Topic 2

Nonlinear Regression

Peter Tea

Department of Mathematics and Statistics
University of Ottawa

Contents

1 Introduction

In general, linear regression analysis involves the estimation of parameters of a linear model. The multiple linear model can be written as follows in matrix form:

$$Y = X\beta + \epsilon \tag{1}$$

where

$$Y = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}, X = \begin{bmatrix} 1 & x_{11} & \dots & x_{1k} \\ 1 & x_{21} & \dots & x_{2k} \\ 1 & x_{n1} & \dots & x_{nk} \end{bmatrix}, \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix}, \epsilon = \begin{bmatrix} \epsilon_0 \\ \epsilon_1 \\ \vdots \\ \epsilon_k \end{bmatrix}$$

- Y is a (n x 1) matrix representing the n different response variables
- X is a (n x p) matrix representing the k different predictor variables
- β is a (p x 1) matrix representing the p different parameters in the model. Here, $p = k + 1$.
- ϵ is a (n x 1) matrix representing the random errors.

We usually make assumptions about the error terms, ϵ. Most notably, we assume that the errors are:

$$\epsilon \sim N(0, \sigma^2) \tag{2}$$

By definition, for models to be considered linear, the model must be in the form of a linear combination of its parameters. For example, the following model

$$Y = \beta_0 + \beta_1 x_1 + ... + \beta_k x_k$$

is a linear model. Additionally, it can be shown that all derivatives of this linear model with respect to any of its p parameters are free from any and all parameters.

Similar to linear regression models, nonlinear regression models also attempt to model the relationship between response variables and predictor

variables. The relationship between these two variables is expressed through some expectation function and some error term of the form:

$$Y = f(X_n; \theta) + \epsilon \tag{3}$$

Where:

- Y is the nth response variable ($n = 1, 2, ..., N$) where N is the total number of observations
- $f(X_n; \theta)$ is the mean function, composed of regressors and parameters.
- x_n is the nth regressor variable
- θ is a parameter vector: $\theta = (\theta_1, \theta_2, ..., \theta_p)$ with p parameters in the model.
- ϵ is the nth error term which describes the distance from an observed value to its expected value, caused by some unknown source of variation.

2 Nonlinear Regression

Often viewed as an extension to linear regression, nonlinear regression is used when there exists some model that relates the response variable to the predictor variable where the parameters of the model are nonlinear. These nonlinear models are usually derived from theoretical investigations *(Seber 2003)*.

An example of a nonlinear model is the Michaelis-Menten expression which models the relationship between the rate of a reaction (the response variable) and the concentration of a substrate (the predictor variable). The expectation function is given as:

$$f(x, \theta) = \frac{\theta_1 x}{\theta_2 + x}$$

In this model, the parameters θ_1 and θ_2 represent the maximum velocity of the reaction and the Km (the substrate concentration at half the maximum velocity) respectively.

We can easily see that the parameters in the Michaelis-Menten model are not linear. By taking the derivatives of the expectation function with respect to its parameters (θ_1 and θ_2), we get the following results:

$$\frac{\partial f}{\partial \theta_1} = \frac{x}{\theta_2 + x}$$

$$\frac{\partial f}{\partial \theta_2} = \frac{-\theta_1 x}{(\theta_2 + x)^2}$$

The derivatives for both parameters are dependent on at least one of the unknown parameters, meaning that this model is indeed nonlinear.

Another distinction between nonlinear and linear regression is the relationship between the number of parameters and the number of predictor variables. For a linear regression model, the value of p (the number of unknown parameters) is usually defined as $k + 1$, where k is the number of predictors. However, for nonlinear regression models, p does not necessarily equal $k + 1$.
For example, consider the following nonlinear equation:

$$Y_i = \beta_0 + \beta_1 e^{-\beta_2 x_i} + \epsilon_i$$

Here, the model is nonlinear for the parameter β_2. Also, the model contains 3 parameters (p = 3) and only 1 predictor (k = 1), meaning that $p \neq k + 1$ *(Graybill 1994)*.

In some cases, nonlinear and linear regression (eg. polynomial linear regression) models fitted to some nonlinear dataset may both be able to explain the variance of the data. In fact, it is often much easier to estimate the unknown parameters of a linear model and also to perform statistical inferences on such linear parameters. However, the use of nonlinear models over linear models is often motivated when the nonlinear model is superior in capturing and explaining the underlying scientific phenomena linked to the experiment/dataset *(Lim, 2013)*.

