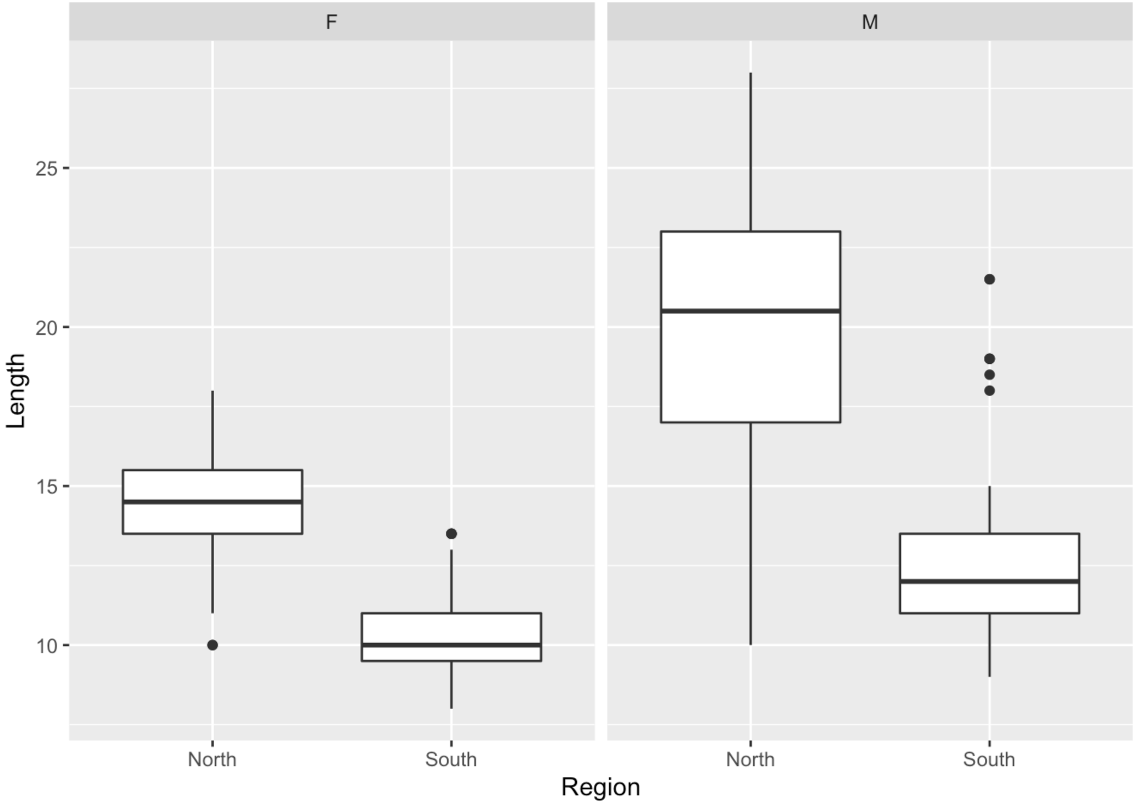


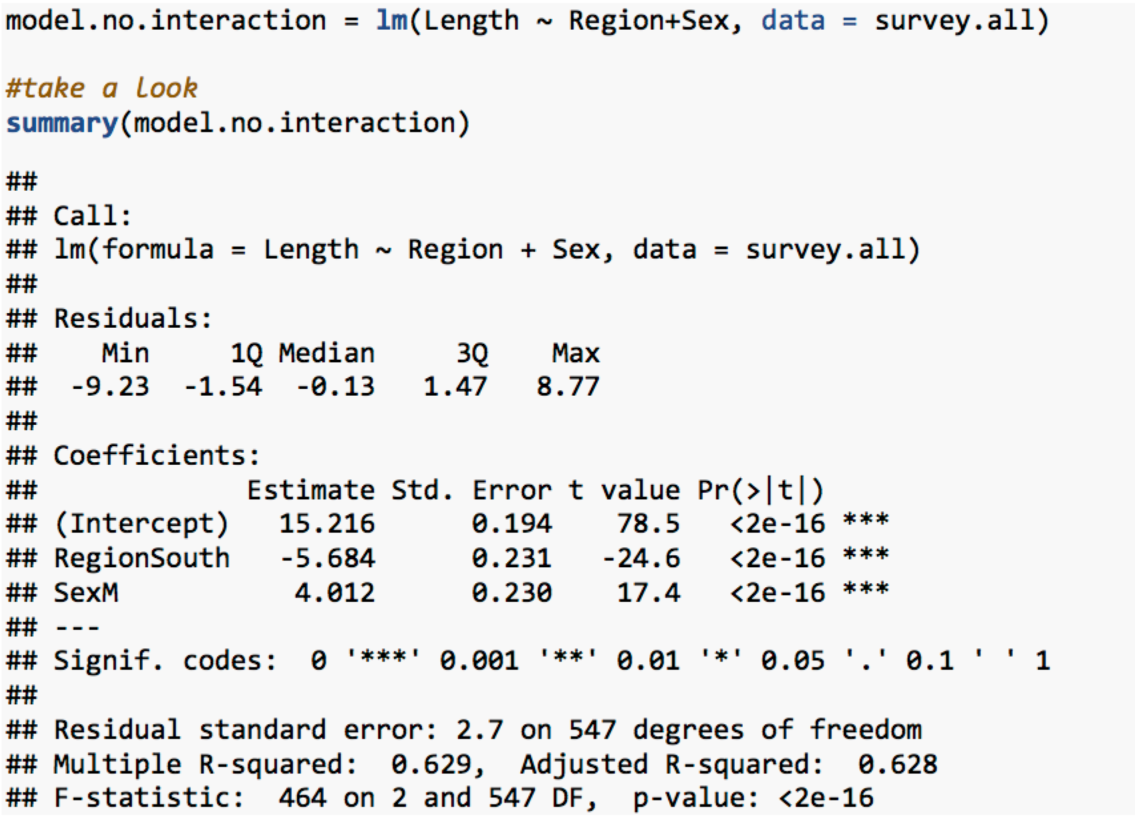
Indeed, the three populations have a similar spread of residuals. Here for diagnostics I've just made some visual comparisons. This is actually a pretty common approach, although in the past you may have used significance tests to test model assumptions (e.g. Levene's test for equality of variances). The problem with applying these tests is that if you have a small sample size you don't have enough power to really test the assumptions, and if you have a large sample size you will often reject the null hypothesis even when the data are very close to meeting the assumptions. So a good approach is to get in the habit of looking at these diagnostics, and over time you will develop an intuition for when things look problematic.

Multi-way Anova

Multi-way Anova in R is a straightfoward extension of the one-way model we just discussed. If you want to test the importance of multiple factors (e.g. multiple experimental treatments, or one treatment variable and one blocking variable), then you just add those variables to the model. To continue with the isopod dataset, we can look at whether isopod size differs between regions (North and South), while also looking at whether males and females are of different size, and whether the effect of Region differs between males and females.

```
#plot the raw data
