May 10, 2012

# Regression Models

Perhaps the most used statistical methodology in practice is regression analysis. Basically the idea of regression is to relate one variable, called a response or dependent variable, to one or more predictor or regressor variables. This is typically done using a model. This chapter begins with the simple linear regression model and then introduces the multiple regression model with more than one predictor. Analysis of covariance (ANCOVA) models and nonlinear regression models are also covered, as well as other regression topics.

## 1 Statistical Models

The chapter on sampling dealt with the statistical problem of collecting data for observational studies. This chapter introduces the statistical problem of modeling data. Research questions generally deal with a population of interest that is defined in terms of population parameters. Statistical models can be used to describe populations of interest. Like the population, the models are defined in terms of parameters and they provide relations between variables on interest.

A very simple model for a variable $y$ is

$$y = \mu + \epsilon, \tag{1}$$

where $\mu$ can be regarded as the population mean or average and $\epsilon$ is a random error with mean zero. Suppose $y$ represents the shell length of a random selected zebra mussel from a stream or lake in Michigan. Then the model states that the length of a randomly selected zebra mussel is equal to some overall average value plus some random error. The random error is needed since no two mussels have exactly the same shell length. In other words, there is variability in shell lengths of the zebra mussels. Once a sample of data is collected, the mean shell length $\mu$ of the population of zebra mussels can be estimated, which we would typically denote by $\hat{\mu}$. An important point to keep in mind is that if we had obtained a different sample of zebra mussels, then our estimate of $\mu$ would come out differently. However, using results from statistical inference, we can quantify how much our estimator of $\mu$ will vary from sample to sample.

The simple model given in (1) is defined in terms of the model parameter $\mu$. However, there is another very important parameter not explicitly noted in (1) – the variance of the error term $\epsilon$. The variance is typically denoted by the Greek letter sigma-squared: $\sigma^2$ which is a measure of "spread" of the distribution. If $\sigma^2 = 0$, then there would be no variability of shell lengths which means all the zebra mussel would have exactly the same shell length. Of course, this is not the case. Mathematically, the variance

$\sigma^2$ is the average over the entire population of the squared deviations: $(y - \mu)^2$. The formal definition involves integral calculus for continuous random variables.

Another issue of importance when specifying the model in (1) concerns the probability distribution of the random error $\epsilon$. Often in practice it is assumed that $\epsilon$ has a normal distribution when the variable $y$ is continuous. Although this normality assumption is often approximately valid, many times it is not valid. The data should always be examined (both graphically and using statistical diagnostic tools) to access if the distribution of the error (and hence $y$) is approximately normal or skewed, or some other shape.

The primary focus of this chapter is to introduce more complicated models. In particular, we shall focus on simple and multiple regression models, nonlinear regression models. In the next chapter we will look at a more general class of regression models: generalized linear models. This class includes logistic regression and Poisson regression. We shall start with an example to review the simple linear regression model.

# 2 The Simple Linear Regression Model

There are many reasons for the vast popularity of regression models in statistical practice. One of the leading reasons is that regression models allow us to relate variables together in a mathematical form which can provide insight into the relationships between variables of interest. Related to this reason, regression models allow us to determine statistically if a response variable is related to one or more other explanatory variables. For instance, we may want to determine the effect of increasing levels of DDT on eggshell thicknesses – how does increasing levels of DDT effect eggshell thickness?. Regression models can help answer these sorts of questions. Another common use of regression models is to *predict* a response. For instance, if water is contaminated with a certain level of toxin, can we predict the amount of accumulation of this toxin in a fish that lives in the water? Once again, regression models can be used to answer such questions.

Many statistical applications deal with modeling how one variable $y$, called a response or dependent variable, depends on another variable $x$ which is called the independent or predictor variable (also called the regressor variable). In order to illustrate the basic principals and ideas, we provide an example of a simple linear regression model next.

**Example.** The population of the shrimp-like crustacean called *Diporeia* has been virtually wiped out in large areas of the lakes. The cause of this elimination is thought to be due to the relatively recent introduction of the Zebra mussel into these lakes. In studying this problem Nalepa *et al.* (2000) examine the effect of the depth (in meters) of water in southern Lake Michigan where the Diporeia were found on the dry weight (in mg) of this crustacean. Figure 1 shows a scatterplot of the data similar to the data found in the article. The data below are similar to the data in the article:

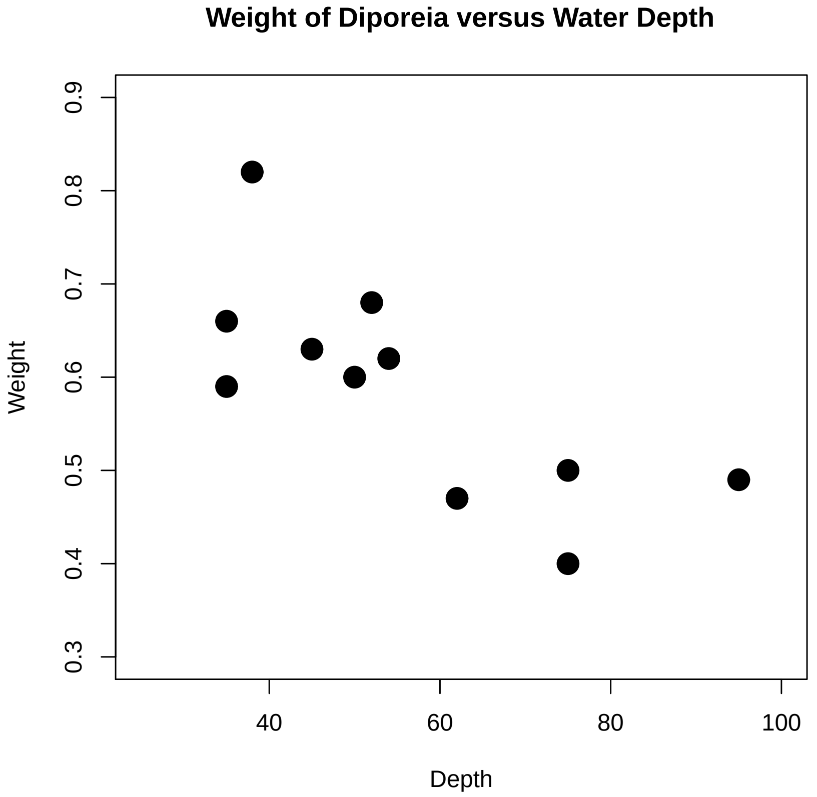

Figure 1: Scatterplot of the dry weight (in mg) of a sample of *Diporeia* versus the depth of water (in meters) in the southern portion of Lake Michigan where they were sampled.

```
Depth Weight
 35   0.59
 35   0.66
 38   0.82
 45   0.63
 50   0.60
 52   0.68
 54   0.62
 62   0.47
 75   0.40
 75   0.50
 95   0.49
```

From Figure 1, one can see a fairly strong relationship between between the weight of the Diporeia and the depth of water where the Diporeia are found. From the plot, it appears that a straight line relation between weight of Diporeia $y$ and water depth $x$ may be a reasonable way to model the data. This suggest a model of the form

$$y = \beta_0 + \beta_1 x,$$

where $\beta_0$ and $\beta_1$ are the $y$-intercept and slope of the line. The problem with this model is that it is completely deterministic. If one were to collect data on any two variables from an experiment, even if there is a linear relationship between the variables, the data points will not fall exactly on a line, as is the case in Figure 1. Thus, a probabilistic model is needed to account for the variability of points about the line. This can be achieved by adding a random error the linear relationship above:

$$y = \beta_0 + \beta_1 x + \epsilon, \tag{2}$$

where the error $\epsilon$ is a random variable with mean zero. If a sample of Diporeia is obtained, then the classical assumptions are that the random errors from (2) will be independent of each other and that they all have the same variance. If these assumptions are not approximately valid, then the statistical inference procedures need to be modified (e.g. using random effects, or weights). The model given in (2) is called the **simple linear regression model**.

## 2.1 Least-Squares Estimation

One of the primary goals of a regression analysis is to estimate the parameters $\beta_0$ and $\beta_1$ in (2). The problem of parameter estimation can be motivated by the simple model in (1) with $y_i = \mu + \epsilon_i$, for observations $i = 1, 2, \ldots, n$. One criterion for estimating $\mu$ from the data is to determine the value of $\mu$ that minimizes the sum of squares:

$$\sum_{i=1}^{n} (y_i - \mu)^2.$$

The value of $\mu$ that minimizes this sum of squares is $\hat{\mu} = \bar{y}$, the sample mean which is what is typically used to estimate the mean. In the simple linear regression framework, the same criterion is also used. That is, find the values of $\beta_0$ and $\beta_1$ that minimize the sum of squares:

$$\sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_i))^2.$$

This method is known as *least-squares* since we are finding the estimates of the parameters that make the sum of squares take the least value possible. Geometrically, finding the least-squares estimates corresponds to determining the best fitting line to the data where "best" means the line minimizing the sum of squared vertical differences between the actual data points and the corresponding points on the line as illustrated in Figure 2.

The least-squares solution can be found easily using differential calculus which we will not go into here. The least squares estimators are denoted by $\hat{\beta}_0$ and $\hat{\beta}_1$ and are given by the following formulas:

$$\begin{aligned}\hat{\beta}_0 &= \bar{y} - \hat{\beta}_1 \bar{x} \\ \hat{\beta}_1 &= \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}.\end{aligned}$$

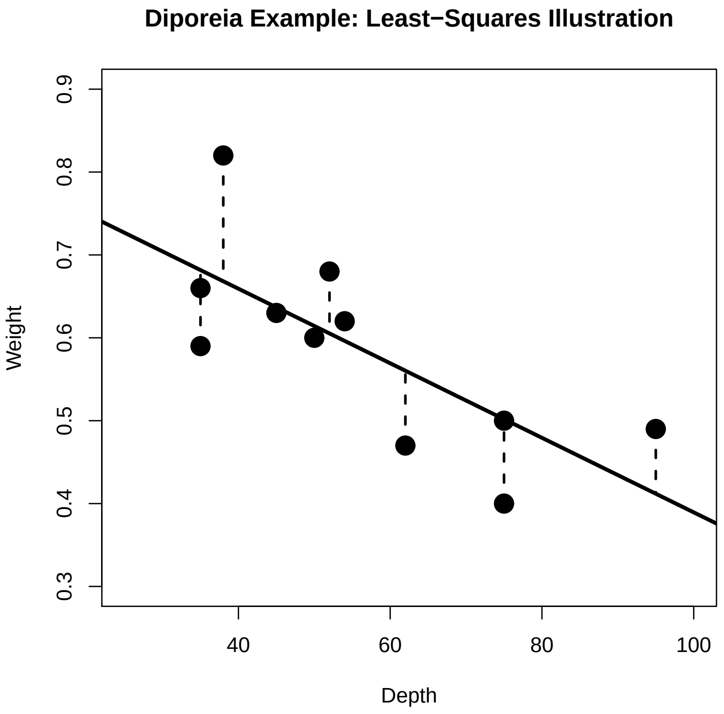

Figure 2: Illustration of the least-squares criteria for fitting a line to data. The figure shows the scatterplot of the dry weight of *Diporeia* versus the depth of water as in Figure 2 along with the least-squares regression line. The line is found by determining the best fitting line in terms of minimizing the sum of squared vertical distances between the points and the corresponding point on the line.

Note that the formula for the slope can be interpreted as the ratio of the correlation between $x$ and $y$ to the variance of $x$. Statistical software programs typically perform these computations for us. In SAS, one can use Proc Reg using the following syntax:

```
proc reg;
     model y = x;
run;
```

Linear regression models are an example of a broad class of models called *linear models* where the response $y$ is a linear function of the coefficients, in this case $\beta_1$ (although, as we shall see, we could have many coefficients). Later in this chapter we shall see how to deal with nonlinear models where the response is not linearly related to the model coefficients. The classic two-sample $t$-test and ANOVA are linear models where the regressor variables are indicators for the levels of the factors involved. In the R software, the function "lm" is typically used to fit regression models where the "lm" stands for "linear model". The R code for the *Diporeia* example is as follows:

```
dip=read.table("diporeia.dat", header=T)
names(dip)
dip
Depth=dip$Depth
Weight=dip$Weight
plot(Depth, Weight,  pch=19, cex=2, xlim=c(25,100), ylim=c(.3,.9),
  main="Weight of Diporeia versus Water Depth")
fit=lm(Weight ~ Depth, data=dip)
summary(fit)
```

We have named the results of fitting the model "fit" and "summary(fit)" prints the results of running this simple linear regression which we display here:

```
Call:
lm(formula = Weight ~ Depth, data = dip)

Residuals:
      Min        1Q    Median        3Q       Max
-0.101819 -0.056004 -0.006746  0.049235  0.151772

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.839135   0.081394  10.310 2.77e-06 ***
Depth       -0.004498   0.001382  -3.254  0.00993 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.08363 on 9 degrees of freedom
Multiple R-squared: 0.5406,     Adjusted R-squared: 0.4896
F-statistic: 10.59 on 1 and 9 DF,  p-value: 0.009926
```

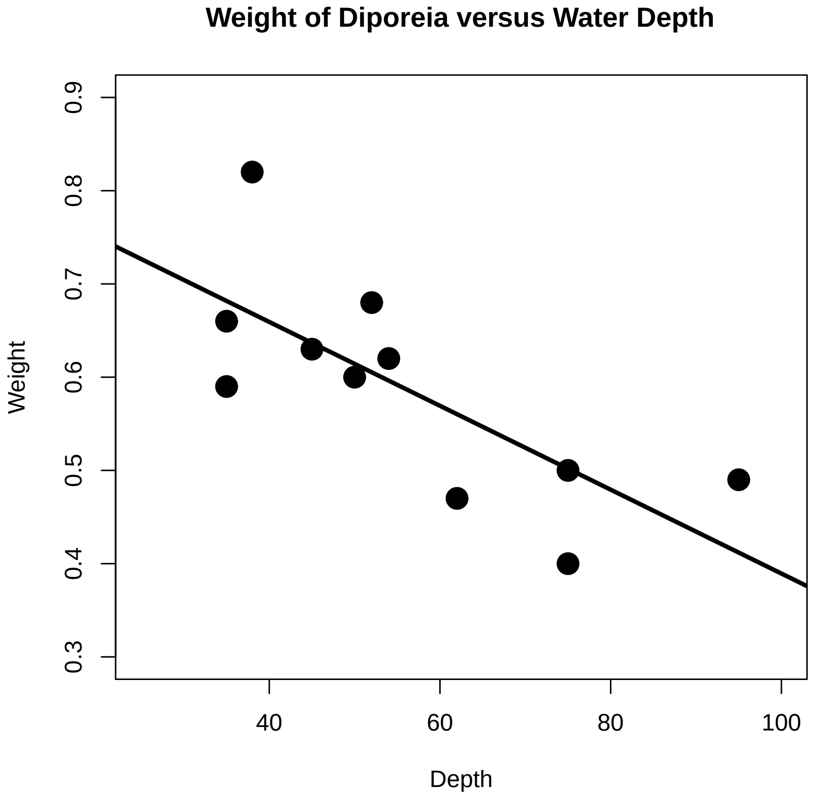

Figure 3: Scatterplot of the dry weight (in mg) of a sample of *Diporeia* versus the depth of water (in meters) along with the least-squares regression line

From this output, we see that the least-squares regression line is given by

$$\begin{aligned} \hat{y} &= \hat{\beta}_0 + \hat{\beta}_1 x \\ &= 0.83914 - 0.0045x, \end{aligned}$$

where $\hat{y}$ is known as the *predicted value* from the regression line. This estimated regression line is plotted with the raw data in Figure 3. The standard deviation of the error $\epsilon$ is estimated to be $\hat{\sigma} = 0.08363$, which R calls the residual standard error.

## 2.2 Hypothesis Testing for Simple Linear Regression

Typically, the parameter of primary importance in a simple linear regression is the slope $\beta_1$. The slope measures the average rate of change in the response variable relative to the predictor variable. Occasionally interest also lies in the $y$-intercept $\beta_0$, but usually only in cases where $x$ values are collected near the origin. Otherwise, the $y$-intercept may not have any practical meaning. From the fitted regression line, the slope is estimated to be $\hat{\beta}_1 = -0.0045$ which can be interpreted as following: the average weight of the Diporeia decreases by 0.0045 mg. at each additional meter of depth in the lake.

It is natural to ask at this point if the slope differs significantly from zero. If the slope $\beta_1$ does indeed equal zero and the model is correctly specified, then the weight of the *Diporeia* will not depend on the depth of the water and the regression line will simply be horizontal. Note that if the relation is quadratic, then one could fit a straight line and get an estimated slope near zero which could be very misleading.

Assuming the distribution of the error $\epsilon$ is normal, then the estimated regression parameters $\hat{\beta}_0$ and $\hat{\beta}_1$ will also have normal sampling distributions. The formulas for the estimated standard errors for these estimators are messy, but usually statistical software programs give the estimated standard errors automatically. From the R output above, the estimated standard errors for the intercept and slope are

$$\hat{se}(\hat{\beta}_0) = 0.08139 \text{ and } \hat{se}(\hat{\beta}_1) = 0.00138.$$

One can test the following hypotheses:

$$H_0 : \beta_0 = 0 \text{ and } H_0 : \beta_1 = 0,$$

using a $t$-test statistic:

$$t = \frac{\hat{\beta}_0}{\hat{se}(\hat{\beta}_0)} \text{ and } t = \frac{\hat{\beta}_1}{\hat{se}(\hat{\beta}_1)}.$$

If the null hypotheses are true, then the $t$-test statistics follow $t$-distributions on $n-2$ degrees of freedom (note that we lose two degrees of freedom by estimating the $y$-intercept and the slope). As long as the error distribution does not deviate to much from normality, the inference based on the $t$-distribution should be approximately correct. The R output from lm also gives the *two*-tailed $p$-values for these tests.

For example, to test if the slope in the regression model for Diporeia is zero or not, the $t$-test statistic is

$$t = \frac{\hat{\beta}_1}{\hat{se}(\hat{\beta}_1)} = \frac{-0.00450}{0.00138} = -3.25.$$

There were $n = 11$ observations in this data set and the degrees of freedom for the $t$-test statistic is $n - 2 = 11 - 2 = 9$. The resulting two-tailed $p$-value is $p = 0.0099$ indicating strong evidence that the true slope (that we are estimating) is not zero. In other words, we have strong evidence that the average weight of the diporeia does vary as a function of the depth at which they are found.