2.1 Least Squares

Similar to linear regression, nonlinear regression can provide estimates of unknown parameters through least-squares method, if the assumption (2) holds.

The least squares method minimizes the residual sum of squares (RSS). I.e. The squared distance between the observed point and its corresponding fitted value:

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

for some response values $y_i = y_1, y_2, ..., y_n$ and some predictor values $x_i = x_1, x_2, ..., x_n$

Geometrically, the expectation function, $f(x_i; \theta)$ is often replaced with an N-dimensional expected response vector denoted $\eta(\theta)$, where $\eta(\theta) = f(x_i; \theta)$.

It can be shown that the calculation of these sum of squares is difficult for nonlinear cases since the expectation surface (also denoted as $\eta(\theta)$) is curved of finite extent *(Bates 1988)*. Therefore, iterative methods (eg. Gauss-Newton method which uses a linear approximation to the expectation function) are used to calculate the least squares estimates for nonlinear regression.

2.2 Iteration and Starting values

To use these iterative algorithms, an initial estimate of the parameters must be provided. Starting with these initial input parameter values, the algorithms will make adjustments to the parameters in order to improve the fit of the model (i.e. to further minimize the residual sums of squares). The iterative method stops when it can no longer improve the fit of the model.

Although iterative methods will converge to a minimum value for the residual sums of squares, there is no guarantee that the converged minimum value is a global minimum value rather than a local minimum value. However, if the initial values provided to the algorithm is close to the optimal parameter values, then the global minimisation will likely be reached

The initial starting values are purposefully chosen such that convergence is quickly obtained through the iterative methods. Usually, the parameters are meaningful biologically, graphically or in some other capacity which makes it easier to determine some sensible starting values. Many strategies

exist for choosing ideal starting values such as: analysing the expectation function (and its derivatives) in terms of the parameters, transforming the expectation function into a linear model such that linear least squares can be used to provide estimates of the parameters, and many other strategies.

For example, it can be noted that the Michaelis-Menten model can be transformed to a linear model (linear in its parameters) as follows:

$$\frac{1}{f} = \frac{1}{\theta_1} + \frac{\theta_2}{x\theta_1} \tag{4}$$

Initial estimates for the parameters θ_1 and θ_2 can then be easily obtained through linear least squares method.

Although transforming nonlinear models to induce linearity and then proceeding with analysis through linear regression may, at first glance, appear to be a good strategy, this technique is usually not recommended. These transformations cause the assumptions (2) of the nonlinear model to fail; the experimental error term, ϵ, becomes distorted following a transformation. The assumption that these error terms are normally distributed with a constant variance, σ^2, for the transformed data is usually not conserved following a transformation. The assumptions are only valid for the original model (3). Proceeding with nonlinear regression is instead recommended.

2.3 An example in R

With the Michaelis-Menten expression, we can model the relationship between the rate of a reaction and substrate concentration, with a dataset acquired from an enzyme kinetics experiment. The following *Puromycin* dataset explores how the rate of reaction differs when an enzyme is given puromycin (an enzyme inhibitor).

```
library(nlstools)
### Visualize the data
plot(rate ~ conc, data = Puromycin[Puromycin$state=="treated",],
     xlab = "Substrate concentration (ppm)",
     ylab = "Reaction rate (counts/min/min)"
     )
```