ggplot(survey.all, aes(x = Region, y=  Length)) + geom_boxplot() +
facet_wrap(~Sex)
```





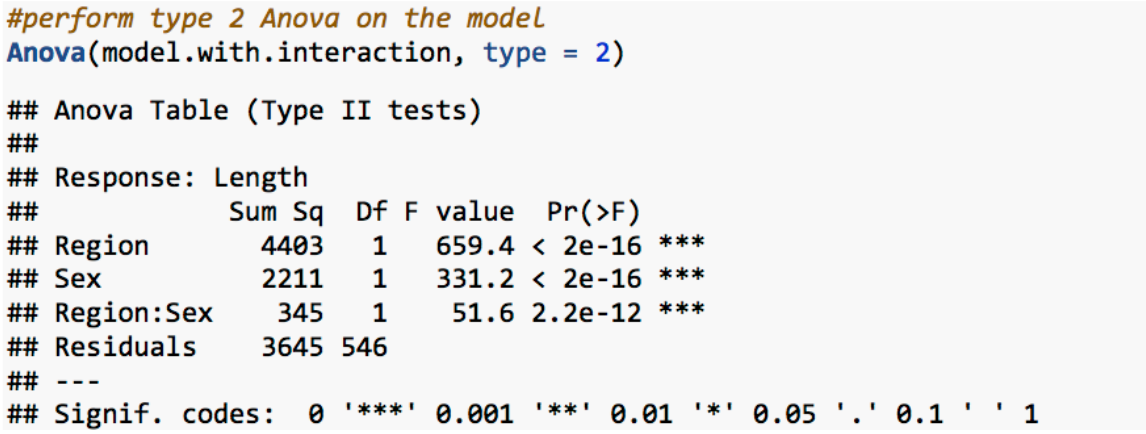
```
model.no.interaction = lm(Length ~ Region+Sex, data = survey.all)

#take a look
summary(model.no.interaction)

##
## Call:
## lm(formula = Length ~ Region + Sex, data = survey.all)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -9.23  -1.54  -0.13   1.47   8.77
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   15.216      0.194    78.5   <2e-16 ***
## RegionSouth   -5.684      0.231   -24.6   <2e-16 ***
## SexM           4.012      0.230    17.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.7 on 547 degrees of freedom
## Multiple R-squared:  0.629,  Adjusted R-squared:  0.628
## F-statistic:  464 on 2 and 547 DF,  p-value: <2e-16
```

The raw data suggest differences between Regions as well as differences between Sexes. The model includes Region and Sex, but no interaction between them. How is the model parameterized? By looking at the factor levels, as well as model.matrix() if necessary, we can see that the (Intercept) coefficient is the mean length for females from the North. RegionSouth is the difference in length between South and North, and SexM is the difference in length between males and females. To calculate the mean for males from the south, for example, we just add (Intercept) + RegionSouth + SexM.