Although R reports a $p$-value for a two-tailed test, we can use the statistics to test more general hypotheses about the slope. For instance, we can test the hypothesis $H_0 : \beta_1 = \beta_1^*$ where $\beta_1^*$ is some hypothesized value of the slope. The test statistic then becomes the standardized difference between the estimated slope and the hypothetical slope:

$$t = \frac{\hat{\beta}_1 - \beta_1^*}{\hat{se}(\hat{\beta}_1)},$$

which should be compared to a $t$-distribution on $n-2$ degrees of freedom. The testing procedure of $H_0 : \beta_1 = \beta_1^*$ at a significance level $\alpha$ is given as follows: Reject $H_0$ and accept

$$
\begin{aligned}
&H_a : \beta_1 > \beta_1^* \quad \text{if} \quad t > t_{\alpha,n-2} \\
&H_a : \beta_1 < \beta_1^* \quad \text{if} \quad t < -t_{\alpha,n-2} \\
&H_a : \beta_1 \neq \beta_1^* \quad \text{if} \quad t > t_{\alpha/2,n-2} \text{ or } t < -t_{\alpha/2,n-2}
\end{aligned}
$$

**Confidence intervals for Regression Coefficients.** Confidence intervals can be computed for the $y$-intercept and the slope coefficients. A $(1-\alpha)100\%$ confidence interval for the slope $\beta_1$ is given by

$$\hat{\beta}_1 \pm t_{\alpha/2,n-2}\hat{se}(\hat{\beta}_1).$$

A confidence interval for the intercept can be computed similarly. The confidence intervals have the same interpretation as confidence intervals for any other parameter. If we compute a 95% confidence interval for $\beta_1$, then we can interpret this as a procedure that will generate an interval containing the true value of $\beta_1$ 95% of the time when repeating the experiment over and over again.

## 2.3 $R^2$: The Coefficient of Determination

One of the most important statistics in regression analysis is the *coefficient of determination*, better known as the $R^2$ ("R-squared"). In a simple linear regression $R^2$ is just the square of the correlation between $x$ and $y$. We shall give a more general definition when we discuss models with more than one predictor variable later. From the R output, we see that the $R^2 = 0.5406$. The value of $R^2$ is always between zero and one and it represents the proportion of variability in the response that is explained by the regression model. Thus, of all the variability in the weights of the *Diporeia*, 54% of this variability is explained by the different depths the animals are found.

## 2.4 Accessing the Fit of the Model

We assumed that the *Diporeia* data could be modeled adequately by a simple linear regression. In other words, the functional relationship between the mean *Diporiea* weight and water depth is linear. At this point, it may be useful to recall the famous quote of George Box (Box, 1980):

> "All models are wrong, but some are useful."

The point of the quote is that the true functional relationship between weight and depth is most likely not exactly linear and in fact probably very complicated. Thus, there is a temptation to call the model wrong. However, models are generally regarded as approximations to the truth and it is odd to call an approximation wrong. The straight line approximation for the *Diporeia* data seems to provide a decent approximation to the truth. Of course, it is not always the case that the relationship

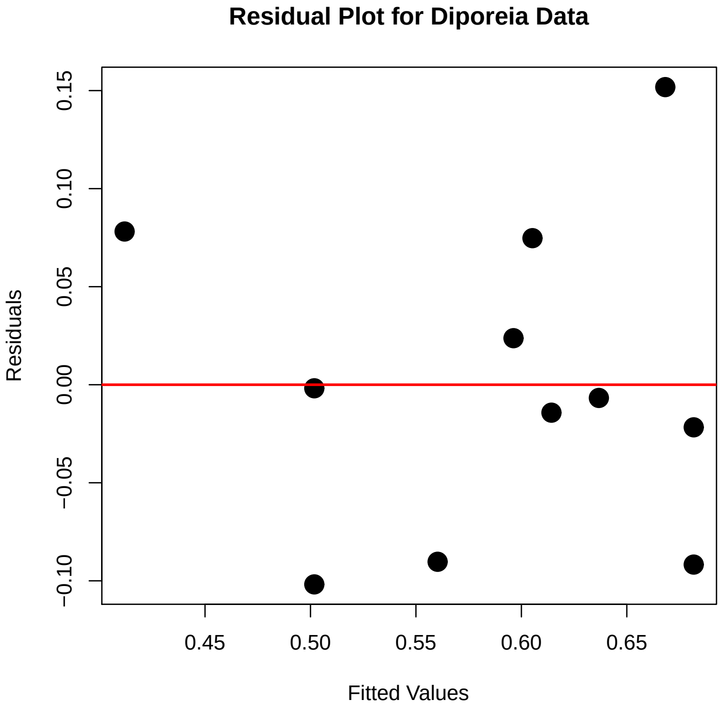

Figure 4: Residual plot for the Diporeia data versus fitted values $\hat{y}_i$.

between a response and a predictor variable will be approximated well by a straight line. If the straight line approximation is poor, then one may want to consider a different type of model. The most common way of access the fit of a regression model to data is to examine a *residual plot*.

Once the model is fitted, we can compute predicted values $\hat{y}_i$ for the $i$th observation. Ideally, we would like the straight line to provide a good fit to the data and consequently we would like to see small differences between the observed response $y_i$ and the corresponding fitted value $\hat{y}_i$. The $i$th *residual* $r_i$ is defined as the difference between these two quantities:

$$r_i = y_i - \hat{y}_i.$$

Note that the error $\epsilon$ in (2) can be expressed as $\epsilon = y - \beta_0 - \beta_1 x$. Thus, one can think of the $i$th residual as an estimate of the $i$th error term. The error in the model is suppose to be random, and consequently, the residuals from the fitted model should look like a random scatter of points. A common model diagnostic tool is to plot the residuals versus the fitted values ($r_i$ vs $\hat{y}_i$) or the residuals versus the $x_i$ values. If the model is specified correctly, then these plots should show no structure. If there is some obvious structure in a residual plot, then the model is poorly specified (perhaps the straight line relationship is inadequate). Figure 3 shows a residual plot versus predicted values.

Other types of residual plots can be used for diagnostic purposes. For instance, one could plot the standardized residuals which are simply the usual residuals divided by the square root of the mean squared error (MSE) of the regression fit. The MSE (also denoted $s^2$) is an estimate of the variance of the model error $\epsilon$ and is given by the formula:

$$\text{MSE} = \sum_{i=1}^{n} r_i^2/(n-2).$$

Assuming the error is approximately normal, about 95% of the residuals should fall within ±2 of zero. Recall that the random error $\epsilon$ has mean zero. Corresponding, the average values of the residuals is always zero. Because the residuals should show a random scatter about zero, it is helpful in residual plots to include a horizontal line at $r = 0$ as can be seen in Figure 4. Figure 4 does not show any apparent structure which indicates that there are no obvious problems with the fit of the straight line model.

Another popular transformation of residuals is the *studentized residual.* In our typical model, we assume the errors $\epsilon_i$'s all have an equal variance. However, the variances of the residuals depend on the value of the predictor variable. To get rid of this effect, one can compute a *studentized residual*:

$$\textbf{Studentized Residual}: \ \frac{r_i}{\sqrt{s^2(1-h_{ii})}}, \tag{3}$$

where $h_{ii}$ is a measure of how far $x_i$ lies from the other $x$ values (the technical definition requires some matrix notation - see Section 4.2).

**Leave-one-out.** An extreme point in the predictor space tends to pull the regression lines towards itself. This *influential* effect can mask points that have a strong influence on the estimated slope of the line (especially in the case where there are many predictor variables which is covered later) since this effect will tend to make the residual less extreme. An idea to address this problem is to use the *jacknife*: exclude the $i$th point and recompute the least-squares estimates with the $i$th point left out. Denote the least-squares estimate of the coefficient vector estimated without the $i$th point as $\hat{\boldsymbol{\beta}}_{(i)}$, where the boldface $\boldsymbol{\beta}$ represents the vector consisting of the $y$-intercept and the slope. Now we can predict the $i$th response $y_i$ using $\hat{\boldsymbol{\beta}}_{(i)}$, and denote this by $\hat{y}_{(i)}$. We can now form the residual $e_{(i)} = y_i - \hat{y}_{(i)}$, (which is known as the *PRESS* residual – see below) and standardize using the square root of the mean squared error that is also estimated with the $i$th observation left out.

Using the *leave-one-out* idea, a popular diagnostic tool is the *R-Student* (or externally studentized) residual is given by

$$t_i = \frac{e_i}{\sqrt{s_{(i)}^2(1-h_{ii})}}$$

where $s_{(i)}^2$ is the mean squared error from the model estimated without the $i$th observation. We can obtain these from SAS using "RSTUDENT=" in the output statement. One nice aspect of the leave-one-out idea in regression is that these diagnostic statistics can be computed without having to re-fit the model each time an observation is left out. Using some matrix algebra, one can show that the leave-one-out diagnostics can be obtained by fitting the model only once to the full data set. In particular, the press residual is equal to

$$e_{(i)} = \frac{e_i}{1-h_{ii}}, \tag{4}$$

where $e_i$ is the usual residual computed from the full data set.

**PRESS Statistic.** Often we use regression models to predict future observations. We can use our data to fit the model. However, it is cheating to then access how well the model predicts responses using the same data that was used to estimate the model – this will tend to give overly optimistic results in terms of how well a model is able to predict future observations. If we leave out an observation, fit the model and then predict the left out response, then this will give a less biased idea of how well the model predicts. A popular statistic for this purpose is the PRESS statistic which stands for *Prediction Error Sum of Squares* which is simply the sum of squared press residuals:

$$\text{PRESS} = \sum_{i=1}^{n} (y_i - \hat{y}_{(i)})^2. \tag{5}$$

This leave-one-out approach is a special case of *cross-validation* where we use part of the data to fit a model and part of the data to access how well the model performs. For more complicated models with several predictors, a very common problem is to choose a model to use from a large collection of possible model choices. One way this can be accomplished is to choose the model with the smallest PRESS statistic, particularly if the primary goal of the model is prediction.

The next example illustrates a case where the straight line model is clearly inappropriate which is highlighted by a clear structure in the residual plot.

**Example.** Here is an example where a simple linear regression model does not provide a good fit to the data. Walleye fish were caught in Butternut Lake, Wisconsin at three different periods. Figure 5 shows a plot of length versus age for the period before 1990 in the left panel. There is clear curvature in this plot and a straight line fit is clearly not appropriate. This is highlighted in the right panel which shows the residual plot. The residual plot is not a random scatter of points indicating that the model is not correctly specified. As the fish get older, they get longer as to be expected. An initial temptation would be to add a quadratic term to the model and fit

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \epsilon,$$

which is an example of a polynomial model. The quadratic model may provide a good approximation over a limited range of fish ages, but the model is also clearly inappropriate because it looks as if the curvature is leveling off with age which is to be expected. As organisms grow older, they generally do not continue to grow at the same rate; instead, the rate of growth slows and the regression curve should approach a horizontal asymptote.

**Estimating a Mean Response and Predicting a New Response.**

Regression models are often used to predict a new response or estimate a mean response for a given value of the predictor $x$. We have seen how to compute a predicted value $\hat{y}$ as

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x.$$

However, as with parameter estimates, we need a measure of reliability associated with $\hat{y}$.

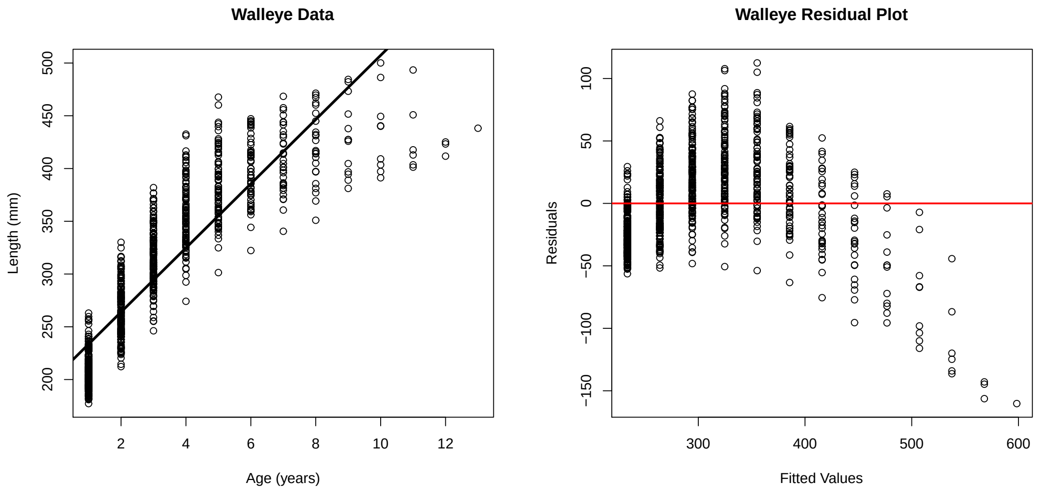

Figure 5: Walleye Data: Plot of Length versus Age along with an estimated regression line (Left Panel); The residual plot is in the right panel.

Returning to the *Diporeia* example, the fitted regression line is given by

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x = 0.83914 - 0.0045x,$$

where $x$ is the water depth and $y$ is the weight of the *Diporeia*. To illustrate, suppose we want to predict the weight of a *Diporeia* at a depth of 40 meters. Then we would simply plug $x = 40$ into the estimated regression equation to get a predicted value of $\hat{y} = 0.83914 - 0.0045(40) = 0.65914$ mg.

**Conditional Expectation**: Regression analysis is really a problem of estimating a *conditional mean or expectation*, denoted

$$E[y|x].$$

what this terminology corresponds to is the average value of the response $y$ for all units in the population with a specific $x$ value. Suppose we want to estimate the mean weight of the *Diporeia* found at a depth of 40 meters. For the prediction problem, we want to predict the weight of a single *Diporeia*. For the estimation problem, we want to estimate the mean of a *conditional* population, i.e. the population of *Diporeia* found at a depth of 40 meters. In both cases, we shall use $\hat{y} = 0.65914$ as the predicted weight and as the estimate of the mean weight of *Diporeia* found at a depth of 40 meters. In both cases, we need to associate a measure of uncertainty to these predicted and estimated values since they are based on a random sample from the population. It is customary to compute confidence intervals for an estimated mean response for a given value of $x$ and a *prediction* interval for a predicted value. The idea of a prediction interval is to determine an interval that will contain a certain percentage of the population. Because a prediction interval is attempting to capture a single, random future response as opposed to the mean of the conditional population, the prediction interval must be wider than the confidence interval. The formulas for a $(1 - \alpha)100\%$ confidence interval for a mean response at $x = x_0$ and a $(1 - \alpha)100\%$

prediction interval for predicting a new response for $x = x_0$ are given by:

$$\text{Confidence Interval for a Mean Response: } \hat{y} \pm t_{\alpha/2,n-2}\sqrt{\text{MSE}(\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{SS_{xx}})},$$

and

$$\text{Prediction Interval for a New Response: } \hat{y} \pm t_{\alpha/2,n-2}\sqrt{\text{MSE}(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{SS_{xx}})},$$

where

$$S_{xx} = \sum_{i=1}^{n} (x_i - \bar{x})^2.$$

Notice that the only difference between the confidence interval and the prediction interval is that the prediction interval has an extra term of 1 under the radical sign. This extra 1 is needed to account for the fact that we are predicting a random response $y$ at $x = x_0$ as opposed to a fixed population mean. Note also that both the confidence interval for the mean response and the prediction interval for a new response are both narrowest at $x_0 = \bar{x}$.

# 3 Correlation

One of the most used statistics in common practice is the correlation between two variables. Suppose we have data on pairs $(x_1, y_1), \ldots, (x_n, y_n)$. The sample *correlation*, which we shall denote by $r$, is a measure of the strength of the linear relation between $x$ and $y$. The formula for the sample correlation is:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}. \tag{6}$$

Thus, the correlation is the covariance between $x$ and $y$ divided by the standard deviations of $x$ and $y$. The sample correlation $r$ is an estimate of the population correlation, which is typically denoted by the Greek letter $\rho$. A formal definition of the population correlation requires integral calculus for continuous variables. However, one can think of the population correlation as computed by formula (6) by summing over the entire population of values. Two properties for the correlation $r$ follow:

1) $-1 \leq r \leq 1$.

2) If $r = \pm 1$, then $x$ and $y$ are perfectly related by a linear transformation, that is, there exists constants $a$ and $b \neq 0$ so that $y = a + bx$.

Property (1) highlights the fact that the correlation is a unitless quantity. Property (2) highlights the fact that the correlation is a measure of the strength of the *linear* relation between $x$ and $y$. A perfect linear relation produces a correlation of 1 or $-1$. A correlation of zero indicates no linear relation between the two random variables. Figure 6 shows scatterplots of simulated data obtained from bivariate distributions

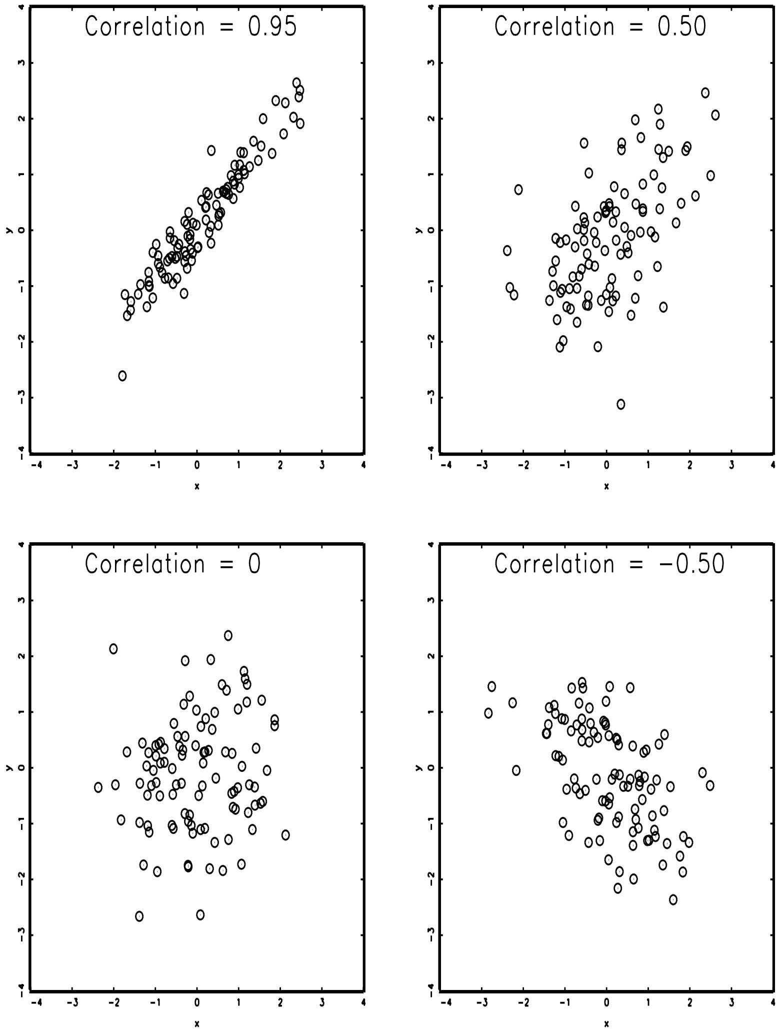

Figure 6: Scatterplots of data obtained from bivariate distributions with different correlations.