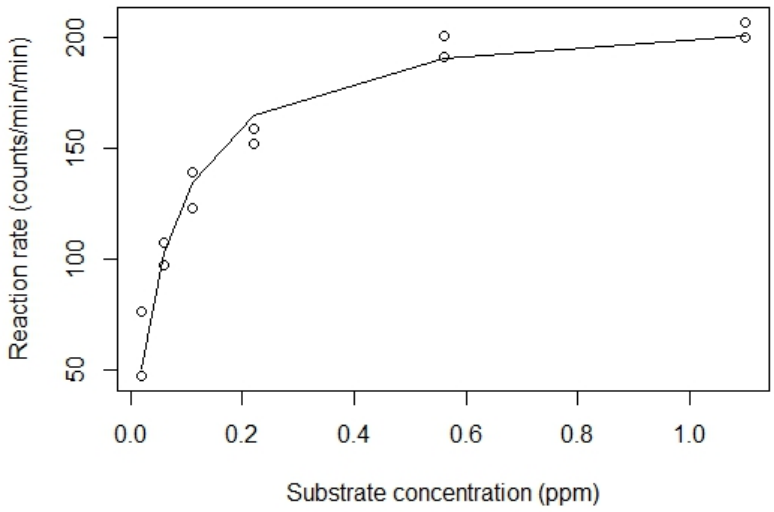

Figure 1: Plot of the effect of treated puromycin on enzyme kinetics

.

Due to the curvature of the plot, it is sensible to proceed with nonlinear regression analysis. The parameters in the Michaelis-Menten have biological meaning and can actually be deduced from figure 1. Vmax (θ_1) is defined as the maximum velocity, which is the horizontal asymptote near the value 200. Km (θ_1) is the substrate concentration at half the vmax. Looking at the graph, this value is approximately 0.05.

```
### Michaelis-Menten equation ('conc' is substrate concentration):
# rate = (Vmax * conc) / (Km + conc)

nonlinear.model <- nls(rate ~ (Vmax * conc) / (Km + conc),
#Set the nonlinear equation
                       data = Puromycin[Puromycin$state=="treated",],
                       start = c(Vmax = 200, Km = 0.05),
#Give the starting values
                       trace = TRUE
```

```
#Show all iteration values
                        )
#The results from left to right indicate: The residual sums of square value,
# the Vmax parameter estimate, the Km parameter estimate.

1636.586 :  2e+02 5e-02
1205.62 :  211.15721948   0.06162713
1195.573 :  212.51134162   0.06384178
1195.45 :  212.66623435   0.06409387
1195.449 :  212.68204562   0.06411864
1195.449 :  212.68357913   0.06412103
```

From the output, we see that the residual sums of squares is consistently decreasing until the value of 1195.449 is obtained. At this point, convergence to the parameter estimates is obtained ($\theta_1 \approx 212.68$ and $\theta_2 \approx 0.0641$). The confidence intervals for these estimates are calculated:

```
confint2(nonlinear.model, level = 0.95)
 2.5%        97.5%
Vmax 197.30212848 229.29006490
Km     0.04692517   0.08615995
```

The confidence intervals for the parameters Vmax and Km are (197.3, 229.3) and (0.047, 0.0861) respectively. For illustration purposes we can also show the detriments of linearly transforming the nonlinear data. The data can be modelled as illustrated in equation (4) as follows:

```
### Induce linearity to the data (for illustration purposes)
# First, create and add columns with the transformed data to the dataset
Puromycin$Inverse.conc <- with(Puromycin, 1/conc)
Puromycin$Inverse.rate <- with(Puromycin, 1/rate)

plot(Inverse.rate ~ Inverse.conc, data = Puromycin[Puromycin$state=="treated",],
     xlab = "Substrate concentration (ppm)^-1",
     ylab = "Reaction rate (counts/min/min)^-1"
  )
```

The plot of this model is illustrated in Figure 2. As previously stated, the assumptions of the error terms may not be appropriate following a transformation. We explore this notion below:

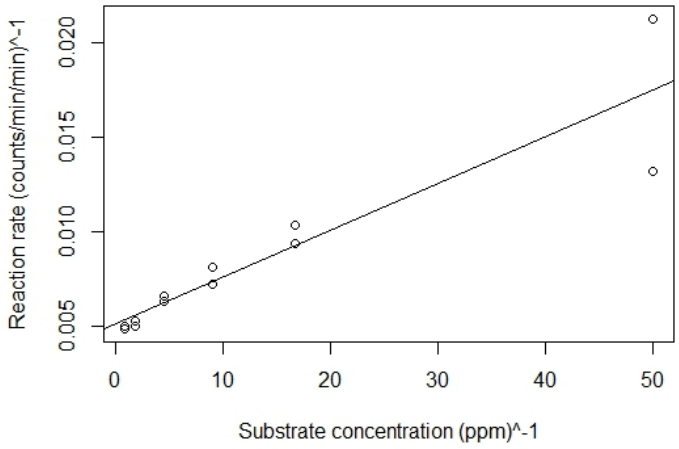

Figure 2: Linearly transformed plot of the effect of treated puromycin on enzyme kinetics