The data also suggest that the difference between regions may be larger for males. To test these we include an interaction between factors, using an colon, or using an asterisk that is an abbreviation for all three terms:

```
#fit a model with an interaction
model.with.interaction = lm(Length ~ Region*Sex, data = survey.all)
#equivalent syntax!
#model.with.interaction = lm(Length ~ Region + Sex + Region:Sex, data =
survey.all)
```

To test whether the interaction is significant, we do an F-test with Anova():



```
#perform type 2 Anova on the model
Anova(model.with.interaction, type = 2)

## Anova Table (Type II tests)
##
## Response: Length
##             Sum Sq  Df F value  Pr(>F)
## Region        4403   1   659.4 < 2e-16 ***
## Sex           2211   1   331.2 < 2e-16 ***
## Region:Sex     345   1    51.6 2.2e-12 ***
## Residuals     3645 546
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Here I've specified "type = 2" (this is actually the default setting for this function, but I've specified it here for emphasis). What does this mean? It has to do with the appropriate way to test the effect of one predictor when there are multiple predictors in a model.

Marginal tests. In general, the kosher way to test the importance of a predictors is with a "marginal" test: is predictor A important, when all the other predictors are already accounted for? This is a conservative test, because predictors may be partially correlated with each other. For example, in the ocean temperature and nutrient concentration are negatively correlated due to stratification, so if I wanted to test the importance of nitrate concentration for some response variable (e.g. chlorophyll concentration), I would want to ask "how much variation can nitrate explain when temperature is already in the model?". In general, for classical regression or Anova models we can do marginal tests with so-called "Type II" or "Type III" tests. The difference between Type II and Type III is somewhat contentious and philosophical, but it won't really matter for this course, and I will suggest using Type II. The way Type II works is that the highest-order interactions are tested first (in our example, Region*Sex), and then the main effects of Region and Sex are each tested with the assumption that the interaction is not present. You can refer to chapter 4 of the Fox book (on Laulima) for more info.

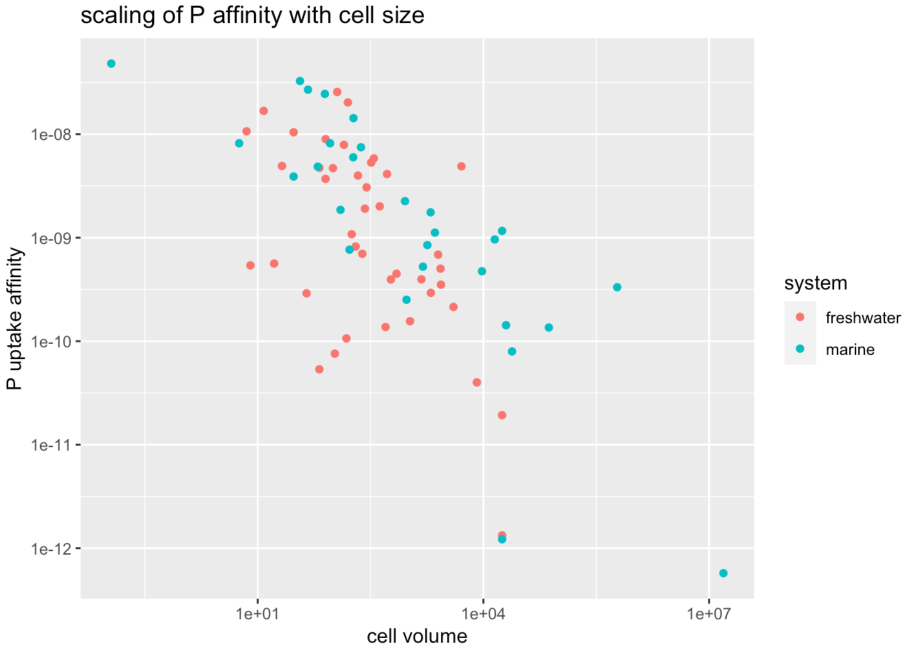
Combining categorical and continuous predictors

Now that we are getting familiar with the linear model framework, combining categorical and continuous predictors in one model is easy. This comes up in many situations. For example, you may want to test for a difference between males and females while controlling for differences in body size. This is the classic ANCOVA

(Analysis of Covariance) situation. Or, you may want to test whether the slope of some relationship differs between groups.

The data plotted below show specific phosphate affinity (a measure of competitive ability for phosphate) for a number of phytoplankton species, both marine and freshwater. Theory predicts that specific affinity should decline as cell size increases (larger cells are poorer competitors), and indeed that is what the data show. However, it is also interesting if the scaling relationship differs between marine and freshwater species, indicating a difference in how nutrient competition has evolved in those systems.