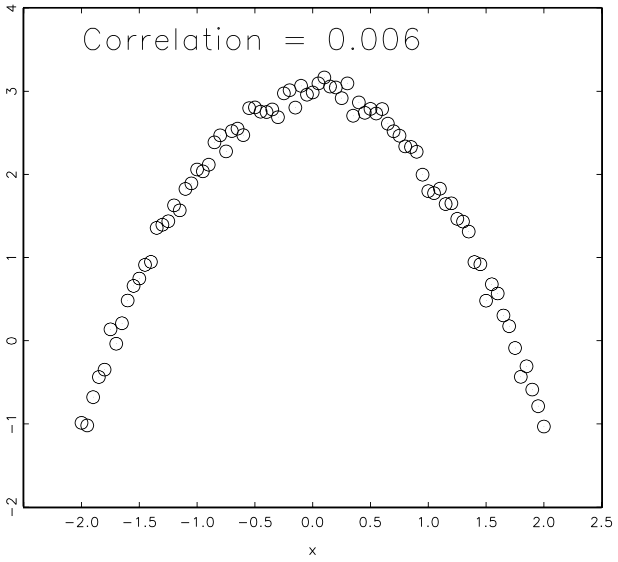

Figure 7: A scatterplot showing a very strong but nonlinear relationship between $x$ and $y$. The correlation is nearly zero.

with different correlations. The distribution for the top-left panel had a correlation of $\rho = 0.95$. The plot shows a strong positive relation between $x$ and $y$ with the points tightly clustered together in a linear pattern. The correlation for the top-right panel is also positive with $\rho = 0.50$ and again we see a positive relation between the two variables, but not as strong as in the top-right panel. The bottom-left panel corresponds to a correlation of $\rho = 0$ and consequently, we see no relationship evident between $x$ and $y$ in this plot. Finally, the bottom-right panel shows a negative linear relation with a correlation of $\rho = -0.50$.

A note of caution is in order: two variables $x$ and $y$ can be strongly related, but the relation may be nonlinear in which case the correlation may not be a reasonable measure of association. Figure 7 shows a scatterplot of data from a bivariate distribution. There is clearly a very strong relation between $x$ and $y$, but the relation is nonlinear. The correlation is not an appropriate measure of association for this data. In fact, the correlation is nearly zero. To say $x$ and $y$ are unrelated because they are uncorrelated can be misleading if the relation is nonlinear. In R, the function "cor" will compute the sample correlation coefficient. For instance, typing "cor(Depth, Weight)" in the *Diporeia* example gives a value of $-0.7352674$, which is a fairly large negative correlation.

Here are a couple notes about correlation:

- Correlation does not necessarily imply causation. In some examples, a strong correlation is indicative of a causal relationship. For instance, the height $y$ of a plant may be highly correlated with the amount of fertilizer $x$ and it is reasonable in this example to assume that higher levels of fertilizer cause the plant to grow higher. On the other hand, consider an example of the relationship between $y$, the dollar amount of damage from a fire and $x$, the number of fire fighters that tried to put the fire out. There will almost surely be a positive correlation in this example but it would be silly to conclude that sending more fire fighters to a fire will increase the dollar amount of the damage. The reason one sees more damage at fires where there were more fire fighters is that big fires require more fire fighters and big fires cause more damage.

- In a simple linear regression, the slope is zero if and only if the correlation is zero. Thus, one can test for a correlation of zero by simply testing if the slope of the regression line is zero. If we want to test if the population correlation $\rho$ takes a value other than zero, then the testing procedure is a bit more complicated (e.g. see Montgomery and Peck, 1992, page 56).

We shall finish our discussion of the simple linear regression model with some cautionary notes on the use of regression models and a famous simulated example:

**Example: Anscombe's Regression Data.** Anscombe (1973) simulated 4 very different data sets that each produce an identical least-square regression line. One of the benefits of this example is to illustrate the importance of plotting your data. Figure 8 shows scatterplots of the 4 data sets along with the fitted regression line. The top-left panel shows a nice scatter of points with a linear trend and the regression line provides a nice fit to the data. The data in the top-right panel shows a very distinct non-linear pattern. Although one can fit a straight line to such data, the straight line model is clearly wrong. Instead one could try to fit a quadratic curve (see polynomial regression). The points in the bottom left plot all lie in a line except a single point. The least squares regression line is pulled towards this single outlying point. In a simple linear regression it is fairly easy to detect a highly unusual point as in this plot. However, in multiple regression (see next section) with several regressor variables, it can be difficult to detect extreme points graphically. There exist many diagnostic tools for accessing how influential individual points are when fitting a model. There also exist *robust* regression techniques that prevent the fit of the line to be unduly influenced by a small number of observations. The bottom-right panel shows data from a very poorly designed experiment where all but one observation was obtained at one level of the $x$ variable. The single point on the right determines the slope of the fitted regression line. The bottom-right panel demonstrates how a single point can be very *influential* when a least-squares line is fit to the data.

## 3.1 Cautionary Notes on the Use of Linear Regression

1. **Prediction.** It is dangerous to use a regression model to predict responses outside the range of data that was used to predict the model. We may see a nice linear relationship between the response $y$ and the predictor $x$ in our data, but we may not know if this linear relationship persists for values of $x$ and $y$ outside the range we have observed. A stark illustration of this danger is shown in Figure 9. This plot shows the winning times for the men and women runners in the Boston marathon over the last century (note that women were not allowed into the race until the mid-70's). From the plot we see that there was a dramatic improvement in the winning times for women in the first several years and then

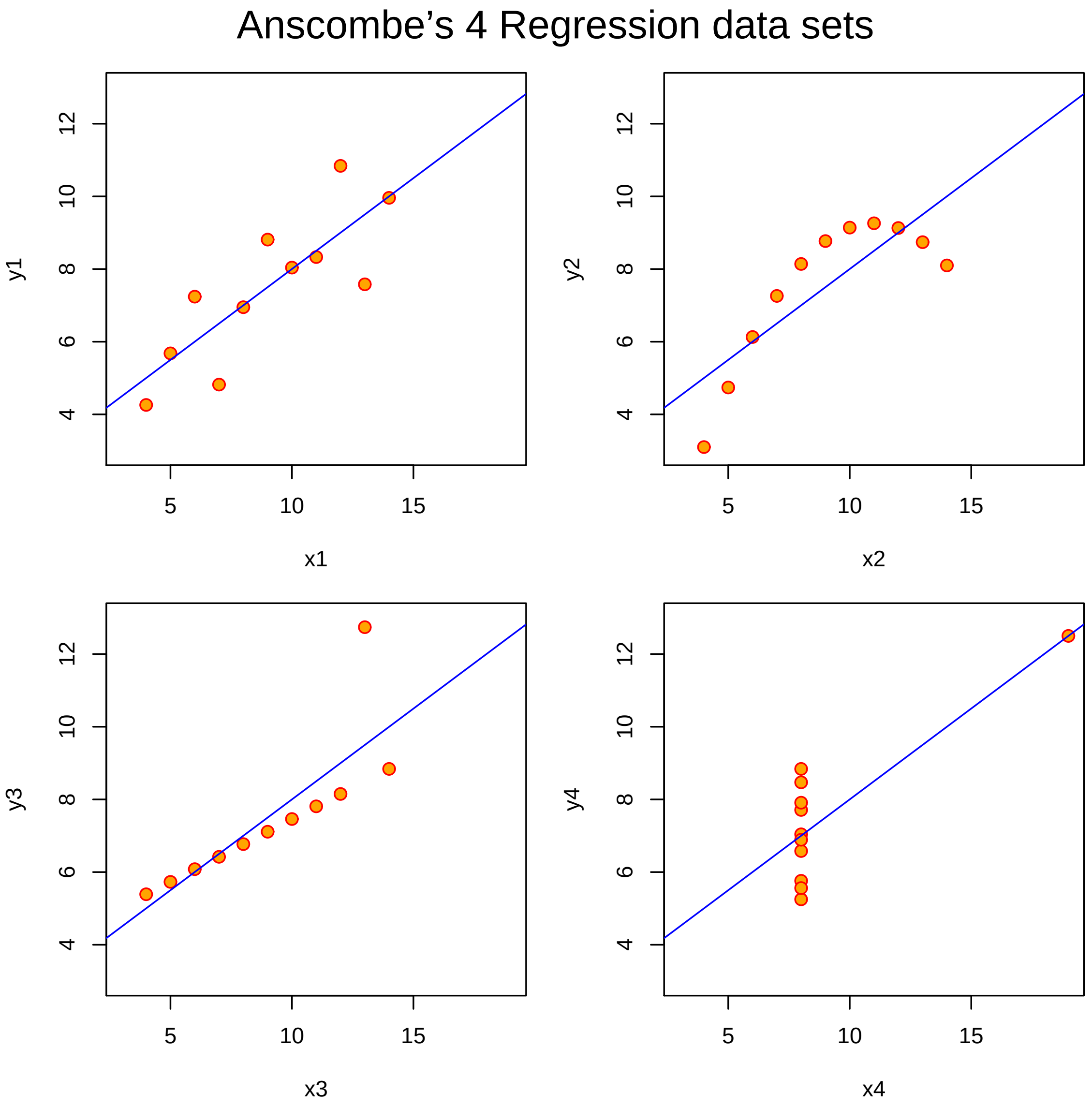

Figure 8: Anscombe simple linear regression data. Four very different data sets yielding exactly the same least squares regression line.

the rate of improvement levels off. If the rate of improvement changes, then a linear function is not appropriate. The estimated regression lines are extended beyond the range of the data and we see that the lines eventually cross. If we were to use the models to extrapolate, then we would conclude that top woman runner will eventually beat the top male runner. This may happen, but we cannot use these regression models to make this claim because the models are not valid for predicting future winning times far out into the future. In fact, suppose we extrapolated further into the future – then eventually both lines would cross the $x$-axis giving negative winning times in the marathon! This is clearly impossible. The appropriate model for the Boston marathon data probably requires a horizontal asymptote for both the men and women's data. Recalling George Box's quote, the straight line model is wrong. However, even for complex nonlinear relationships, a straight line model may provide a very good approximation over a limited range of the regressor variable.

2. The least-squares regression line can be highly influenced by one or more points. For instance, in the bottom right frame of Figure 8, there is a single point determining the slope of the regression line. This is a very unstable situation and very undesirable. If we moved that point down, then the slope would change from positive to negative. Because the estimated regression line can be strongly influenced by just a few points, it is very important to plot your data to determine if this could be a problem. A residual plot is very useful for identifying some outliers – if a point has a large residual, then there could be problem with that point (e.g. maybe its value was recorded incorrectly). *However*, note that the influential point in the bottom right frame of Figure 8 will have residual of zero. This illustrates that the residual plot by itself will not necessarily identify all problematic points from your data. Influential points have the tendency to *pull the regression line* towards themselves which in turn makes their residuals small. There are numerous regression diagnostic tools for identifying influential points. In a simple linear regression, a simple plot of the data will often suffice for identifying problematic points, but in multiple regression (next section), the data is higher dimensional and thus it is more difficult to find influential points graphically. One simple and popular idea is the *jackknife* idea mentioned earlier. Recall that the idea behind the jackknife is to leave out a single observation and fit the regression equation with out this single point and see how much the estimated coefficients change. Now repeat this process for all the data points and see which data points cause a big change in the fit of the model when left out. There also exist *robust* regression fitting techniques that were developed in the hopes that the resulting regression line will not be unduly influenced by a small number of points.

# 4 Multiple Regression

The simple linear regression model

$$y = \beta_0 + \beta_1 x + \epsilon,$$

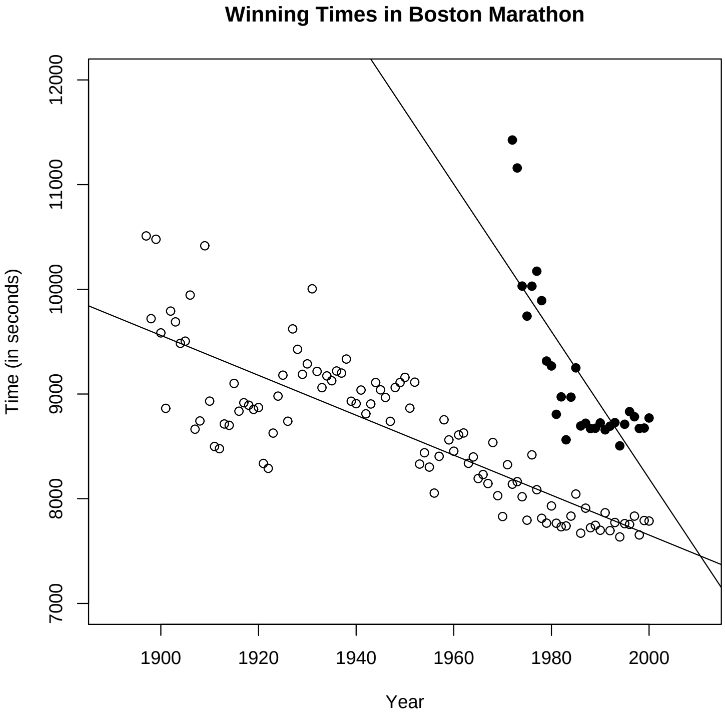

Figure 9: Winning times in the Boston Marathon versus year for men (open circles) and women (solid circles). Also plotted are the least-squares regression lines for the men and women champions.

with a single predictor variable can be easily generalized to handle models with $p > 1$ predictor variables $x_1, \ldots, x_p$ by

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \epsilon. \tag{7}$$

In order perform statistical inference using a multiple regression model, several assumptions are made. We require that the $n$ observations are independent and that the variability of the error $\epsilon$ is constant for all values of the regressors. In addition, many of the statistical tests require that the error distribution is normal. If the error distribution deviates somewhat from normality, the inference procedures will remain approximately valid.

The slope parameters $\beta_j$ are estimated using least-squares, as was done in simple linear regression, by determining the values of the $\beta_j$'s that minimize

$$\text{SSE} = \sum_{i=1}^{n} \{y_i - (\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip})\}^2.$$

Here, the index $i$ represents the observation number, $i = 1, \ldots, n$. The formulas for obtaining the least-squares estimators are derived from multivariate calculus and and closed form formulas are obtained using linear algebra. The calculations for determining the estimated slope parameters are built into most statistical software packages.

Similar to an Analysis of Variance (ANOVA), the total variability in the response variable $y$ is partitioned into an error component and the component that is explained by the regressor variables. The total sum of squares is

$$\text{SST} = \sum_{i=1}^{n} (y_i - \bar{y})^2,$$

where $\bar{y}$ is the overall mean value of the response variable. Writing

$$(y_i - \bar{y}) = (\hat{y}_i - \bar{y}) + (y_i - \hat{y}_i)$$

shows how to partition the variability in the response into the part due to the regression and the part due to error. The $r_i = (y_i - \hat{y}_i)$ is just the $i$th residual. If we square both sides of the previous equation and sum over all $i = 1, \ldots, n$ observations, we get

$$\text{SST} = \text{SSR} + \text{SSE},$$

where SSR stands for the *regression sum of squares* and is given by

$$\text{SSR} = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2,$$

and the error sum of squares (SSE) is

$$\text{SSE} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2.$$

A statistic that is almost always reported in the multiple regression examples is the $R^2$ ("R-squared"). $R^2$ represents the proportion of variability in the response explained by the regressor variables. $R^2$ is known as the *coefficient of determination* and is defined as

$$R^2 = \frac{\text{SSR}}{\text{SST}}.$$

Equivalently, one can write

$$R^2 = 1 - \frac{\text{SSE}}{\text{SST}}.$$

By definition, $R^2$ can take values only between 0 and 1. If $R^2 = 1$, then all the points lie exactly on a plane and SSE = 0. Often one would like the regression model to explain a large proportion of the variability in the response $y$. However, the criteria for what is considered a large $R^2$ varies across different fields of study. In many engineering applications, high $R^2$'s will be in the 90% or higher range whereas in many social science applications, an $R^2 = 0.5$ may be considered high.

There are some dangers on relying on $R^2$ to heavily when interpreting multiple regression output. For instance, a high $R^2$ in a simple linear regression model does not necessarily mean the relationship between $x$ and $y$ is linear. Also, if the model is to be used for prediction, a high $R^2$ will not necessarily indicate that the model can provide predictions with the needed precision. Nonetheless, $R^2$'s are reported very frequently in regression settings. $R^2$'s can also be useful when attempting to determine a parsimonious model for predicting $y$ from several regressor variables – various models can be compared in terms of $R^2$ where models with higher $R^2$ may be preferable.

One very important point to note is that $R^2$ can always be made larger (or at least not smaller) by including more regressor variables. Therefore, picking a model based on the largest $R^2$ is a bad idea because it will always lead to the most complicated model. If a simpler model (i.e. fewer regressor variables) explains practically the same amount of variability as a more complicated model, then the simpler model is to be preferred.

As in an ANOVA, an $F$-test can be carried out by comparing the variability explained by the regression relationship (SSR) with the error variability $SSE$. Formally, the $F$-test is a test of the null hypothesis:

$$H_0 : \beta_1 = \cdots = \beta_p = 0$$

versus

$$H_a : \text{ not all } \beta_j \text{ are zero.}$$

If $H_0$ is true, then the observed variability in $y$ due to the regressors should be comparable to the error variability. If $H_0$ is false, then the variability due to the regressors will exceed the error variability. In order to compare these two sources of variability, the respective sum of squares are normalized by dividing each by their respective degrees of freedom which yields the mean squares. The degrees of freedom for SSR is $p$ for the $p$ regressor variables and the degrees of freedom for the error SSE is $n - p - 1$. Thus,

$$\text{Mean Square for Regression} = \text{MSR} = \frac{\text{SSR}}{p},$$

and

$$\text{Mean Square Error} = \text{MSE} = \frac{\text{SSE}}{n-p-1}.$$

In order to test $H_0$, an $F$-test statistic is computed as

$$F = \frac{\text{MSR}}{\text{MSE}}.$$

$H_0$ is rejected at level of significance $\alpha$ if $F$ exceeds the $\alpha$ critical value of the $F$-distribution on $p$ numerator degrees of freedom and $n-p-1$ denominator degrees of freedom. Typically, statistical software packages will simply report the $p$-value of this test:

$$p\text{-value} = P(F > \text{observed } F \text{ value}),$$

where $F$ in the probability statement represents an $F$ random variable on $p$ and $n-p-1$ numerator and denominator degrees of freedom. If $H_0$ is rejected, then one can conclude that at least one of the regression coefficients is nonzero. If a regression coefficient $\beta_j \neq 0$, then it seems reasonable to assume that the corresponding regressor $x_j$ influences the response $y$. However, care must be taken when interpreting regression coefficients in a multiple regression framework as we shall see.

If the overall $F$ test just described rejects $H_0$, then a natural question to ask is which $\beta_j$'s differ significantly from zero? To answer this question, $t$-test statistics are computed individually for the hypotheses

$$H_0 : \beta_j = 0, \ j = 1, \ldots, p$$

using

$$t_j = \frac{\hat{\beta}_j}{\hat{se}(\hat{\beta}_j)},$$

which is to be compared to the $t$ critical value on $n-p-1$ degrees of freedom. Typically software programs will report the value of the estimated parameter, the estimated standard error, the $t$-test statistic for each of the $\beta_j$'s along with the corresponding $p$-values. The alternative hypothesis concerning $\beta_j$ can be one- or two-sided. In a multiple regression setup, the formulas for the estimated standard errors $\hat{se}(\hat{\beta}_j)$ are quite complicated to write out without the aid of matrix notation (see Section 4.2). Confidence intervals for individual slope parameters can also be computed using the estimate and its standard error, as was done in the simple linear regression setup.