.

```
linear.model <- lm(Inverse.rate ~ Inverse.conc,
data = Puromycin[Puromycin$state=="treated",])

summary(model)
Call:
lm(formula = Inverse.rate ~ Inverse.conc, data = Puromycin[Puromycin$state ==
    "treated", ])
Residuals:
       Min         1Q     Median         3Q        Max
-0.0043103 -0.0003742 -0.0000510  0.0004549  0.0038084

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.0051072  0.0007040   7.255 2.74e-05 ***
Inverse.conc 0.0002472  0.0000321   7.700 1.64e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
Residual standard error: 0.001892 on 10 degrees of freedom
Multiple R-squared:  0.8557,Adjusted R-squared:  0.8413
F-statistic:  59.3 on 1 and 10 DF,  p-value: 1.642e-05
```

The results seem to indicate significance since the p-value is below 0.05. However, upon further analysis it is clear that these results should not be trusted.

```
library(car)
ncvTest(linear.model) # H_0 is that there is constant variance.
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 22.73466    Df = 1     p = 1.85983e-06
```

The null hypothesis of this non-constant variance test is rejected, meaning that the assumption of constant variance is not valid. While this linearly transformed model failed, it can still be used to provide estimates for the unknown parameters (to be used in the iteration nonlinear least squares method.) From the summary output:

$$\text{Intercept} = 0.0051072 = \frac{1}{\theta_1}$$

$$\theta_1 = \frac{1}{0.0051072}, \approx 195.8$$

And:

$$0.0002472 = \frac{\theta_2}{\theta_1}$$

$$\theta_2 = 0.0002472x195.8, \approx 0.048$$

```
 #Use the estimation from linear regression as starting values:
nonlinear.model2 <- nls(rate ~ (Vmax * conc) / (Km + conc),
     data = Puromycin[Puromycin$state=="treated",],
     start = c(Vmax = 195.8, Km = 0.048),
```

```
     trace = FALSE

> nonlinear.model2
Nonlinear regression model
  model: rate ~ (Vmax * conc)/(Km + conc)
   data: Puromycin[Puromycin$state == "treated", ]
     Vmax        Km
212.68372   0.06412
 residual sum-of-squares: 1195

Number of iterations to convergence: 6
Achieved convergence tolerance: 9.991e-07
```

Using the estimates from the transformed model, the iterative method was able to output parameter estimates which were very similar to the ones using the previous starting values.

The least squares parameter estimates can be accompanied with contour plots which illustrate their marginal inference intervals.

```
nonlinear.model.contour <- nlsContourRSS(nonlinear.model)
# Obtain contour plots for the residual sum of squares of a
combination of any two paramter values:

plot(nonlinear.model.contour, col = FALSE,
     nlev = 10)
```

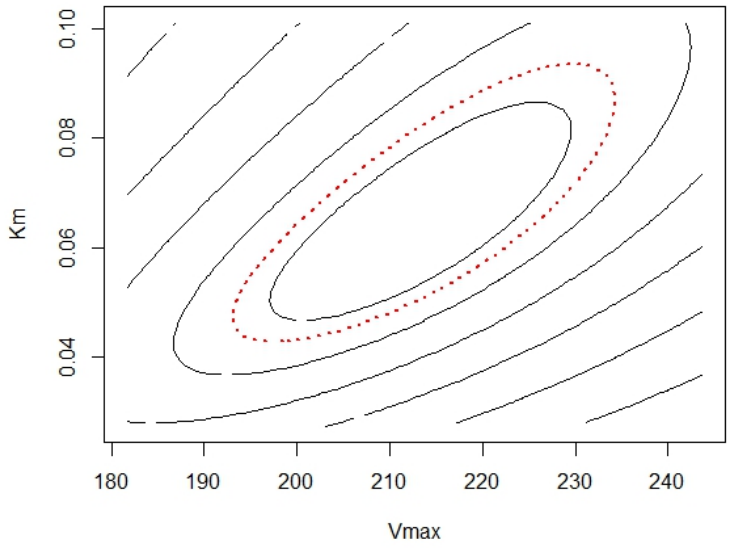

Figure 3: Contour plot for the least squares estimates of the two parameters in the Puromycin data. The dashed region indicates the 95% confidence region for the two parameters.