```
ggplot(nuts.ave, aes(x = volume, y = P.aff, col = system)) +
geom_point() + scale_y_log10() + scale_x_log10() + labs(x = 'cell
volume', y = 'P uptake affinity', title = 'scaling of P affinity with
cell size')
```



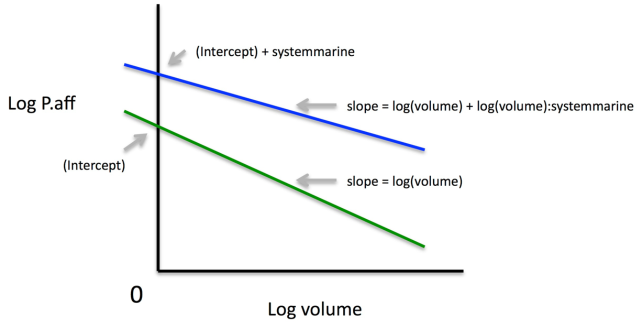
The following model effectively fits two regressions simultaneously: one for marine species, and one for freshwater species.

```
nuts.ave$logPaff = log(nuts.ave$P.aff)
nuts.ave$logVolume = log(nuts.ave$volume)
mod = lm(logPaff ~ logVolume*system, data = nuts.ave)
summary(mod)

##
## Call:
## lm(formula = logPaff ~ logVolume * system, data = nuts.ave)
##
```

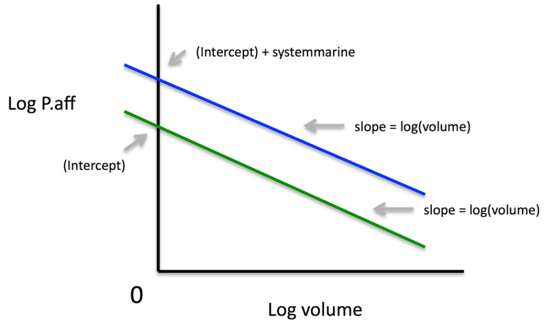
```
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -5.1650 -0.8521  0.3206  1.0421  3.3222 
## 
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)    
## (Intercept)            -17.407569   0.810767 -21.471  < 2e-16 ***
## logVolume               -0.591620   0.135210  -4.376 4.48e-05 ***
## systemmarine             0.899698   1.058600   0.850    0.399    
## logVolume:systemmarine   0.002908   0.160944   0.018    0.986    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.7 on 65 degrees of freedom
## Multiple R-squared:  0.4995, Adjusted R-squared:  0.4764 
## F-statistic: 21.62 on 3 and 65 DF,  p-value: 7.951e-10
```

How do we interpret these results? The terms with ‘logVolume’ are describing the effects of the continuous predictor, while the terms with ‘systemmarine’ are describing the difference between marine and freshwater species. Therefore, (Intercept) is the predicted value of logPaff for freshwater species when logVolume) is zero; “logVolume” is the slope for freshwater species; “systemmarine” is the difference in intercept between marine and freshwater species; and “logVolume:systemmarine” is the difference in slope between marine and freshwater species.

Continuous + Categorical: with interaction


(Intercept) + systemmarine
Log P.aff
slope = log(volume) + log(volume):systemmarine
(Intercept)
slope = log(volume)
0
Log volume

Continuous + Categorical: no interaction



To test whether the are any differences between marine and freshwater species, we just use Anova() again:

```
Anova(mod, type = 2)

## Anova Table (Type II tests)
##
## Response: logPaff
##                    Sum Sq Df F value    Pr(>F)
## logVolume         186.684  1 64.6204 2.492e-11 ***
## system             13.391  1  4.6354   0.03503 *
## logVolume:system    0.001  1  0.0003   0.98564
## Residuals         187.780 65
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It looks like the allometric scaling slope do not actually differ, suggesting that the effect of cell size is consistent across environments. In addition, there is no mean difference between the two environments when cell size is accounted for (that is the effect of "system").