The estimated slope parameters $\hat{\beta}_j$'s are random variables (or realizations of random variables if the data is already collected). Unless a carefully designed experiment is performed, the estimated slope parameters will be correlated with one another. If the coefficient estimators are correlated, then computing confidence intervals for individual coefficients can be misleading. For instance, when considering confidence intervals for coefficients $\beta_1$ and $\beta_2$ jointly, the joint region determined by two intervals is their Cartesian product which forms a rectangle. However, if $\hat{\beta}_1$ and $\hat{\beta}_2$ are correlated, then their values tend to vary jointly in an *elliptical* region in most cases. This suggests that an elliptical confidence region should be used for $(\beta_1, \beta_2)$ instead of a rectangular region. The details of elliptical confidence regions can be found in many multivariate statistics textbooks. Another point of importance is that if confidence intervals (or hypothesis tests) are considered for more than one slope coefficient, some sort of multiplicity correction should be considered, like the Bonferroni correction.

## 4.1 Partial $F$-Tests

In multiple regression, we have seen how to use an $F$-test to test the null hypothesis that all the regression coefficients are zero versus the alternative that at least one of the coefficients differs from zero. One of the important principals in estimating statistical models is that the simplest model that adequately explains the data should be used. This is sometimes known as the *principal of parsimony.* If a regression model has redundant regressors, they should be dropped out of the model. We can use an $F$-test for this purpose to test if a subset of regression coefficients are all zero. Here is the setup: consider the usual multiple regression model:

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q + \beta_{q+1} x_{q+1} + \cdots + \beta_p x_p + \epsilon,$$

where $q < p$. Our interest now is to test the null hypothesis

$$H_0 : \beta_{q+1} = \cdots = \beta_p = 0,$$

versus $H_a$ : that at least one of the $\beta_j \neq 0, q+1 \leq j \leq p$. If $H_0$ is true, then the model reduces to the *reduced model*:

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q + \epsilon \quad \text{(REDUCED MODEL)}.$$

In other words, we want to test if we can safely drop the regressors $x_{q+1}$ to $x_p$ from the model. If these regressors do not add significantly to the model, then dropping them will make the model simpler and thus preferable. An $F$-test can be used to test this hypothesis. We call the model with all the regressors the FULL model and the model with only regressors $x_1, \ldots, x_q$, the REDUCED model. The $F$-test statistic is computed as

$$F = \frac{\text{(SSR(Full) - SSR(Reduced))}/(p-q)}{\text{MSE(full)}}. \tag{8}$$

If the null hypothesis is true, then this $F$-test statistic follows an $F$ distribution on $p-q$ numerator degrees of freedom and $n-p-1$ denominator degrees of freedom. SSR(Full) is the regression sum of squares from the full model and SSR(Reduced) is the regression sum of squares from the reduced model. The denominator of this $F$-statistic is the MSE from the full model. In order to perform this test in practice, simply run the full and reduced models and plug the resulting sum of squares and MSE into the $F$-test statistic formula given in (8). The next example illustrates the procedure.

**Example.** (*open mercury.r.*) Mercury contamination is a serious health problem for humans. A study of mercury contamination in Largemouth bass from 53 different Florida lakes was undertaken to examine the factors that are associated with the level of mercury contamination. Water samples were collected from the surface of the middle of each lake in the early 1990's. The pH level, the amount of chlorophyll, calcium, and alkalinity were measured in each sample. From fish samples, the average mercury concentration in a three year old fish was estimated for each lake. (This data was made available on the Data and Story Library (DASL) found at http://lib.stat.cmu.edu/DASL/DataArchive.html.) The data are below:

| Observation | alkalinity | ph | calcium | chlorophyll | mercury |
|---:|---:|---:|---:|---:|---:|
| 1 | 5.9 | 6.1 | 3.0 | 0.7 | 1.53 |
| 2 | 3.5 | 5.1 | 1.9 | 3.2 | 1.33 |
| 3 | 116.0 | 9.1 | 44.1 | 128.3 | 0.04 |
| 4 | 39.4 | 6.9 | 16.4 | 3.5 | 0.44 |
| 5 | 2.5 | 4.6 | 2.9 | 1.8 | 1.33 |
| 6 | 19.6 | 7.3 | 4.5 | 44.1 | 0.25 |
| 7 | 5.2 | 5.4 | 2.8 | 3.4 | 0.45 |
| 8 | 71.4 | 8.1 | 55.2 | 33.7 | 0.16 |
| 9 | 26.4 | 5.8 | 9.2 | 1.6 | 0.72 |
| 10 | 4.8 | 6.4 | 4.6 | 22.5 | 0.81 |
| 11 | 6.6 | 5.4 | 2.7 | 14.9 | 0.71 |
| 12 | 16.5 | 7.2 | 13.8 | 4.0 | 0.51 |
| 13 | 25.4 | 7.2 | 25.2 | 11.6 | 0.54 |
| 14 | 7.1 | 5.8 | 5.2 | 5.8 | 1.00 |
| 15 | 128.0 | 7.6 | 86.5 | 71.1 | 0.05 |
| 16 | 83.7 | 8.2 | 66.5 | 78.6 | 0.15 |
| 17 | 108.5 | 8.7 | 35.6 | 80.1 | 0.19 |
| 18 | 61.3 | 7.8 | 57.4 | 13.9 | 0.49 |
| 19 | 6.4 | 5.8 | 4.0 | 4.6 | 1.02 |
| 20 | 31.0 | 6.7 | 15.0 | 17.0 | 0.70 |
| 21 | 7.5 | 4.4 | 2.0 | 9.6 | 0.45 |
| 22 | 17.3 | 6.7 | 10.7 | 9.5 | 0.59 |
| 23 | 12.6 | 6.1 | 3.7 | 21.0 | 0.41 |
| 24 | 7.0 | 6.9 | 6.3 | 32.1 | 0.81 |
| 25 | 10.5 | 5.5 | 6.3 | 1.6 | 0.42 |
| 26 | 30.0 | 6.9 | 13.9 | 21.5 | 0.53 |
| 27 | 55.4 | 7.3 | 15.9 | 24.7 | 0.31 |
| 28 | 3.9 | 4.5 | 3.3 | 7.0 | 0.87 |
| 29 | 5.5 | 4.8 | 1.7 | 14.8 | 0.50 |
| 30 | 6.3 | 5.8 | 3.3 | 0.7 | 0.47 |
| 31 | 67.0 | 7.8 | 58.6 | 43.8 | 0.25 |
| 32 | 28.8 | 7.4 | 10.2 | 32.7 | 0.41 |
| 33 | 5.8 | 3.6 | 1.6 | 3.2 | 0.87 |
| 34 | 4.5 | 4.4 | 1.1 | 3.2 | 0.56 |
| 35 | 119.1 | 7.9 | 38.4 | 16.1 | 0.16 |
| 36 | 25.4 | 7.1 | 8.8 | 45.2 | 0.16 |
| 37 | 106.5 | 6.8 | 90.7 | 16.5 | 0.23 |
| 38 | 53.0 | 8.4 | 45.6 | 152.4 | 0.04 |
| 39 | 8.5 | 7.0 | 2.5 | 12.8 | 0.56 |
| 40 | 87.6 | 7.5 | 85.5 | 20.1 | 0.89 |

| | | | | | |
|---|---|---|---|---|---|
| 41 | 114.0 | 7.0 | 72.6 | 6.4 | 0.18 |
| 42 | 97.5 | 6.8 | 45.5 | 6.2 | 0.19 |
| 43 | 11.8 | 5.9 | 24.2 | 1.6 | 0.44 |
| 44 | 66.5 | 8.3 | 26.0 | 68.2 | 0.16 |
| 45 | 16.0 | 6.7 | 41.2 | 24.1 | 0.67 |
| 46 | 5.0 | 6.2 | 23.6 | 9.6 | 0.55 |
| 47 | 25.6 | 6.2 | 12.6 | 27.7 | 0.58 |
| 48 | 81.5 | 8.9 | 20.5 | 9.6 | 0.27 |
| 49 | 1.2 | 4.3 | 2.1 | 6.4 | 0.98 |
| 50 | 34.0 | 7.0 | 13.1 | 4.6 | 0.31 |
| 51 | 15.5 | 6.9 | 5.2 | 16.5 | 0.43 |
| 52 | 17.3 | 5.2 | 3.0 | 2.6 | 0.28 |
| 53 | 71.8 | 7.9 | 20.5 | 8.8 | 0.25 |

All the variables measured (except pH) are quite strongly skewed to the right, so the log-transformed variables were used in the analysis. The following full model was fit to the data:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \epsilon,$$

where $x_1$ to $x_4$ are the logarithms of alkalinity (mg/L as Calcium Carbonate), pH, calcium (mg/l), and chlorophyll (mg/l) respectively. The response $y$ is the average mercury concentration in a three-year old fish (parts per million). R code for fitting the full model is given by

```
hg=read.table("mercury.dat")
names(hg)=c("alkalinity", "ph", "calcium", "chlorophyll",
              "mercury", "numfish", "min", "max", "mer", "age")
head(hg)
lalk=log(hg$alkalinity)
lcalcium=log(hg$calcium)
lchlor=log(hg$chlorophyll)
lmer=log(hg$mer)
ph=hg$ph
fitfull=lm(lmer ~ lalk + ph + lcalcium + lchlor)
summary(fitfull)
```

The result of the summary command is as follows:

```
Call:
lm(formula = lmer ~ lalk + ph + lcalcium + lchlor)

Residuals:
     Min       1Q   Median       3Q      Max
-1.44731 -0.26100  0.06144  0.33606  1.40865

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept)  0.715904   0.486492   1.472  0.14766
lalk        -0.471477   0.136730  -3.448  0.00118 **
ph           0.003489   0.115744   0.030  0.97608
lcalcium     0.104484   0.118089   0.885  0.38068
lchlor      -0.207699   0.075299  -2.758  0.00820 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.5277 on 48 degrees of freedom
Multiple R-squared: 0.6146,     Adjusted R-squared: 0.5825
F-statistic: 19.14 on 4 and 48 DF,  p-value: 1.811e-09
```

In this output, "lalk" stands for log(alkalinity), "lcalcium" stands for log(calcium) etc. First note that $R^2 = 0.6146$ which means that about 61% of the variability in mercury levels in the bass can be explained by these four predictors. Note that the overall $F$-test statistic if $F = 19.14$ with 4 numerator degrees of freedom (for the four predictors) and 48 denominator degrees of freedom ($48 = n-p-1 = 53-4-1$). The $p$-value for this $F$-test is given as $1.811\times10^{-9}$ which is extremely small. Therefore, there is very strong evidence that at least one of the regression coefficients differs from zero. However, from the partial $t$-test statistics for testing if individual coefficients differ from zero or not, we see that the coefficient of pH and log(calcium) give relatively large $p$-values of $p = 0.9761$ and 0.3807 respectively. Ordinarily we would like to specify what hypothesis tests to perform before looking at the results of fitting the full model. However, for the sake of illustrating the partial $F$-test, we shall test $H_0 : \beta_2 = \beta_3 = 0$ versus the alternative that at least one of these coefficients differs from zero. To perform this test, we run a reduced model using only log(alkalinity) and log(chlorophyll):

```
fitreduced=lm(lmer ~ lalk + lchlor)
summary(fitreduced)
```

which produces the following output:

```
Call:
lm(formula = lmer ~ lalk + lchlor)

Residuals:
     Min       1Q   Median       3Q      Max
-1.40178 -0.24320  0.05559  0.31108  1.48473

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.69449    0.19809   3.506  0.00097 ***
lalk        -0.37877    0.07002  -5.410 1.78e-06 ***
lchlor      -0.20049    0.06688  -2.998  0.00423 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
Residual standard error: 0.5215 on 50 degrees of freedom
Multiple R-squared: 0.608,      Adjusted R-squared: 0.5923
F-statistic: 38.77 on 2 and 50 DF,  p-value: 6.808e-11
```

Note that the $R^2$ has only dropped a little in value to 0.608. We can compute the regression sum of squares needed for the partial $F$ test in R using

```
# Partial F-test (the hard way)
anova(fitfull)
names(anova(fitfull))
SSRfull = sum(anova(fitfull)$"Sum Sq"[1:4])
SSRfull

anova(fitreduced)
names(anova(fitreduced))
SSRreduced = sum(anova(fitreduced)$"Sum Sq"[1:2])
SSRreduced
```

For example, "anova(fitfull)" gives

```
Response: lmer
          Df  Sum Sq Mean Sq F value    Pr(>F)
lalk       1 18.6452 18.6452 66.9530 1.169e-10 ***
ph         1  0.3370  0.3370  1.2103  0.276762
lcalcium   1  0.2196  0.2196  0.7887  0.378933
lchlor     1  2.1188  2.1188  7.6083  0.008196 **
Residuals 48 13.3671  0.2785
```

and we see that $mse = 0.2785$ for the full model and the regression sum of squares for the full model is the sum $18.6452 + 0.3370 + 0.2196 + 2.1188$. We find that the regression sum of squares for the full and reduced models are 21.32066 and 21.08907 respectively. The partial $F$-test statistic is computed as

$$F = \frac{(21.32066 - 21.08907)/(4-2)}{0.2785} = 0.4158$$

where the denominator 0.27848 is the MSE from the full model with all four regressors. The $p$-value for this test can be computed directly using the "pf" function which gives cumulative $F$ probabilities. In particular, typing "1-pf(0.4158, 2, 48)" gives the $p$ value of $p = 0.6622$ indicating that the coefficients for pH and log(calcium) do not differ significantly from zero. This suggests that we can drop the regressors pH and log(calcium) from our model and use the reduced model:

$$\hat{y} = 0.69449 - 0.37877\text{log(alkalinity)} - 0.20049\text{log(chlorophyll)}.$$

The above illustration of the partial $F$ test was a bit tedious, but that was for the purpose of illustration. We can have R perform this test directly using the command "anova(fitreduced, fitfull)" which gives the following output

```
Model 1: lmer ~ lalk + lchlor
Model 2: lmer ~ lalk + ph + lcalcium + lchlor
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     50 13.599
2     48 13.367  2   0.23158 0.4158 0.6622
```

which matches with what we already computed.

Unlike the simple linear regression model, the modeling strategy for multiple regression is often much more complex. One of the reasons for the added complexity is that the regression variables (the predictors) are often correlated with one another. In a simple linear regression, the slope of the model corresponds to the average rate of change in the response variable corresponding to changes in the predictor variable. The same interpretation exists in the multiple regression setting with one important caveat: the slope $\beta_j$ for the $j$th regressor variable $x_j$ represents the average change in the the response $y$ for a unit change in $x_j$ *provided all other regressor variables are held fixed*. The problem with slope interpretation in multiple regression is that if one regressor changes, then other regressor variables tend to change as well when the regressors are correlated. Therefore, it is very difficult to access the effect of a regressor on the response by examining an individual slope coefficient $\beta_j$ by itself. We shall discuss this issue more in the section on multicollinearity in Section 6.

To highlight the problem, consider the mercury example above again. From our modeling we decided that log(calcium) was not a significant predictor in the model and we dropped it out. However, if we run a simple linear regression of log(mercury) on log(calcium), we get the following results:

```
> fitcal=lm(lmer ~ lcalcium)
> summary(fitcal)

Call:
lm(formula = lmer ~ lcalcium)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.01289    0.19484   0.066    0.948
lcalcium    -0.39099    0.07184  -5.443 1.51e-06 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.6559 on 51 degrees of freedom
Multiple R-squared: 0.3674,     Adjusted R-squared: 0.355
F-statistic: 29.62 on 1 and 51 DF,  p-value: 1.509e-06
```

From this model, we now see that log(calcium) is highly a significant predictor of mercury levels, whereas before we seemed to be claiming it was not. The point here is that log(calcium) is indeed significant by itself. However, in a multiple regression model that also contains the other predictors, log(calcium) has little additional information to add and hence becomes insignificant.

Often the problem of fitting a multiple linear regression model is a trial and error process. One may begin by postulating a model only to find that it is inadequate because some of the estimated slope coefficients are unstable (i.e. large $p$-values when testing if the coefficient is zero) and/or because the residual plots show structure indicating the model is not appropriate. Possible approaches to obtaining an adequate model may involve dropping some variables. It may also be necessary to add higher order polynomial terms, see Section 5, and interaction terms (i.e. products of predictors).

## 4.2 Matrix Notation

In models with many predictors, the use of matrix notation makes it much easier to express the models and estimators. In the mercury in bass example, let $y_1, y_2, \ldots, y_n$ denote the observed mercury levels in the sample of fish. Then we can stack these values into a vector which we will denote using boldface as $\boldsymbol{y}$. Similarly, for the $i$th observation ($i = 1, 2, \ldots, n$), we have $p$ predictors $x_{i1}, x_{i2}, \ldots, x_{ip}$. We can form a matrix where the $i$th row corresponds to these $p$ values. We also include a value of one for the first element of the row to correspond to the intercept. By stacking these predictor values up into a matrix, we obtain a matrix whose $i$th row is

$$1 \; x_{i1} \; x_{i2} \; \ldots \; x_{ip},$$

thereby constructing a matrix with $n$ rows and $(p+1)$ columns, which we denote by $\boldsymbol{X}$. This is called the *design matrix*:

$$\boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad \text{and} \quad \boldsymbol{X} = \begin{pmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1p} \\ 1 & x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots & \\ 1 & x_{n1} & x_{n2} & \ldots & x_{np} \end{pmatrix}.$$

Similarly, we can stack up the coefficients and the errors in the model by defining

$$\boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} \quad \text{and} \quad \epsilon = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}.$$

Then in matrix notation, the regression multiple regression model for the entire set of data can be succinctly written as

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}.$$

It follows that the least squares estimators of the regression coefficients, which we will denote by $\hat{\boldsymbol{\beta}}$ after stacking them into a vector, have a very simple formula:

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y},$$

where $(\boldsymbol{X}'\boldsymbol{X})^{-1}$ denotes the inverse of the matrix $(\boldsymbol{X}'\boldsymbol{X})$.

Earlier, in the discussion of the leave-one-out cross-validation ideas, we introduced $h_{ii}$ in (3). The matrix defined by

$$\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}' \tag{9}$$

is known as the *hat matrix* and the diagonal elements of this matrix are the $h_{ii}$.

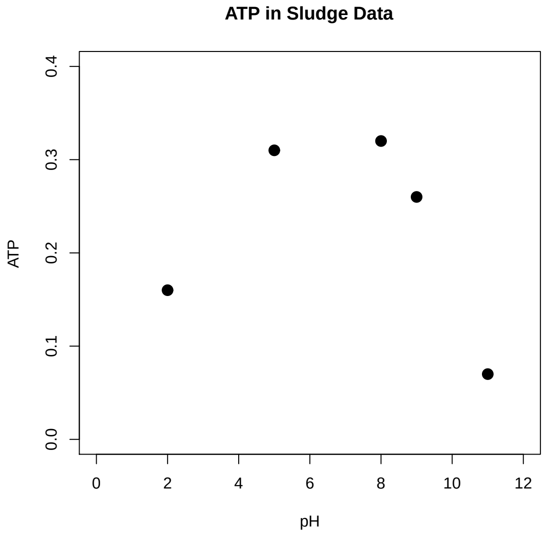

Figure 10: A plot of ATP versus pH in industrial sludge

# 5 Polynomial Regression

A very important special case of the multivariate regression model useful for situations where the relation between a response $y$ and a predictor $x$ appears nonlinear is a polynomial regression model. Suppose we are interested in a situation where a response $y$ is related to a single regressor variable $x$, but the relationship is nonlinear:

$$y = f(x) + \epsilon,$$

for some nonlinear function $f(x)$. Often in practice the functional relationship between $x$ and $y$ is unknown. However, if the function $f$ is smooth, then $f(x)$ can be approximated fairly well by a polynomial due to Taylor's theorem. Therefore, the following polynomial model often works well in practice:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_p x^p + \epsilon.$$

Note that this is just a special case of a multiple regression model with regressors $x, x^2, \ldots, x^p$. For this reason, the model is still called a *linear model* because the model is linear in the parameters $\beta_0, \ldots, \beta_p$, even though the relationship between $y$ and $x$ may be nonlinear. In other words, because we can express the polynomial relationship as a special case of a multiple regression model, polynomial regression models are linear models. We illustrate now with an example.