3 Model Adequacy

3.1 Assumptions

As is the case for linear regression, the underlying assumptions of nonlinear regression must also be validated before we make inferences from our analysis. Regression models of the form described by equation (3) make the following assumptions:

- Homogeneity of variance (The error terms all have the same constant variance)
- Error terms are normally distributed
- Error terms are mutually independent

Should these assumptions fail, then the generated parameter estimates will be inaccurate (I.e. they will have distorted standard errors).With residual plots, we can easily assess the assumptions of homogeneity of variance (also called homoscedasticity) and of normality.

If the residuals are spread in a random manner around the horizontal line of a *fitted values vs residual plot*, then it is good indication that the assumption of variance homogeneity is satisfied. The normality assumption is verified through a QQ plot.

There exists many strategies when assumptions fail. For example, data transformation can be used in order to stabilize non constant variances in a model. For linear regression, it is common to only transform the response variable and not modify the model. For nonlinear regression, the mean function clearly defines the fundamental relationship between the response and predictor, so to preserve this relationship we must apply the same transformation to both the left side and right side of equation (3).

$$h(Y) = h(f(X_n; \theta))$$

For example, if a logarithmic transformation is appropriate, then we would fit the logarithm of the mean function to the logarithms of the data.

3.2 An example in R

The following dataset is derived from an experiment which tests the effects of two different herbicides on the biomass of the

plant *Sinapis alba*. The experiment can be classified as a dose-response experiment, where the response *drymass* of the plant is recorded for a range of doses of the herbicide. Dose-response experiments can be modelled with the four parameter logistic regression model:

$$f(dose; b, c, d, e) = b + \frac{c - b}{1 + exp[\frac{d - dose}{e}]}$$

Where:

- c is the lower horizontal asymptote
- b is the upper horizontal asymptote
- d is the inflection point
- e is the slope at the inflection point c.

We will only analyse one of the herbicides (glyphosate):

```
library(nlstools)
Glyphosate <- S.alba[S.alba$Herbicide=="Glyphosate",]
#Subset the data.

Glyphosate$Dose[Glyphosate$Dose== 0]<- 0.001
#Change values of 0 into non-zero values.

plot(DryMatter ~ Dose, data = Glyphosate, log="x",
     xlab = "Dose",
     ylab = "drymatter"
)
```

The plot of the figure, along with its superimposed nonlinear regression model will be presented in Figure 4.

We then move onto finding the least square estimates for this model:

```
nonlinear.model <- nls(DryMatter ~ SSfpl(Dose,b,c,d,e),
                       data = Glyphosate)
  #SSfpl is a built in function which calculates
  some starting values for the parameters in the
  specified logistic model.

summary(nonlinear.model)
Formula: DryMatter ~ SSfpl(Dose, b, c, d, e)

Parameters:
  Estimate Std. Error t value Pr(>|t|)
b   4.1241     0.3076  13.406 1.05e-13 ***
c   1.0025     0.1503   6.669 3.08e-07 ***
d  59.1090     7.9553   7.430 4.31e-08 ***
e  22.0614     6.9828   3.159  0.00377 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.4022 on 28 degrees of freedom

Number of iterations to convergence: 5
Achieved convergence tolerance: 3.899e-06


#Write the expectation function:
expfct <- function(Dose, b, c, d, e)
                  {
b + ((c-b)/ (1 + exp((d-Dose)/e)))
```

```
}


curve(expfct(x, b = 4.1241, c = 1,
                d = 59, e = 22.06), add = TRUE)
     #Fit the model with its estimated paramters to the data
```

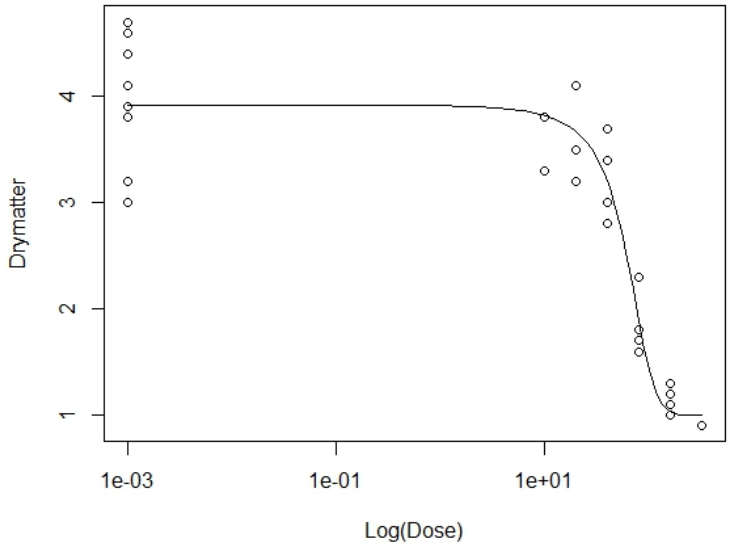

Figure 4: The dose-response curve of the effects of glyphosate on the drymass of S.Alba. The nonlinear regression model is superimposed on the date.

Oftentimes, we would like to verify whether the fitted model can be simplified by removing some "extra" parameters that are not needed. Thus, we test the hypotheses:

$$H_0 : \beta_j = \beta_{0j}$$

where j represents the jth parameter in the model and $\beta_{0j} = 0$. From the summary output, we see that the p-values associated with each fitted parameter is significant at the level of 0.05.

Thus, we can reject the null hypothesis that (for each case) the parameter is equal to zero. Additionally, if we looked at the confidence intervals for each parameter, none of the intervals contain the value zero, which agrees well with the results of the previous tests.

```
confint2(nonlinear.model)
       2.5 %    97.5 %
b  3.4939182  4.754288
c  0.6946109  1.310424
d 42.8127830 75.404799
e  7.7576256 36.365313
```

We can also perform a lack of fit test for our fitted nonlinear regression model. Here, we compare the nonlinear model to a more general ANOVA model. We test that the ANOVA model can be reduced to the nonlinear model:

```
anova.model <- lm(DryMatter ~ as.factor(Dose),
                  data = Glyphosate)

anova(nonlinear.model, anova.model)
Analysis of Variance Table

Model 1: DryMatter ~ SSfpl(Dose, b, c, d, e)
Model 2: DryMatter ~ as.factor(Dose)
  Res.Df Res.Sum Sq Df  Sum Sq F value Pr(>F)
1     28     4.5296
2     25     4.3213  3 0.20833  0.4018  0.753
```

The p-value was not significant at the level of 0.05. Therefore, we cannot reject the null hypothesis that the ANOVA model can be simplified into the nonlinear model.

Continuing with model diagnostics:

```
###Model Diagnostics
plot(nonlinear.model)
```

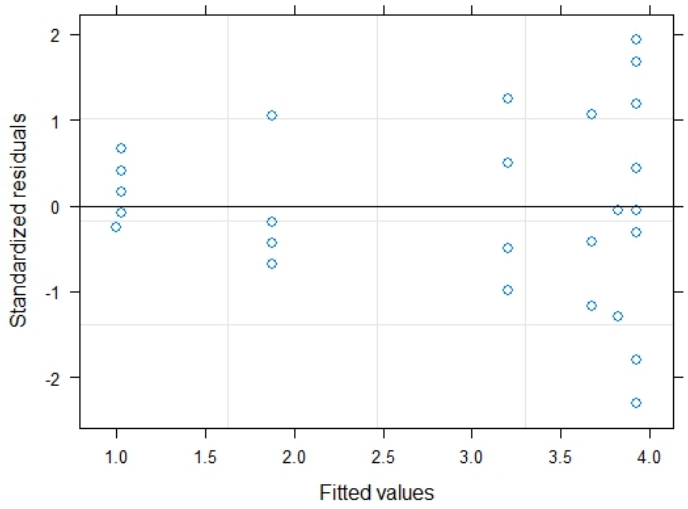

Figure 5: Fitted values vs. standardized residuals plot. There appears to be a random spread of residuals about the horizontal line, indicating the presence of constant variance.

For a more formal test on constant variance:

```
library(car)
with(Glyphosate, leveneTest(DryMatter,as.factor(Dose)))
#H_0 There is constant variance

Levene's Test for Homogeneity of Variance (center = median)
      Df F value  Pr(>F)
group  6  2.9989 0.02394 *
```

From the Levene's test, we would reject the null hypothesis that there is constant variance in our fitted model.