**ATP Sludge Example** The *Adenosine triphosphate* (ATP) molecule is the universal currency of energy exchange in biological systems. ATP bioluminescence assay is routinely used for microbiology quality assurance purposes. It provides rapid quantification of bacteria in biological samples. A test involving changes in ATP content used to determine toxicity of industrial waste water discharged to a sewer was conducted. ATP changes in the organism *Photobacterium Phosphoreum* in industrial sludge was measured for different levels of the pH (Arretxe et al 1997). The data in the following table were derived from plots in the Arretxe *et al.* (1997):

| pH | ATP (mg/g) |
|---|---|
| 2 | .16 |
| 5 | .31 |
| 8 | .32 |
| 9 | .26 |
| 11 | .07 |

A plot of ATP versus pH for this data is in Figure 10 which shows a clearly nonlinear trend. Fitting a straight line to these data is clearly not a good idea. Instead, we could try to fit a quadratic model

$$\text{ATP} = \beta_0 + \beta_1(\text{pH}) + \beta_2(\text{pH})^2 + \epsilon.$$

The following R code fits this model:

```
atp=read.table("atp.dat", header=T)
y=atp[,2]
ph=atp[,1]
plot(ph, y, ylim=c(0,.4), xlim=c(0,12),
   main="ATP in Sludge Data", xlab="pH", ylab="ATP", pch=19, cex=1.5)

quadfit=lm(y~ ph + I(ph^2)) # Fit a quadratic model
summary(quadfit)
windows()
x=seq(0,12, by=.1)
atpcurve=coef(quadfit)[1]+coef(quadfit)[2]*x+coef(quadfit)[3]*x^2
plot(ph, y, ylim=c(0,.4), xlim=c(0,12),
   main="ATP in Sludge Data", xlab="pH", ylab="ATP", pch=19, cex=1.5)
lines(x, atpcurve, lwd=2, col=2)
```

which produces the following output:

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.077579   0.039691  -1.955   0.1898
ph           0.137719   0.014166   9.722   0.0104 *
I(ph^2)     -0.011228   0.001084 -10.355   0.0092 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.02005 on 2 degrees of freedom
Multiple R-squared: 0.9824,     Adjusted R-squared: 0.9648
F-statistic: 55.85 on 2 and 2 DF,  p-value: 0.01759
```

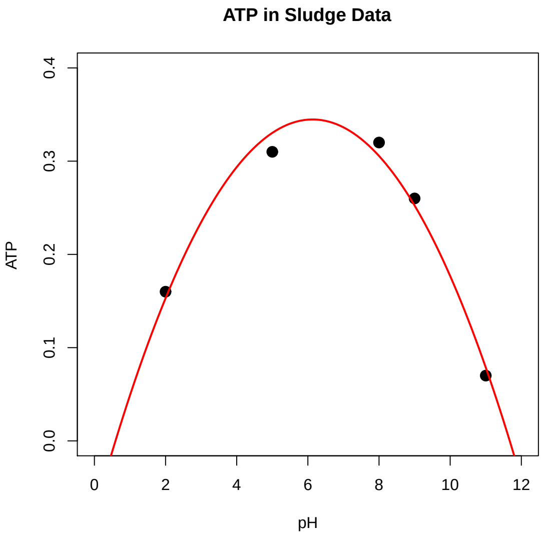

Figure 11: A plot of ATP versus pH in industrial sludge along with a fitted quadratic curve

Notice that the quadratic term in the model is highly significant with a $p$-value of 0.0092 and the $R^2 = 0.9824$ which is very large. A plot of the raw data and the fitted quadratic curve appear in Figure 11

**Cautionary Notes on the Use of Polynomial Regression.** One must use extreme care when fitting polynomial models. In the ATP example above, fitting a quadratic model is certainly a reasonable approach. However, it is tempting in practice to *overfit* models using polynomial regression. If, for example, we added a cubic term to the quadratic model (i.e. $\text{ATP} = \beta_0 + \beta_1(\text{pH}) + \beta_2(\text{pH})^2 + \beta_3(\text{pH})^3 + \epsilon$), that will increase the coefficient of determination to $R^2 = 0.9987$ which is usually considered desirable. However, this is only a small increase in $R^2$ compared to the quadratic model. The SAS output for the estimated parameters of the cubic fit is given below:

```
> fit3=lm(y ~ ph + I(ph^2) + I(ph^3))
> summary(fit3)

Call:
lm(formula = y ~ ph + I(ph^2) + I(ph^3))

Residuals:
         1          2          3          4          5
 0.0002607 -0.0013686  0.0054744 -0.0052789  0.0009124

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.0256118  0.0332474   0.770    0.582
ph           0.0674916  0.0207904   3.246    0.190
```

```
I(ph^2)      0.0010222  0.0035226   0.290    0.820
I(ph^3)     -0.0006180  0.0001764  -3.503    0.177

Residual standard error: 0.007785 on 1 degrees of freedom
Multiple R-squared: 0.9987,     Adjusted R-squared: 0.9947
F-statistic: 251.1 on 3 and 1 DF,  p-value: 0.04635
```

Note that none of the estimated coefficients are stable (all the $p$-values are large). This is clearly an undesirable model. The quadratic model fits the data very well and there is no need to further complicate the model by the inclusion of higher order polynomial terms. In fact, higher order terms of the form $x^2, x^3, x^4$ and so on tend to be highly correlated with one another which leads to very unstable estimated models. High correlations between regressor variables is the topic of the next section.

# 6 Multicollinearity

One of the most serious problems in a multiple regression setting is multicollinearity (or just collinearity) which occurs when the regressor variables are correlated with one another. Collinearity causes many problems in a multiple regression model. One of the main problems is that if collinearity is severe, the estimated regression model becomes very unstable and the resulting coefficient estimates become difficult to impossible to interpret. In the previous example with ATP measured from sludge, we saw that the cubic polynomial coefficients were very unstable even though the over all $F$ test was significant ($p = 0.04635$). Typically when the overall $F$ tests says that at least one of the coefficients differs from zero, but all of the partial $t$-tests say all the coefficients are indistinguishable from zero, then that is a sign of a collinearity problem. In order to illustrate the problem of collinearity, we give an example:

**Heart Catheter Example.** A study was conducted and data collected to fit a regression model to predict the length of a catheter needed to pass from a major artery at the femoral region and moved into the heart for children (Weisberg 1980). For 12 children, the proper catheter length was determined by checking with a fluoroscope that the catheter tip had reached the right position. The goal is to determine a model where a child's height and weight could be used to predict the proper catheter length. The data are given in the following table:

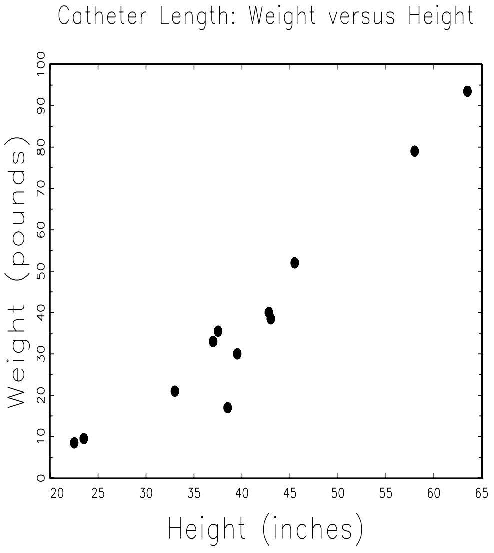

Figure 12: A scatterplot of weight versus height for $n = 12$ children in an experiment used to predict the required length of a catheter to the heart based on the child's height and weight.

| Height | Weight | Length |
|---|---|---|
| 42.8 | 40.0 | 37 |
| 63.5 | 93.5 | 50 |
| 37.5 | 35.5 | 34 |
| 39.5 | 30.0 | 36 |
| 45.5 | 52.0 | 43 |
| 38.5 | 17.0 | 28 |
| 43.0 | 38.5 | 37 |
| 22.5 | 8.5 | 20 |
| 37.0 | 33.0 | 34 |
| 23.5 | 9.5 | 30 |
| 33.0 | 21.0 | 38 |
| 58.0 | 79.0 | 47 |

After fitting a multiple regression model using height and weight as regressors, how do we interpret the resulting coefficients? The coefficient for height tells us how much longer the catheter needs to be for each additional inch of height of the child *provided the weight of the child stays constant.* But the taller the child, the heavier the child tends to be. Figure 12 shows a scatterplot of weight versus height for the $n = 12$ children from this experiment. The plot shows a very strong linear relationship between height and weight. The correlation between height and weight is $r = 0.9611$. This large correlation complicates the interpretation of the regression coefficients.

Figure 13 shows a 3-dimensional plot of the height and weight as well as the response variable $y$ = length. The goal of the least-squares fitting procedure is to determine the best-fitting plane through these points. However, the points lie roughly along a straight line in the height-weight plane. Consequently, fitting the regression plane is

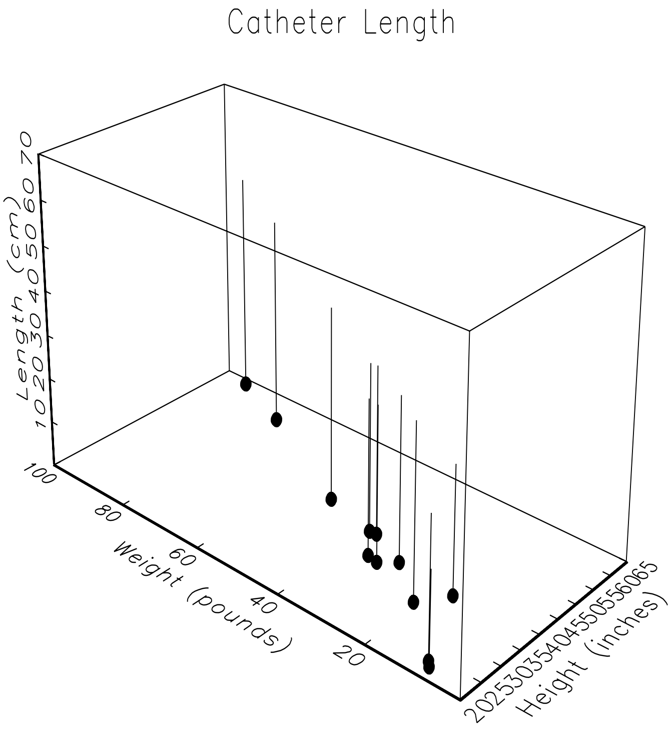

Figure 13: 3-D plot of the heart catheter data.

analogous to trying to build a table when all the legs of the table lie roughly in a straight line. The result is a very wobbly table. Ordinarily, tables are designed so that the legs are far apart and spread out over the surface of the table. When fitting a regression surface to highly correlated regressors, the resulting fit is very unstable. Slight changes in the values of the regressor variables can lead to dramatic differences in the estimated parameters. Consequently, the standard errors of the estimated regression coefficients tend to be inflated. In fact, it is quite common for none of the regression coefficients to differ significantly from zero when individual $t$-tests are computed for regression coefficients. Additionally, the regression coefficients can have the wrong sign – one may obtain a negative slope coefficient when instead a positive coefficient is expected.

Another major problem when collinearity is present is that the fitted model will not be able to produce reliable predictions for values of the regressors away from the range of the regressor values. If the data changes just slightly, then the predicted values outside the range of the data can change dramatically when collinearity is a present – just think of the wobbly table analogy.

In a well designed experiment, the values of the regressor variables will be orthogonal which means that the covariances between estimated coefficients will be zero. Figure 14 illustrates an orthogonal design akin to the usual placement of legs on a table.

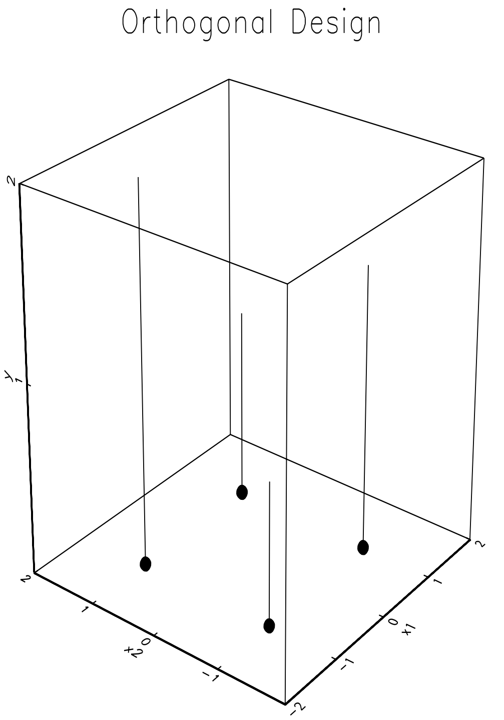

Figure 14: 3-D plot of orthogonal regressor variables.

## 6.1 Accessing Multicollinearity

One of the most common signs of a multicollinearity problem is that when you run a multiple regression the overall $F$-test statistic is highly significant indicating that at least one of the regression coefficients differs from zero **but** none of the $t$-tests for individual coefficients is significant. This apparent paradox is usually caused by multicollinearity. When multicollinearity is present, the variance of the coefficient estimators become inflated which makes the denominator of the $t$-test statistics big and thus makes the $t$-test statistic small and insignificant.

Perhaps the easiest way to access multicollinearity is to simply compute the correlation matrix of the regressors – this can be done with the "cor" function in R which will produce a matrix of pairwise correlations of all the predictors. However, correlations only measure pairwise linear associations and sometimes associations can be more complicated.

A popular diagnostic for multicollinearity is the *variance inflation factor*.

**Definition: Variance Inflation Factor (VIF)**: The VIF for the $j$th predictor is

$$\text{VIF}_j = \frac{1}{1 - R_j^2}, \tag{10}$$

where $R_j^2$ is the coefficient of determination when the $j$th predictor is regressed on all the other predictors. If the $j$th predictor can be approximated well by a linear combination of the other predictors in the model, then $R_j^2$ will tend to be large making $VIF_j$ large. In particular, one can show that the variance of an estimated regression coefficient $\hat{\beta}_j$ is given by

$$\mathrm{var}(\hat{\beta}_j) = \frac{\mathrm{MSE}\times\ \mathrm{VIF}}{(n-1)s_j^2}, \tag{11}$$

where $s_j^2$ is the sample variance of $x_j$, the $j$th regressor. If a regressor is highly correlated to the other regressors in the model, the result is an inflated variance of the estimated regression coefficient.

In such cases, one may question whether the $j$th predictor is needed in the model. However, there could be several predictors with large VIF's. It's difficult to say how large a VIF has to be in order to say the $j$th variable is causing a multicollinearity problem. One popular rule of thumb is become concerned whenever the $VIF > 10$.

There are other methods of accessing multicollinearity that involve examining the eigenvalues of the covariance matrix of the predictors. The eigenvalues cannot be negative. If any are near zero, then multicollinearity will tend to be a problem.

## 6.2 Solutions to the Multicollinearity Problem

There are many ways of dealing with the multicollinearity problem which we highlight below.

- The easiest solution is to simply drop regressors from the model. In the heart catheter example, height and weight are highly correlated. Therefore, if we fit a model using only height as a regressor, then we do not gain much additional information by adding weight to the model. In fact, if we fit a simple linear regression using only height as a regressor, the coefficient of height is highly significant ($p$-value $< 0.0001$) and the $R^2 = 0.7971$ is just slightly less than if we had used both height and weight as regressors. (Fitting a regression using only weight as a regressor yields an $R^2 = 0.8181$.) When there are several regressors, the problem becomes one of trying to decide upon which subset of regressors works best and which regressors to throw out. Often times, there may be several different subsets of regressors that work reasonably well. The problem of determining an appropriate set of regressor variables to use in the final model involves both statistical considerations (i.e. minimizing the collinearity problem, stable coefficient estimates, etc.) as well as input from the scientists who are knowledgable about which regressors are most important to the problem at hand. Deciding which variables to keep and which to throw out is a model building problem. There are many algorithms for model building in regression. Perhaps the most promising is the *lasso* that has been recently developed, see Friedman *et al.* (2010, 2009); Tibshirani (1996).

- Collect more data with the goal of spreading around the values of the regressors so they do not form the "picket fence" type pattern as seen in Figure 13. Collecting more data may not solve the problem in situations where the experimenter has no control over the relationships between the regressors, as in the heart catheter example.

- Biased regression techniques such as *ridge regression* and *principal component regression.* Details can be found in advanced textbooks on regression. These methods are called biased because the resulting slope estimators are biased. The tradeoff is that the resulting estimators will be more stable.

In polynomial regression, collinearity is a very common problem when fitting higher order polynomials and the resulting least-squares estimators tend to be very unstable. Centering the regressor variable $x$ at its mean (i.e. using $x_i - \bar{x}$ instead of $x_i$) helps alleviate this problem to some extent. Another solution is to fit a polynomial regression using *orthogonal polynomials* (details of this method can also be found in many advanced texts on regression analysis).

# 7 Analysis of Covariance (ANCOVA)

In many statistical applications, the goal is to compare two or more groups in terms of a continuous response variable $y$. The statistical tests for making such comparisons are the two-sample $t$-test or an analysis of variance (ANOVA) for comparing more than two population means. In many such cases, there may be additional information available that can help in the comparison of the populations. This additional information may exist in the form of a continuous variable $x$ that is correlated with the response variable $y$. Thus, we could model the response $y$ using a regression relation with $x$, but we would also like to build into the model a way of differentiating between the different populations. Sometimes this continuous regressor is called a *covariate.*

In order to compare different levels of a factor of interest, there could be confounding due to a covariate. For instance, if we are looking at plant growth in two different locations, there could be differences due to the some other factors due to soil conditions, like the amount of phosphorus in the soil. Phosphorus may very well explain a lot of the variability seen in the sizes of plants. Thus, it would be ideal to control for this variability so that a more focused examination on the differences in location could be made.

To make matters clearer, we illustrate with an example:

**Example.** (*open fruitfly.r*). It has been established that increased reproduction reduces longevity in female fruitflies. A study was conducted to see if the same effect exists for male fruitflies (Hanley and Shapiro, 1994). The experiment consisted of five groups: males forced to (i) live alone, (ii) to live with one pregnant female, (iii) to live with eight pregnant females, (iv) to live with one fertile female, and (v) to live with eight fertile females. The response of interest is longevity (measured in days). Variables also measured were thorax length (mm), and the percentage of each day spent sleeping. For our analysis, we will only focus on two groups: control group of males living with one pregnant female and an experiment group of males living with one fertile female. The data is in the file *fruitfly.dat.*

An initial attempt at analyzing the data is to just perform a simple two-sample $t$-test to determine if there is a significant difference in lifespan for males living with a single female who is pregnant and who is fertile. Here is some R-code that does some data formatting:

```
fruitfly <- read.table("fruitfly.dat")
control=fruitfly[fruitfly[,2]==1 & fruitfly[,3]==0,]
n0=dim(control)[1]
trt1=fruitfly[fruitfly[,2]==1 & fruitfly[,3]==1,]
n1=dim(trt1)[1]
fruit=rbind(control,trt1)
d=rbind(matrix(0,n0,1), matrix(1,n1,1)) # d=dummy variable for group
fruit=cbind(fruit,d)
y=fruit[,4]
thorax=fruit[,5]

# First, perform a simple t-test to see if there is a difference
# in mean lifespan for the control and experimental groups.
fit0=lm(y~d)
summary(fit0)
```

In this code, we have defined a binary dummy variable $d$ to act as an indicator variable for the two groups of male fruitflies. The $t$-test can be performed in R using the linear model function "lm" by simply regressing $y$ on the dummy variable (this is an illustration that ANOVA is really just regression with indicator variables). The output from running this model is

```
Call:
lm(formula = y ~ d)

Residuals:
   Min     1Q Median     3Q    Max 
-35.76  -8.79   0.20  10.46  32.20 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   64.800      3.059  21.184   <2e-16 ***
d             -8.040      4.326  -1.859   0.0692 .  
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1 

Residual standard error: 15.29 on 48 degrees of freedom
Multiple R-squared: 0.06713,    Adjusted R-squared: 0.0477 
F-statistic: 3.454 on 1 and 48 DF,  p-value: 0.06923 
```

More formally, the binary indicator variable $d$ is defined as:

$$d = \begin{cases} 0 & \text{if male lives with pregnant female} \\ 1 & \text{if male lives with fertile female.} \end{cases}$$

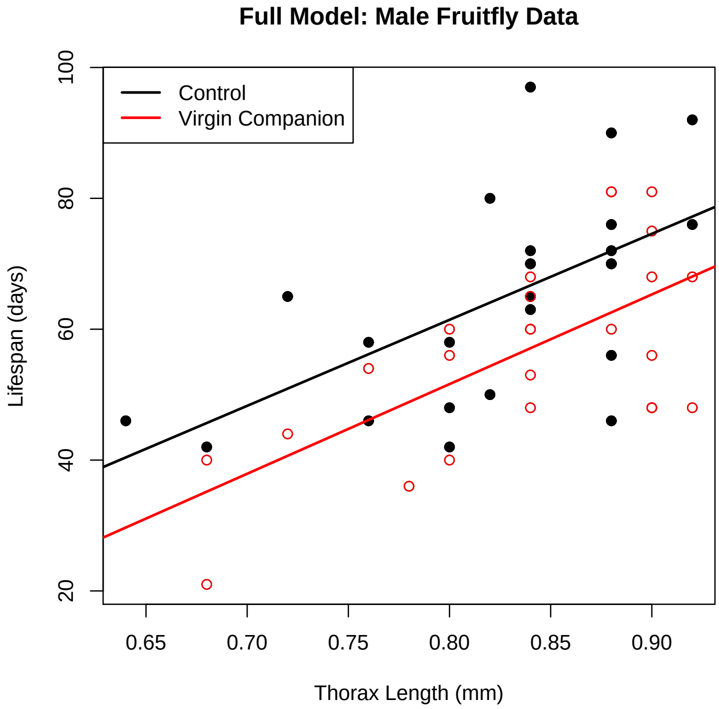

Figure 15: Scatterplot of lifespan versus thorax length for male fruitflies living with a pregnant female (solid points) and males living with a fertile female (open points). Also shown are the unconstrained fitted regression lines

Note that the $p$-value for the $t$-test is $p = 0.06923$ which does not meet the usual criterion for statistical significance, namely $\alpha = 0.05$ and hence one may be tempted to say there is no significant difference between the lifespan of males in the two experimental conditions. However, a lot of the variability in this data set is due to differences in thorax lengths of the fruitflies. If we regress lifespan on thorax, we get an $R^2 = 0.3255$ indicating the thorax length explains about a third of all variation in the lifespans. If we introduce thorax length as a covariate, we can factor out the variation due to this variable.

Let $x$ denote a covariate variable (such as thorax length) and let $d$ be a binary indicator variable differentiating two groups (note: this can be generalized to more than two groups). The **full** model under consideration here is

$$y = \beta_0 + \beta_1 x + \beta_2 d + \beta_3(d \times x) + \epsilon = \begin{cases} \beta_0 + \beta_1 x + \epsilon, & \text{if } d = 0 \\ (\beta_0 + \beta_2) + (\beta_1 + \beta_3)x + \epsilon, & \text{if } d = 1 \end{cases}. \quad (12)$$

The term $d \times x$ is called the *interaction* term. Note that the full model produces two distinct regression lines with slopes $\beta_1$ and $\beta_1 + \beta_3$ respectively for the two groups. Also, both lines have distinct $y$-intercepts, namely $\beta_0$ and $\beta_0 + \beta_2$ respectively. This model was fit to the data and the plot is shown in Figure 15. Note that the two lines are not parallel, but they are close to being parallel. Note that it is *the interaction term that causes the slopes of the two models to differ.* The output from fitting the full model is

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -43.725     29.766  -1.469 0.148655
thorax       131.450     35.931   3.658 0.000651 ***
d            -14.267     42.200  -0.338 0.736836
thorax:d       5.551     50.575   0.110 0.913072

Residual standard error: 12.3 on 46 degrees of freedom
Multiple R-squared: 0.4217,     Adjusted R-squared: 0.384
F-statistic: 11.18 on 3 and 46 DF,  p-value: 1.244e-05
```

Next, we can perform a test for whether or not a single regression line will suffice for the two groups of males and consequently there is no difference in lifespan for the two groups of male fruitflies. We can do this by dropping all the terms in the reduced model corresponding to thorax and fit the *reduced model*

$$y = \beta_0 + \beta_1 x + \epsilon, \tag{13}$$

and then perform an $F$-test that will simultaneously test if $H_0 : \beta_2 = \beta_3 = 0$ in the full model (12). We can do this in R using:

```
fitfull=lm(y ~ thorax + d + thorax*d)
fitreduced=lm(y~ thorax)
anova(fitreduced, fitfull)
```

The output from the anova function is

```
Analysis of Variance Table

Model 1: y ~ thorax
Model 2: y ~ thorax + d + thorax * d
  Res.Df    RSS Df Sum of Sq      F  Pr(>F)
1     48 8118.4
2     46 6961.1  2    1157.3 3.8239 0.02909 *
```

The $p$-value of $p = 0.029$ indicates that we cannot reject $H_0$ and that at least one of the coefficients $\beta_2$ and $\beta_3$ in the full model are non-zero.

Typically at this point, a natural next step is to test if the regression lines for the two groups are parallel. That is, we would like to test $H_0 : \beta_3 = 0$ which would drop the interaction term out of the model. In R, this hypothesis can be test using:

```
fitparallel=lm(y ~ thorax + d)
anova(fitparallel, fitfull)
```

which give the following output:

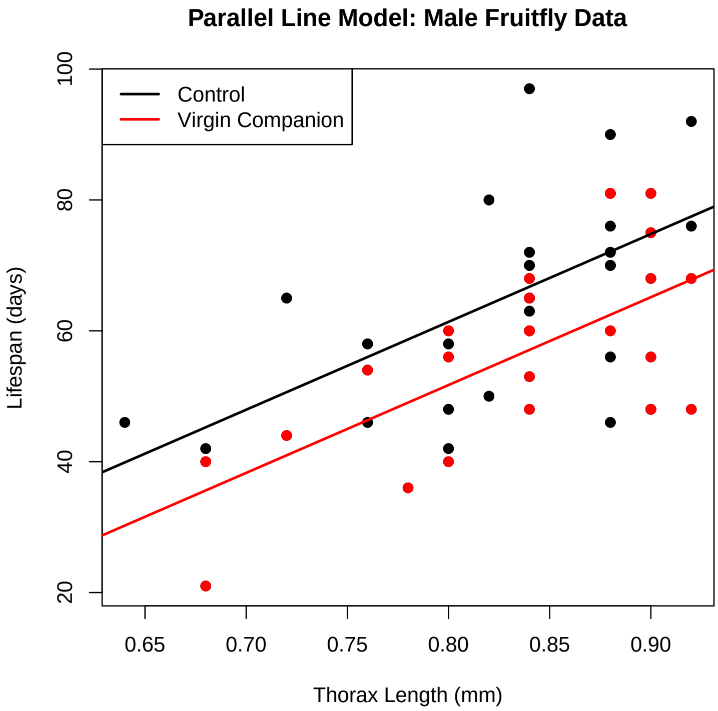

Figure 16: Scatterplot of lifespan versus thorax length for male fruitflies living with a pregnant female (solid points) and males living with a fertile female (open points). The least-squares regression lines are from the parallel line model.

```
Model 1: y ~ thorax + d
Model 2: y ~ thorax + d + thorax * d
  Res.Df    RSS Df Sum of Sq     F Pr(>F)
1     47 6962.9
2     46 6961.1  1    1.8233 0.012  0.913
```

Here the $p$-value is very large ($p = 0.913$) indicating that the parallel line model is consistent with the observed data. Note that this $p$-value is exactly the $p$-value from the full model $t$-test of $H_0 : \beta_3 = 0$; in fact, these two tests are exactly the same. Ordinarily statisticians are adverse to saying they would accept a null hypothesis. However, we are trying to build a useful and interpretable model. Based on the output here, it makes sense to drop $\beta_3$ from the full model and accept the parallel regression model. This does not mean we necessarily believe that in the full (potentially infinite) population of male fruit flies in this experimental condition that the actual regression lines are parallel. However, the experimental data shows fairly strong statistical evidence that they are very close to being parallel. Since models are approximations to reality, this approximation serves as a useful summary for comparing the two groups. The plot of the raw data and the estimated parallel regression line model is shown in Figure 16. Typing

```
summary(fitparallel)
```

gives the output for this model:

```
Call:
lm(formula = y ~ thorax + d)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -46.038     20.799  -2.214  0.03175 *
thorax       134.252     25.019   5.366 2.42e-06 ***
d             -9.651      3.456  -2.793  0.00753 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 12.17 on 47 degrees of freedom
Multiple R-squared: 0.4215,     Adjusted R-squared: 0.3969
F-statistic: 17.12 on 2 and 47 DF,  p-value: 2.593e-06
```

The first thing to note is that all of the estimated coefficients look very stable, as indicated by the small $p$-values from the partial $t$-tests for whether or not the coefficients are zero. Also, the parallel regression line model has an $R^2 = 0.4215$ which is only a very tiny drop from the $R^2$ from the full model. Finally, the estimated model we arrive at is

$$\hat{y} = -46.038 + 134.252x - 9.651d = \begin{cases} -46.038 + 134.252x, & \text{for males living with pregnant females} \\ -57.992 + 134.252x, & \text{for males living with fertile female.} \end{cases} \tag{14}$$

**Model Interpretation.** Recall that in a simple linear regression, the slope represents the rate of change in the mean response for changes in the predictor. Because both groups of males in the fruitfly example share a common slope (according to our statistical analysis), the rate of change in lifespan for a unit change in thorax length is estimated to be the same for both groups. The estimated slope is $\hat{\beta}_1 = 134.252$. Thus, we estimate that for each additional millimeter in thorax length, the male fruitfly's lifespan increases on average by about 134 days regardless of the type of companion (pregnant or fertile female). This interpretation is not very helpful because the thorax lengths in the data set range from 0.64 to 0.92 mm. A more useful interpretation is that for each addition 0.1 mm length in thorax, the average lifespan for males in either experimental condition increases by 13.4 days on average. Often, in order to give an indication of the precision of our estimates, a confidence interval is useful. R has a built-in function for computing confidence interval – if we type

```
confint(fitparallel)
```

we get 95% confidence intervals for each of the coefficients in the model (95% is the default, but other confidence levels can be specified). The output from this command is

```
                 2.5 %     97.5 %
(Intercept) -87.87987  -4.196411
thorax       83.91956 184.583688
d           -16.60300  -2.699041
```

Thus, the 95% confidence interval for the common slope estimate reported here was computed by

$$\hat{\beta}_1 \pm t_{\alpha/2}\widehat{se}(\hat{\beta}_1).$$

From the output for the parallel model, we see that the estimated standard error for the slope is $\widehat{se}(\hat{\beta}_1) = 25.019$ and the percentile for the $t$ distribution for the confidence interval can be found by typing "qt(.975, 47)" which gives a value of 2.011741. Thus, the confidence interval is

$$134.252 \pm 2.011741(25.019),$$

which gives the confidence interval produced by "confint". With 95% confidence, we estimate that the lifespan of male fruitflies increases on average by about 8.4 to 18.5 days for each additional 0.1 mm in thorax length regardless of they type of female companion.

The other important coefficient in the model is the coefficient of $d$, namely $\beta_3$ which is estimated to be between $-16.603$ to $-2.699$ with 95% confidence. Because the regression lines are parallel, the distance between the lines remains constant over all thorax lengths. The interpretation here is that we estimate with 95% confidence that males with a fertile female companion have a lifespan that is between 2.7 to 16.6 days shorter on average than males with a pregnant female companion *regardless of thorax length*. It is the coefficient $\beta_3$ that tells us the cost in terms of diminished lifespan due to reproduction for males.

A couple of important points should be made here:

- Because the interaction term dropped out, the interpretation of the model is much easier. In particular, the mean difference in lifespans for the two groups remains constant regardless of thorax length. If there was a significant and important interaction, then the difference between the lines would vary with thorax lengths.
- Recall that the initial test performed was a $t$-test comparing the lifespans of the two groups without accounting for thorax length – the result of that $t$-test was that there was no significant difference between the two groups in terms of mean life span. However, once we factored out the variability due to differences in thorax lengths, we were able to see that there is indeed a significant difference in mean life spans depending on the type of female companion.

The confidence interval for the intercept $\beta_0$ is not of interest here because the intercept does not have a clear interpretation – this is because we have not collected data for thorax values near the intercept.

**Bonferroni Correction.** Since we are interested in estimating two of the coefficients using confidence intervals, we should correct for multiplicity. If we form two confidence intervals at level 0.95 and interpret them simultaneously, then the actual confidence level will be less than 0.95. To illustrate this idea, suppose there is a 95% chance I win if I play a particular game. If I play the game twice, the probability I win both games is no longer 95% but less than 95%. The same logic holds if we want to make a confidence statement for a set of parameters at a certain level of confidence. A simple correction for this multiplicity issue is to use the *Bonferroni* correction. Let $g$ denote the number of estimates we wish to make (in our case, $g = 2$). Then in the confidence interval formula above, the $t$-critical value we use would be adjusted to $t_{\alpha/(2g)}$. For 95% confidence with $\alpha = 0.05$, we would get $\alpha/(2g) = .05/(4) = 0.0125$. To get the Bonferroni $t$-critical value in R we could then type in "qt(1-.0125, 47)" which gives 2.3155. Therefore, the Bonferroni correction will lead to a slightly wider set of confidence intervals. The Bonferroni procedure is a bit conservative meaning that it produces confidence intervals that are wider than they need to be. Alternatively to confidence intervals, one could form a confidence region for the pair $(\beta_1, \beta_2)$ using multivariate normal theory which produces an elliptically shaped region in the plane. This procedure yields more precise confidence regions usually (i.e. smaller area than the Cartesian product of two Bonferroni intervals) but they are more difficult to interpret.

In the modeling above, we used a single regression model for both groups of male fruitflies. One could also simply fit regression lines individually for each group. However, doing the fitting using a single model makes it easier to test hypotheses about common slopes etc. Also, fitting a single model allows us to pool all the data to estimate the common slope parameter. On the other hand, the single model we used here assumes that the error variance for both groups is the same. Fitting a single model allows us to pool the data to estimate this common error variance as well. However, if the error variances between the two groups differ dramatically, then separate estimates of the error variances should be made.

**More than two populations.** In the example above, suppose we wanted to compare all the experimental groups instead of just two? If there are $k$ groups to compare in an ANCOVA, then we would need to define $k-1$ indicator variables in order to allow for a separate regression line for each treatment group. The full model would also include interactions of the predictors with the indicator variables.

# 8 Mixed Effect Models in Regression

As we saw in the introductory chapter, models can have fixed and random effects. This is true of regression models as well. Recall that in the classical regression model, the errors are assumed to be independent of one another. However, in many settings, this assumption is violated and using random effects is a way to account for these correlations. For example, if we access the growth of plants over time by taking repeated height measurements on the plants, then the height measurements obtained from a single plant tend to be correlated with one another. If we perform an experiment with plants in various plots, the plants are clustered within plots and plants within the same plot will tend to be correlated with each other. For repeated measurements on a single plant (or organism, etc), we can introduce a random effect for the plant to account for the correlated measures on the same plant. In the case of plants clustered together in plots, a random effect for plot can be introduced in the model to account for correlations of plant measurements taken from the same plot. The next example illustrates the second case.

**Example.** *open sag.r.* The ecological balance of severely nutrient-limited areas such as the Florida Everglades can be jeopardized due to anthropogenic nutrient enrichment. In order to detect and monitor the presence of nutrient contamination in the Everglades, a study was conducted to examine variation in plant morphology that results from soil characteristics (data from this study was provided by C. Ivey). In particular, the negative effects of phosphorus enrichment in the Everglades are a concern. Determining the effect of soil characteristics on the size and/or the shape of plants can be valuable to Everglades managers. Motivation for this model comes from a study of the plant *Sagittaria lancifolia* which is common throughout the wetlands of the southeastern United States. Survey sites were randomly located throughout the Florida Everglades and a sample of plants and soil measurements were obtained at each site during the wet season. The data set *sagwet.dat* has the following columns: 1. Site, 2. Plant base, 3. petiole length (i.e. leaf stalk), 4. leaf length, 5. leaf width, 6. phosphorus in soil, and 7. ash-free dry weight (afdw) of the soil. For this analysis we will focus on the relation of leaf width as a function of phosphorus levels in the soil. Data on plants and soil were collected at 59 sites within the Everglades. The two predictor variables in this example are the soil characteristics (phosphorus and afdw) and all plants in the same plot share the same values of these two predictors.