Now looking at the normality assumption:

```
Standard.Residuals<- residuals(nonlinear.model)/
summary(nonlinear.model)$sigma
  #This calculates the standardised residuals for the
  nonlinear model.
  #Standard residuals = Raw residuals / residual standard error
  #summary(nonlinear.model)$sigma attaches to the
  residual standard error.
```

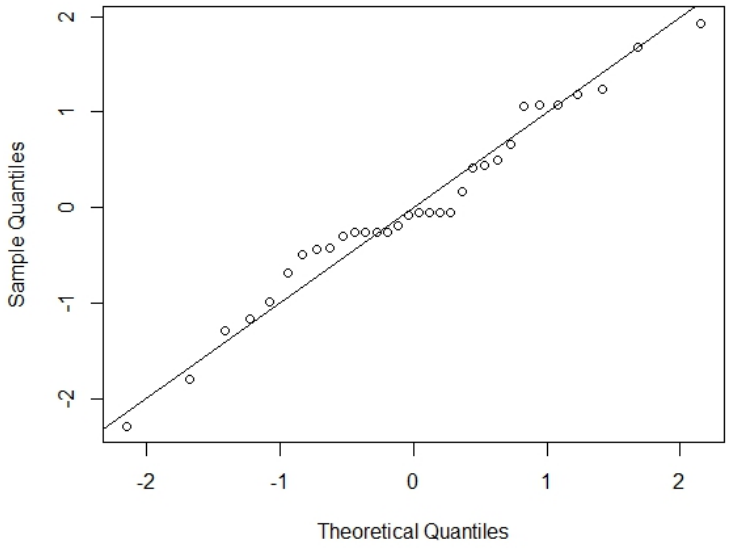

Figure 6: QQ plot to assess the normality assumption.

For a formal test on normality:

```
qqnorm(Standard.Residuals, main = "")
abline(a = 0, b = 1)
```

```
shapiro.test(Standard.Residuals)
  #H_0 Normality assumption is valid.
Shapiro-Wilk normality test

data:  Standard.Residuals
W = 0.97257, p-value = 0.573
```

From the generated QQ plot it appears as though the points approximately follow a straight line indicating that the distribution is normal. The formal Shapiro-Wilk test also indicates a valid normality assumption since there was no concluded significance.

Since the homogeneity of variance assumption was not met, we may be concerned about accepting the inferences of the model. We can estimate the parameters of the model again, but this time using a resampling bootstrap method. Bootstrap is a non-parametric technique and thus does not make any assumptions on the distribution of the data.

```
#Bootstrap
boot <- nlsBoot(nonlinear.model)
summary(boot)

------
Bootstrap statistics
   Estimate Std. error
b  4.274211  0.9171432
c  0.986067  0.1497958
d 56.699689 13.2896604
e 23.063219  9.0169736
```

```
------
Median of bootstrap estimates and percentile confidence intervals
      Median       2.5%     97.5%
b  4.1137319  3.7386115  5.771977
c  0.9897168  0.6837052  1.271249
d 58.5388941 29.6443061 71.972291
e 20.8920763 12.6547005 46.225334
```

If we compare the bootstrap estimates (and their confidence intervals) to our original least squares estimates, the parameter values appear to be very similar. Thus, we might be willing to accept our original inferences of the parameter estimates.

4 Works Cited

Bates, Douglas, Watts, Donald. Nonlinear Regression Analysis and Its Applications. 1988. John Wiley & Sons, Inc.

Graybill, Franklin, Iyer, Hariharan. Regression Analysis: Concepts and Applications. Ch 9 Nonlinear Regression. 1994. 599-607. Duxbury Pr.

Lim, Changwon, et.al. Robust nonlinear regression in applications. Journal of the Indian society of Agricultural Science. 2013.
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4286339/

Motulsky, Harvey, Christopoulos, Arthur. Fitting models to Biological Data using Linear and Nonlinear Regression. A practical guide to curve fitting. GraphPad Software Inc, San Diego CA. 2003. Print.

Ritz, Christian, Streibig Jens. Nonlinear Regression with R. 2008. Springer.

Seber, George A. F, Wild, C. J. Nonlinear Regression. 2003. Hoboken, NJ: Wiley-Interscience. Print.