The first problem we note it that the leaf widths are strongly skewed to the right and clearly non-normal as indicated in the left panel of Figure 17. The right panel shows a histogram of log(width) which is no longer skewed (but still not very normal looking). Figure 18 shows a scatterplot of log(width) versus total soil phosphorus

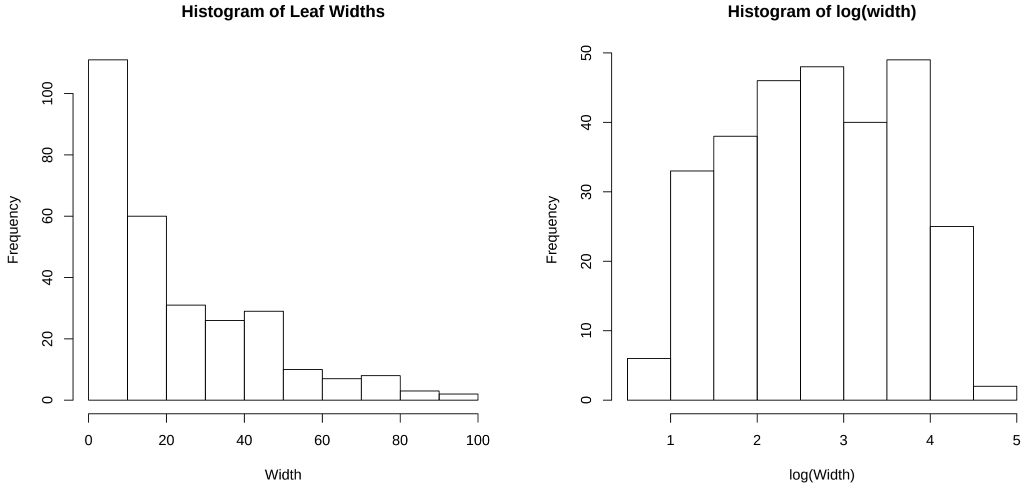

Figure 17: Histograms of leaf widths (left-panel) and log(leaf widths) in the right panel

that appears to show an increasing log(leaf width) with increasing phosphorus levels in the soil.

Because plants are clustered within plots, our regression model must account for this dependency. Let $y_{ij}$ denote the log(width) of the $j$th plant at the $i$th site. Because the sites are randomly selected throughout the Everglades, site should be regarded as a random effect. Plants within the same site will have the same soil characteristics, such as the level of phosphorus. We could simply fit a simple linear regression model

$$y_{ij} = \beta_0 + \beta_1 x_i + \epsilon_{ij} \tag{15}$$

to the data and ignore the information on site. According to this model, log(width) from plants within the same site will be correlated since they have the same $x_i$ value. However, in this model the site to site variability will be pooled into the error terms $\epsilon_{ij}$ as well as the variability between plants within a site. Therefore, according to (15), $\epsilon_{ij}$ and $\epsilon_{ij'}$ would not be independent. Treating them as independent as in (15) will make it appear we have more information in the data than we actually do.

Ideally we would like the error term to reflect the variability between plants only - no two plants under the same conditions will have identical leaf sizes. To account for the site to site variability we can consider the following model

$$y_{ij} = \beta_0 + \beta_1 x_i + b_i + \epsilon_{ij}, \tag{16}$$

where $\beta_0$ and $\beta_1$ are the fixed effect $y$-intercept and slope coefficients; $b_i$ is the random effect for the site assumed to have a normal distribution with mean zero and variance we will denote $\sigma_b^2$; $\epsilon_{ij}$ is the random error assumed to have a normal distribution with mean zero and variance $\sigma^2$ and also assumed to be independent of the random effects. Since we have factored out the site effect using $b_i$, the purpose of the random error $\epsilon_{ij}$ in (8) is to represents the plant to plant variability within a site and can be regarded as independent of $b_i$. Because the model has both fixed and random effects,

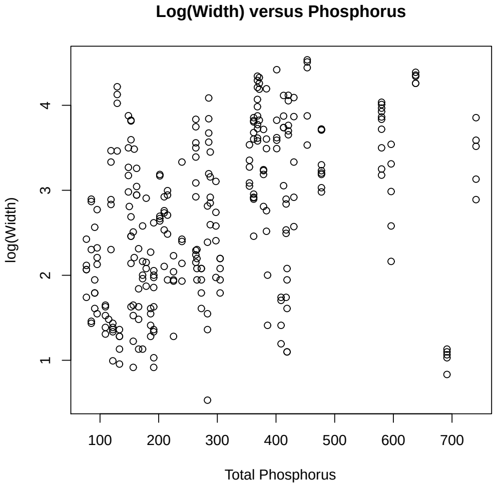

Figure 18: Scatterplot of log(leaf width) versus total soil phosphorus

it is an example of a *mixed effects model.* The responses $y_{ij}$ for plants in the same site will be correlated not only because they share the same $x_i$ value but because they also share common soil characteristics. The random effect for site $b_i$ introduces an additional correlation between responses on plants at the same site not accounted for by the shared values of $x_i$. To fit this model in R we can use the "lmer" function from the lme4 library (that must be loaded onto your computer if it has not already been loaded). Here is the R code:

```
library(lme4)
sagwet=read.table("sagwet.dat", header=T)
names(sagwet)
phos=sagwet$phos
width=sagwet$width
hist(width, main="Histogram of Leaf Widths", xlab="Width")
hist(log(width), main="Histogram of log(width)", xlab="log(Width)")
plot(phos, log(width), xlab="Total Phosphorus", ylab="log(Width)",
     main="Log(Width) versus Phosphorus")

fit=lmer(log(width) ~ phos + (1|site), data=sagwet)
summary(fit)
VarCorr(fit)$site # Covariance matrix for random effects
betahat = fixef(fit)  # Extract the estimated fixed effects
s2 =  attr(VarCorr(fit), "sc") # error variance
se=sqrt(diag(vcov(fit)))  # extract standard errors of fixed effects
t=betahat/se    # manually compute t-test for fixed effects
```

The method of maximum likelihood is used to estimate the parameters of the model which was briefly introduced in the Introduction chapter. The results from the summary statement are as follows:

```
Linear mixed model fit by REML
Formula: log(width) ~ phos + (1 | site)
   Data: sagwet
   AIC BIC logLik deviance REMLdev
 525.4 540 -258.7      502   517.4
Random effects:
 Groups   Name        Variance Std.Dev.
 site     (Intercept) 0.64288  0.80180
 Residual             0.19366  0.44007
Number of obs: 287, groups: site, 57

Fixed effects:
             Estimate Std. Error t value
(Intercept) 1.9883048  0.2289402   8.685
phos        0.0022750  0.0006907   3.294

Correlation of Fixed Effects:
     (Intr)
phos -0.877
```

From this output, we see that the estimated fixed effect intercept and slope are $\hat{\beta}_0 = 1.9883$ and $\hat{\beta}_1 = 0.002275$ respectively. The output says that the model was fit by REML which stands for "restricted maximum likelihood" which is a modification of the usual likelihood estimation that is supposed to produce estimators that are less biased than the usual maximum likelihood estimators. Notice that the output gives the estimates of the random effect variance $\hat{\sigma}_b^2 = 0.64288$ and the error variance $\hat{\sigma}^2 = 0.19366$. The $t$ test statistic for significance of the phosphorus fixed effect slope is

$$t = \frac{\hat{\beta}_1}{\hat{se}(\hat{\beta}_1)} = \frac{0.002275}{0.0006907} = 3.294.$$

The degrees of freedom for the $t$-test can be made conditional on the estimated random effects: there are $n = 287$ observations and we are estimating 2 fixed effects, leaving $287 - 2 = 285$ degrees of freedom for the $t$-test. A $t$-test statistic value of $t = 3.294$ is highly significant indicating that the log(width) of the leaves is indeed influenced by the phosphorus levels in the soil. Figure 19 shows a plot of the raw data again along with the estimated regression line whose $y$-intercept and slope correspond to the estimated fixed effects in the model.

## 8.1 Best Linear Unbiased Predictors (BLUPS)

In the *Sagittaria lancifolia* example above, we treated site as a random effect to recognize that the sites were randomly selected throughout the Everglades. If we had treated site as a fixed effect, then we could have treated it as a factor with fixed levels in the model and we could have estimated the mean effect for each site. However, since we are regarding site as a random factor, it does not make sense to estimate a random quantify. Instead, we can predict the random effect. The way this is done is

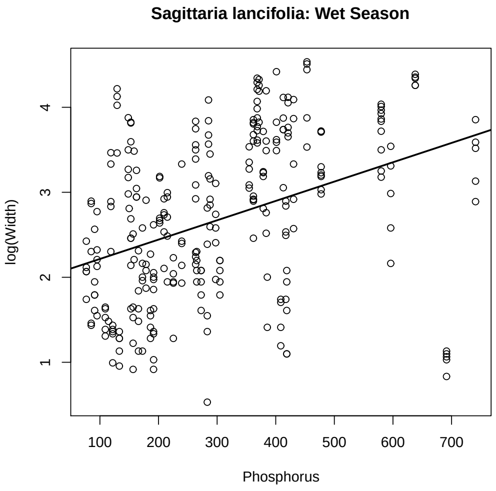

Figure 19: Scatterplot of log(leaf width) versus total soil phosphorus with the fitted regression line from the linear mixed effects model. The plotted line corresponds to the fixed effect estimates of the intercept and slope.

to compute an estimate of the *conditional expectation* of the site effect $b_i$ given the observed log(widths) of the plants at that site. Assuming the joint distribution of the random effects and the responses is well approximated by a multivariate normal distribution, this conditional expectation is a linear function of the responses which is called the *best linear unbiased predictor* or BLUP of the random effect.

In the R program above, we had a random effect for site and we can extract the BLUPs for these random effects in R using the command

```
ranef(fit)$site
```

where "fit" is the name we gave to the object created by fitting the linear mixed effects model using "lmer". If we add the fixed effects to the random effects, we can get predicted responses for each site, as illustrated in the following R code:

```
# Compute the best linear unbiased predictions (BLUPS) + fixed effects
# First, extract each site and phosphorus and afdw values and put into 'dat'
dat=NULL
site = unique(sagwet$site)
for (isite in site){
  dati=sagwet[sagwet$site==isite,]
  dat=rbind(dat, dati[1,c(1,6,7)]) # extract sites & phos
}
# Add a 4rd column to dat = fixed effect + blup
dat=cbind(dat, fixef(fit)[1]+fixef(fit)[2]*dat[,2]+ranef(fit)$site)
```

The data set contains another soil variable, the ash-free dry weight (afdw) of the soil. There appears to be a significant interaction between phosphorsus and ash-free dry weight of the soil. We can fit a model with an interaction between these terms and perform a likelihood ratio test comparing the model with only phosphorus to the model with ash-free dry weight and its interaction with phosphorus. Here is the R code:

```
# Add ash-free dry weight to the model
fit2=lmer(log(width) ~ phos + afdw + I(afdw*phos) + (1|site), data=sagwet, REML=F)
summary(fit2)
# Re-fit original model using maximum likelihood, not REML
fit=lmer(log(width) ~ phos + (1|site), data=sagwet, REML=F)
# Compute likelihood ratio test p-value
as.numeric(1-pchisq(2*(logLik(fit2)-logLik(fit)),2))
```

The last line here produces the $p$-value which is $p = 0.036$ indicating that the addition of afdw and its interaction with phosphorus to the model is helpful (note that afdw added by itself is not significant, but when the interaction is added, the addition is significant). The likelihood ratio test is explained in more detail in the next chapter. Basically, the likelihood ratio test is similar to the $F$-test used earlier to compare a full and reduced model. In the setting of linear mixed effects models, the test statistic follows an approximate chi-square distribution if the null hypothesis is true and the sample size is relatively large. The degrees of freedom for the chi-square test is equal to the difference in the number of fixed effect parameters between the full and reduced models. For this test, the chi-square degrees of freedom is equal to two because the models differ by the terms "afdw" and its interaction with phosphorus. Note that in "lmer" one must specify the "REML=F" so that a standard likelihood estimation approach is used. The restricted maximum likelihood approach (which is the default for "lmer") first transforms the data to make the model free of the fixed effects. Therefore, to test a hypothesis regarding fixed effects, we should use the usual maximum likelihood estimation and not REML.

It is interesting to note that the estimated variance for the site random effect in the model with only phosphorus as a predictor was $\hat{\sigma}_b^2 = 0.64288$ but this variability is reduced to $\hat{\sigma}_b^2 = 0.54461$ after afdw and the interaction are added to the model. These can be computed with the R code

```
> as.numeric(VarCorr(fit)$site)
[1] 0.6428815
> as.numeric(VarCorr(fit2)$site)
[1] 0.5446087
```

Additionally, afdw has an estimated correlation of about 0.18 with the BLUP for site in the original model with only phosphorsus as a predictor. On the other hand, the estimated error variance for the two models remains virtually unchanged:

```
> attr(VarCorr(fit), "sc")
[1] 0.4400699
> attr(VarCorr(fit2), "sc")
[1] 0.4401749
```

**A Technical Aside.** Often in regression modeling problems, particularly those using experimental data, the predictor(s) $x$ is regarded as fixed (say by design). Also, the random effects ($b_i$ in our model) are assumed to be independent of the predictors. However, in the example presented here, it does not make sense to consider the phosphorus values as fixed since they are measured at randomly chosen sites throughout the Everglades. Additionally, the random effect for plot will generally not be independent of $x_i$ because this random effect corresponds to variability due to other soil characteristics. In particular, the ash-free dry weight is correlated with phosphorus with an estimated correlation of 0.42. Since afdw was not used in the original model, its effects, as well as other soil characteristics, were subsumed into the site random effect. In the model (8), we would like to interpret the slope $\beta_1$ of $x_i$ to represent the effect of phosphorus on the log(leaf-widths), independent of the other (measured or unmeasured) soil variables, but unfortunately, this is not the case. Since the random effect will be correlated with $x_i$, we can write $E[b_i|x_i] = \gamma x_i$, assuming a linear conditional expectation. We can define a modified random effect

$$b_i^* = b_i - E[b_i|x_i],$$

as the residual after regressing $b_i$ on $x_i$. This residual $b_i^*$ will be uncorrelated with $x_i$ and will have mean zero and hence satisfies the usual requirements for a random effect. We can now consider the original model (8) conditional on the observed $x_i$ values as

$$y_{ij}|x_i = \beta_0 + (\beta_1 + \gamma_1)x_i + b_i^* + \epsilon_{ij}.$$

When the linear mixed effects model is fit to the data, the estimated slope of $x_i$ is not an estimator of the direct effect of phosphorus on the plants, (i.e. $\beta_1$) but instead is an estimate of $\beta_1 + \gamma_1$. That is, the effect of $x_i$ on the response is the direct effect $\beta_1 x_i$ plus the effect of soil characteristics that are predicted by $x_i$, namely $E[b_i|x_i] = \gamma x_i$.

# 9 Nonlinear Regression

All of the regression models we have considered so far have been linear models. By "linear" we mean that the model is linear in the parameters (i.e. the $\beta$'s). Even the polynomial models are considered linear models even though the response is a nonlinear function of the regressor $x$. Many regression applications involve nonlinear models – that is, models that are not linear in the parameters. A common example is an exponential growth model. Let $\mu_y$ denote the mean size of a population that grows (or decays) with time $t$. Then an exponential model for growth (or decay) is

$$\mu_y = \alpha_0 e^{-\alpha_1 t}. \tag{17}$$

This is a nonlinear model because the functional relationship is not a linear function of the parameters $\alpha_0$ and $\alpha_1$. The model (17) can easily be linearized by taking the natural logarithm on both sides giving:

$$\log(\mu_y) = \beta_0 + \beta_1 t,$$

where $\beta_0 = \log(\alpha_0)$ and $\beta_1 = -\alpha_1$. The logarithm transformation often works well in such cases, but it assumes the error in (17) is multiplicative instead of additive. If the error in (17) is additive, then the logarithm transformation may not yield a very good fit.

Different applications may yield other types of nonlinear models. For example the logistic growth model:

$$y = \frac{\theta_1}{1 + \theta_2 e^{-\theta_3 x}} + \epsilon,$$

is very useful for modeling the growth of a plant or animal. Here, $\theta_1, \theta_2, \theta_3$ are the parameters and they are related to $y$ nonlinearly; hence, it is a nonlinear model. The logistic growth model forms an "S"-shaped curve. In growth applications, typically the size of an organism levels off as they grow older which produces data following this logistic pattern. Note that we cannot linearize this logistic growth model as we did with the exponential growth model.

For linear models, there exist closed form solutions for the least-squares estimators of the parameters. However, for most nonlinear models, there do not exist closed form solutions. Instead, for nonlinear models, the least-square estimators must be determined by iterative algorithms.

Consider the following general setup. Suppose the response variable $y$ is related to the regressor variable $x$ via a nonlinear function $f$ that depends on unknown parameters $\theta_1, \ldots, \theta_k$ (in the logistic growth model, $k = 3$). Then we can write the model as

$$y = f(x; \theta_1, \ldots, \theta_k) + \epsilon.$$

As with any regression model setup, we obtain data $(x_1, y_1), \ldots, (x_n, y_n)$, and use this data to estimate the parameters $\theta_1, \ldots, \theta_k$. Least-squares is used most frequently for parameter estimation: find the values of the $\theta_j$'s that minimize the sum of squares:

$$\sum_{i=1}^{n} (y_i - f(x_i; \theta_1, \ldots, \theta_k))^2.$$

The least squares solutions will be denoted by $\hat{\theta}_j$ for $j = 1, \ldots, k$. It turns out that if the error $\epsilon$ is approximately normal, then the least squares estimators will have approximately normal distributions and they will be approximately unbiased. The approximation becomes better as the sample size gets larger. Most statistical software programs for nonlinear least-squares will provide the parameter estimates as well as estimated standard errors that are based on asymptotic results and the assumption that the error is normal (or at least approximately normal).

An alternative to the typical standard error estimates is to use a method called the *bootstrap* which is a computer intensive technique. For a good introduction to the bootstrap and its uses, see Efron and Tibshirani (1993). We shall illustrate the bootstrap approach below.

The most well-known method of finding the least-squares estimators is the *Gauss-Newton Algorithm.* Most of the common statistical software programs use the Gauss-Newton algorithm or variants of it. The idea behind the Gauss-Newton algorithm involves estimating the function $f$ by the linear portion of its Taylor series expansion. From this linear approximation, the usual least-squares fit can be obtained, which in turn can be used to update the parameter estimates. The Taylor series expansion involves partial differentiation of the function $f$ with respect to the parameters. In a linear model, these partial derivatives do not involve the actual parameters and a closed form expression for the least-square estimators is possible. However, in nonlinear models, the partial derivatives do involve the parameters typically which makes it necessary to estimate the parameters through an iterative process.

To begin the algorithm, initial values need to be proposed for the parameters. One must be careful when providing initial values because the final solution once the algorithm iterates can often depend quite heavily on these initial guesses. If the initial values provided are way off, then it is possible for the algorithm to wander off in the wrong direction. If this happens, then the algorithm may not converge or it may yield very poor solutions for parameter estimates. At each iteration of the algorithm, the parameter estimates are updated to a new value. After the algorithm has gone through several iterations, the change in the parameter estimates from iteration to iteration will be very small. Typically software programs have some cut-off value for the change in value of parameter estimates at which point the algorithm stops. With most software packages, the user can also specify the cut-off value for determining if the algorithm has converged.

It is quite common to run into problems when fitting a nonlinear regression model. The most common problem is that the algorithm will not converge or it produces nonsense solutions (e.g., parameter estimates that are not allowed). This problem is usually due to poor starting values. One possible solution is to try different starting values. Sometimes good starting values can be determined by looking at a plot of the data and using properties of the function such as noting where a horizontal asymptote should occur. Another solution is to perform a grid search over different parameter values in order to find good starting values. The SAS software has a built-in grid search in their nonlinear regression procedure.

In order to illustrate nonlinear regression models, we provide an example.

**Bioconcentration Factors (BCF) using Compartment Models.** Compartment models (e.g. see Bates and Watts, 1988, Chapter 5) are used to describe outcome measures from physical systems of compartments (e.g., in chemical kinetics, pharmacokinetics, and ecological and environmental systems) in which material flows to and from various compartments. The parameters of primary interest in compartment models are the rate parameters which specify the rate of transfer of material between compartments. Bioconcentration factors (BCF's) are indices that quantify how much chemical accumulates in an organism relative to the environmental exposures. The environment is considered one compartment and the organism is considered the other compartment. BCF's can be estimated using compartment models as the ratio of the organism's uptake rate to its discharge rate. If we let $k_u$ denote the uptake rate and $k_d$ denote the rate of depuration (the removal of impurities from the organism), then

$$\text{BCF} = \frac{k_u}{k_d}. \tag{18}$$

Let $C_e$ denote the environmental concentration and $C_o(t)$ denote the organism concentration at time $t$. Assuming the rates of chemical uptake and depuration for an organism remain constant, then we have the following equation:

$$\text{Change in concentration w.r.t time} = k_u C_e - k_d C_o(t).$$

This equation can be expressed as a differential equation:

$$dC_o(t)/dt = k_u C_e - k_d C_o(t).$$

The BCF is the ratio of the organism concentration to the environmental concentration at steady state (i.e. when $dC_o(t)/dt = 0$) which corresponds to the definition given in (18). The function $C_o(t)$ that satisfies the differential equation is

$$C_o(t) = \frac{k_u}{k_d} C_e [1 - e^{-k_d t}]. \tag{19}$$

Note that this is a nonlinear function of the rate parameters $k_u$ and $k_d$ and therefore, nonlinear regression techniques are needed to find estimators of the parameters. Here is an example to illustrate the fitting of the nonlinear compartmental model:

**Example**. (*open pyrene.r*). A study on BCF of the chemical pyrene in *Diporeia* from Lake Erie was undertaken (data provided courtesy of Peter Landrum at NOAA). The data in the following table gives the time (in hours) and the corresponding pyrene concentration (in DPMq/g):

| | |
|---|---|
| 2973.94 | 44.92 |
| 2368.59 | 46.82 |
| 3013.45 | 46.82 |
| 3483.96 | 46.90 |
| 2327.23 | 46.90 |
| 6170.50 | 92.75 |
| 6722.54 | 92.75 |
| 5969.43 | 99.75 |
| 5396.02 | 99.75 |
| 5657.01 | 99.83 |
| 5649.33 | 99.83 |
| 8892.11 | 168.08 |
| 9341.05 | 168.08 |
| 8143.53 | 173.35 |
| 7435.38 | 173.35 |
| 9672.66 | 173.40 |
| 8210.14 | 173.40 |
| 16295.88 | 310.90 |
| 17020.00 | 310.90 |
| 18362.03 | 339.80 |
| 16293.29 | 339.80 |
| 17064.71 | 339.88 |
| 17563.29 | 339.88 |
| 12292.31 | 479.00 |
| 17842.22 | 479.00 |

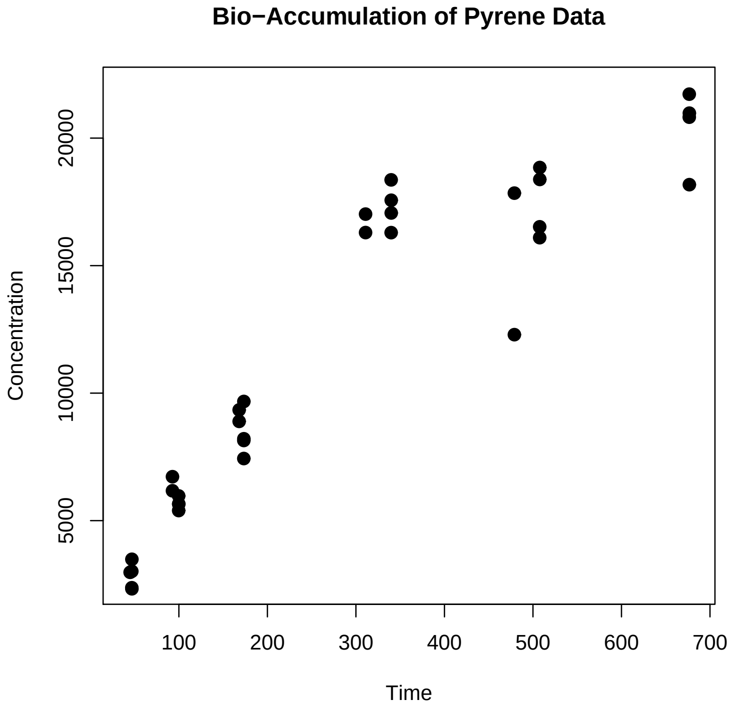

Figure 20: Pyrene accumulation in *Diporeia* versus time (in hours)

```
16095.87 507.60
16521.65 507.60
18379.32 507.72
18846.31 507.72
23298.64 672.48
25863.87 672.48
20820.86 676.57
21723.71 676.57
18174.71 676.68
20978.90 676.68
```

A plot of the data is shown in Figure 20.

Nonlinear regression models can be fit in R using the “nls” function which we illustrate here. Alternatively, in SAS, one can use “PROC NLIN” to fit nonlinear regression models. The R code to fit this nonlinear regression model for the *Diporeia* data is

```
pyrene <- read.table("pyrene.dat", header=F)
t=as.matrix(pyrene[,3])  # t=time
y=as.matrix(pyrene[,1])  # y=pyrene concentration
Ce=17970.3017 # Environmental concentration constant
biofit <- nls( y ~ ku/kd*Ce*(1 - exp(-kd*t)),  # nonlinear regression fit
                        start = list( ku = 0.0036, kd = 0.0018354 ),
                        trace = FALSE )
```

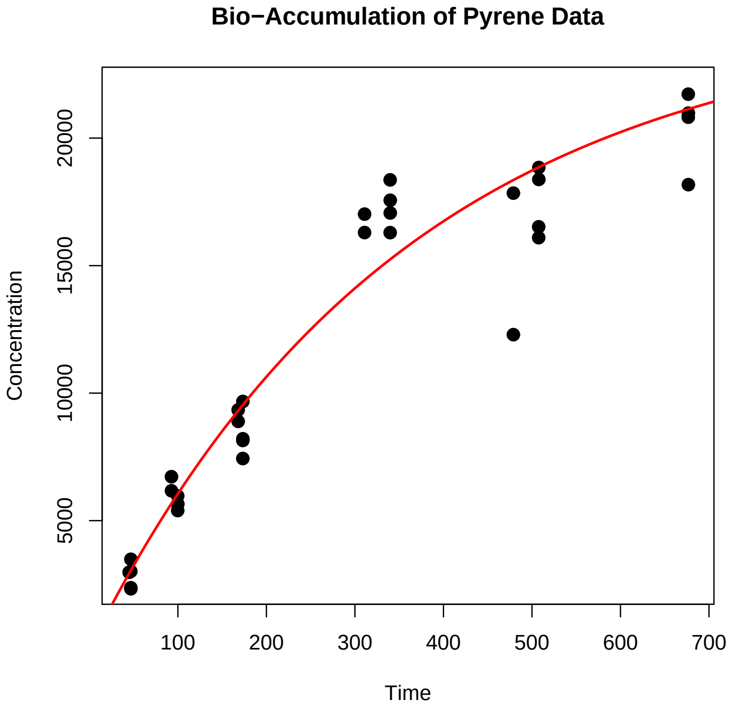

Figure 21: The fitted nonlinear least squares function along with the raw data of pyrene accumulation in *Diporeia* versus time (in hours)

In the “nls” function, the user needs to specify the nonlinear function to be estimated. Also, the user needs to specify initial values for the algorithm (under the “start =” in nls). The “ce=17970.3017” is an estimate of the environmental concentration of pyrene that was estimated separately from another data set from sediment. The starting values used in the code above were found by trial and error. Often one may need to try several different sets of starting values to find a set that works and does not make the algorithm crash.

The output from running this is as follows:

```
 Formula: y ~ ku/kd * Ce * (1 - exp(-kd * t))

Parameters:
    Estimate Std. Error t value Pr(>|t|)
ku 0.0038597  0.0003179   12.14 1.02e-13 ***
kd 0.0027820  0.0004248    6.55 1.93e-07 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 1973 on 33 degrees of freedom

Number of iterations to convergence: 4
Achieved convergence tolerance: 4.63e-08
```

From the R output, we see that the estimated rate parameters are $\hat{k}_u = 0.00386$ and $\hat{k}_d = 0.00278$. R also gives approximate standard errors for these estimated rate parameters. Figure 21 shows the scatterplot of the raw pyrene data along with the nonlinear least-squares fitted function estimate. As the figure shows, the curve provides a pretty good fit to the data. The estimated BCF is given by

$$\hat{\text{BCF}} = \frac{\hat{k}_u}{\hat{k}_d} = \frac{0.00386}{0.00278} = 1.3874.$$

It would be useful to obtain a standard error for this estimated ratio. Bailer and Venis (2000) suggest using the *delta method* (based on a Taylor series approximation to the ratio which defines the BCF) to obtain an estimated standard error. Using the delta method, the approximate standard error for the estimated BCF is (Bailer and Venis, 2000, equation 4):

$$\hat{\text{se}}(\hat{\text{BCF}}) = \sqrt{\frac{\hat{\sigma}_u^2}{\hat{k}_d^2} - \frac{2\hat{k}_u\hat{\sigma}_{ud}}{\hat{k}_d^3} + \frac{\hat{k}_u^2\hat{\sigma}_d^2}{\hat{k}_d^4}}, \tag{20}$$

where $\hat{\sigma}_u$ and $\hat{\sigma}_d$ are the estimated standard errors for the two rate parameters. The term $\hat{\sigma}_{ud}$ is the estimated *covariance* between the two estimated rate parameters. Estimates of these values can be extracted from the fitted model in R by typing "covb = vcov(biofit)" which produces

```
> covb
              ku           kd
ku 1.010814e-07 1.293201e-07
kd 1.293201e-07 1.804213e-07
```

This is the estimated covariance matrix of the estimated rate parameters. The standard errors of $\hat{k}_u$ and $\hat{k}_d$ are the square roots of the diagonal elements of this covariance matrix. In particular,

$$\hat{\sigma}_u = \sqrt{1.010814 \times 10^{-7}}.$$

We can obtain these in R by typing "sqrt(diag(covb))" which yields $\hat{\sigma}_u = 0.0003179330$ and $\hat{\sigma}_d = 0.0004247603$. Also,

$$\hat{\sigma}_{ud} = 1.293201 \times 10^{-7},$$

which is the off-diagonal element of the covariance matrix. These values can be plugged into (20) to obtain the approximate standard error for the estimated BCF which comes out to be $\hat{se}(\text{BCF}) = 0.1076$.

The delta method approach is an asymptotic approach which means that the estimated standard error is only approximate and the approximation gets better as the sample size grows larger. For small to medium sample sizes, it is difficult to gauge the reliability of estimates based on asymptotic results. Fortunately, with the wide availability of high speed computing, we have an alternative approach called the *bootstrap* (e.g. see Efron and Tibshirani, 1993).

## 9.1 Bootstrap Standard Errors

As an alternative to the delta method, one could use the bootstrap idea to obtain a standard error for BCF. In fact, the bootstrap can be used more generally to obtain standard errors and confidence intervals in most statistical model estimation problems.

**Basic Idea of the Bootstrap.** Suppose we collect data $x_1, \ldots, x_n$ from a population with probability distribution function $F(x; \boldsymbol{\theta})$ where $\boldsymbol{\theta}$ represents the collection of parameters for the distribution function. For instance, in the case of a normal distribution, $\boldsymbol{\theta} = (\mu, \sigma^2)$, the mean and variance. In statistics, we naturally form statistics $T(x_1, \ldots, x_n)$, which basically means, throw the data into a formula with the typical goal of estimating a parameter. For example, take

$$T(x_1, \ldots, x_n) = \frac{1}{n}(x_1 + \cdots + x_n) = \bar{x},$$

to be the sample mean and use it to estimate the population mean $\mu$. Ideally, if we wanted to access the how this estimator performs, we would take many many samples from the population (i.e. completely re-do the experiment say 1000 times), compute the statistic for each sample, and then look at the distribution of these statistics. However, this of course is impractical. The idea of the bootstrap is to put a probability of $1/n$ on each data point which is known as the empirical distribution. Since we have the data, we can take as many (bootstrap) samples as we like from the empirical distribution. From each of these bootstrap samples, compute the statistic of interest (e.g. the sample mean). The key here is that the sampling for the bootstrap is done *with replacement.* From a data set of size $n$, a bootstrap sample is obtained by taking a random sample of size $n$ *with replacement* from the raw data.

**Bootstrap Standard Errors**

- To obtain bootstrap standard errors, one can resample the data *with replacement* to obtain a bootstrap sample.
- From the bootstrap sample, estimate the parameters using nonlinear regression.
- Repeat this process for a large number $B$ of bootstrap samples, say $B = 1000$.
- The standard error of the estimated parameter can be obtained by taking the standard deviation of the $B$ of bootstrap estimates.
- **Bootstrap Confidence Interval**: Arrange the bootstrap parameter estimates from smallest to largest and a 95% bootstrap confidence interval will be given by the range of the middle 95% of the bootstrap parameter estimates.

In a regression setting, the two popular ways of doing bootstrap sampling:

1. Resample the entire data pairs $(x_i, y_i)$.

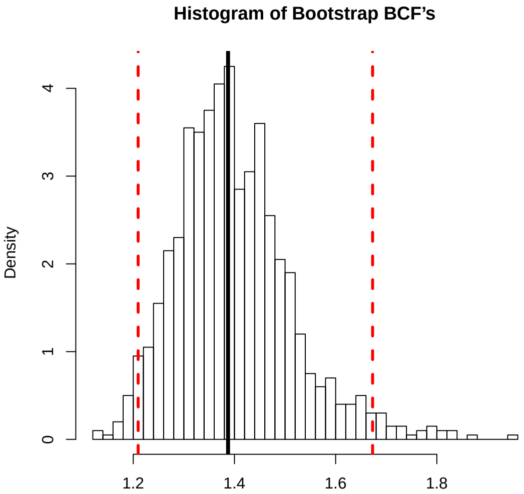

Figure 22: Bootstrap sampling distribution of BCF along with the estimated value of BCF (thick vertical line) and the bootstrap confidence interval indicated by the dashed vertical lines.

2. Resample the residuals from the original fit. Denote these bootstrap residuals by $r_i^*$. The bootstrap sample is then formed as $(x_i, \hat{y}_i + r_i^*)$, for $i = 1, \ldots, n$.

If the $x_i$'s are fixed (say by an experimental design) and not random, then the second method for bootstrapping in regression is preferable. Here is some R code to produce 1000 bootstrap estimates of the BCF in the pyrene example:

```
# Perform a bootstrap sampling to estimate the standard error of bcf:
resid <- resid(biofit)  # get the residuals from the nls fit
nboot <- 1000  # Number of bootstrap samples to obtain
bcfboot <- NULL # Store the results
for (j in 1:nboot){
  rboot <- sample(resid, replace=T) # re-sample the residuals
  yboot <- fitted(biofit) + rboot   # generate a bootstrap response
  bootfit <- nls(yboot ~ ku/kd*Ce*(1 - exp(-kd*t)),
             start = list( ku = coef(biofit)[1], kd = coef(biofit)[2]),
             trace = FALSE )
  bcfboot <- rbind(bcfboot,coef(bootfit)[1]/coef(bootfit)[2])
}
```

Note that we are resampling the residuals from the original fitted model and adding them to the fitted values to get our bootstrap samples. The R command "sample" will obtain a random sample and the command "replace=T" tells R to sample with replacement. The 1000 bootstrap BCF values are stored in "bcfboot". If we take the 2.5 and 97.5 percentiles of these values, we obtain a 95% bootstrap confidence interval. From R we have

```
> quantile(bcfboot, c(.025, .975))
    2.5%    97.5%
1.209644 1.673018
```

and we see that the bootstrap confidence interval for the BCF is $(1.21, 1.67)$ approximately. It is useful to plot a histogram of the bootstrap estimates along with the bootstrap confidence interval, which is shown in Figure 22. This figure was generated using the following R commands:

```
hist(bcfboot, 30, freq=F, xlab="", main="Histogram of Bootstrap BCF's")
abline(v=quantile(bcfboot, c(.025, .975)), col=2, lwd=3, lty=2)
abline(v=bcf, lwd=4)
```

The thick vertical line is the estimated BCF from the nonlinear regression model and the dashed vertical lines indicate the lower and upper confidence limits for the bootstrap confidence interval. Note that the histogram is clearly non-normal and skewed to the right. This is an indication that the asymptotic standard error may provide a poor approximation to the true standard error of our BCF estimate. In fact, typing "sd(bcfboot)" gives a bootstrap estimate of the standard error of 0.1149 which is bigger than our asymptotic estimate of 0.1076 indicating that the asymptotic standard error is a bit too optimistic (i.e. too small).

Another issue that has been ignored is that the environmental concentration $C_e$ is considered a known quantity, but in reality, the environmental concentration needs to be estimated as well. If the variability in the estimate of $C_e$ is large, then accounting for the variability in estimating $C_e$ will inflate the standard error for the estimated BCF (e.g. see Ogden and Tarpey, 2006).

# References

Anscombe, F. J. (1973). Graphs in statistical analysis. *The American Statistician* **27**:17–21.

Arretxe, M., Heap, J. M., and Christofi, N. (1997). The effect of toxic discharges on atp content in activated sludge. *Environmental Toxicology and Water Quality* **12**:23–29.

Bailer, A. J., W. S. E. and Venis, K. J. (2000). Estimating and testing bioconcentration factors. *Environmental Toxicology and Chemistry* **19**:2338–2340.

Bates, D. M. and Watts, D. G. (1988). *Nonlinear Regression Analysis and Its Applications.* Wiley, New York.

Box, G. E. P. (1980). Sampling and bayes' inference in scientific modeling and robustness. *Journal of the Royal Statistical Society, Series A* **143**:383–430.

Efron, B. and Tibshirani, R. (1993). *An introduction to the bootstrap.* Chapman & Hall, New York.

Friedman, J., Hastie, T., and Tibshirani, R. (2009). glmnet: Lasso and elastic-net regularized generalized linear models. r package version 1.1-4 .

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* **33**.

Hanley, J. A. and Shapiro, S. H. (1994). Sexual activity and the lifespan of male fruitflies: A dataset that gets attention. *Journal of Statistics Education* **2**.

Montgomery, D. C. and Peck, E. A. (1992). *Introduction to Linear Regression Analysis, 2nd Edition.* Wiley, New York.

Nalepa, T., Hartson, D., Buchanan, J., Covaletto, J., Lang, G., and Lozano, S. (2000). Spatial variation in density, mean size and physiological condition of holartic amphipod *diporeia* spp. in lake michigan. *Freshwater Biology* **43**:107–119.

Ogden, R. T. and Tarpey, T. (2006). Estimation in regression models with externally estimated parameters. *Biostatistics* **7**:115–129.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B* **58**:267–288.