Chapter 6

Testing for Lack of Fit

How can we tell if a model fits the data? If the model is correct then $\hat{\sigma}^2$ should be an unbiased estimate of σ^2. If we have a model which is not complex enough to fit the data or simply takes the wrong form, then $\hat{\sigma}^2$ will overestimate σ^2. An example can be seen in Figure 6.1. Alternatively, if our model is too complex and overfits the data, then $\hat{\sigma}^2$ will be an underestimate.

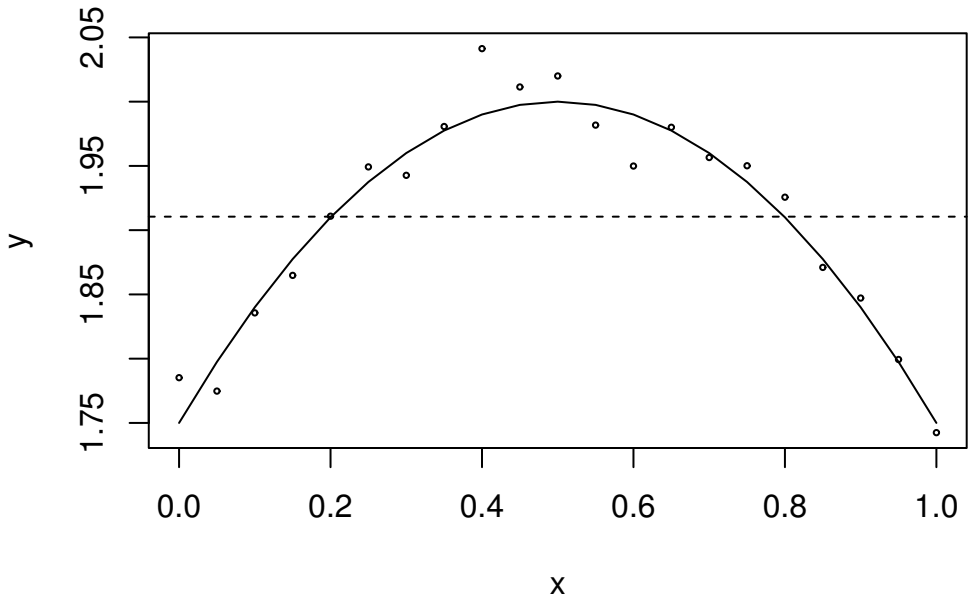

Figure 6.1: True quadratic fit shown with the solid line and incorrect linear fit shown with the dotted line. Estimate of σ^2 will be unbiased for the quadratic model but far too large for the linear model

This suggests a possible testing procedure — we should compare $\hat{\sigma}^2$ to σ^2. There are two cases — one where σ^2 is known and one where it is not.

6.1 σ^2 known

σ^2 known may be known from past experience, knowledge of the measurement error inherent in an instrument or by definition. Recall (from Section 3.4) that

$$\frac{\hat{\sigma}^2}{\sigma^2} \sim \frac{\chi^2_{n-p}}{(n-p)}$$

which leads to the test: Conclude there is a lack of fit if

$$\frac{(n-p)\hat{\sigma}^2}{\sigma^2} > \chi^{2\ (1-\alpha)}_{n-p}$$

If a lack of fit is found, then a new model is needed.

Continuing with the same data as in the weighted least squares example we test to see if a linear model is adequate. In this example, we know the variance almost exactly because each response value is the average of a large number of observations. Because of the way the weights are defined, $w_i = 1/\text{var}\ y_i$, the known variance is implicitly equal to one. There is nothing special about one - we could define $w_i = 99/\text{var}\ y_i$ and the variance would be implicitly 99. However, we would get essentially the same result as the following analysis.

```
> data(strongx)
> g <- lm(crossx ~ energy, weights=sd^-2, strongx)
> summary(g)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   148.47       8.08    18.4  7.9e-08
energy        530.84      47.55    11.2  3.7e-06

Residual standard error: 1.66 on 8 degrees of freedom
Multiple R-Squared: 0.94,       Adjusted R-squared: 0.932
F-statistic:  125 on 1 and 8 degrees of freedom,        p-value: 3.71e-06
```

Examine the R^2 - do you think the model is a good fit?

Now plot the data and the fitted regression line (shown as a solid line on Figure 6.2.

```
> plot(strongx$energy,strongx$crossx,xlab="Energy",ylab="Crossection")
> abline(g$coef)
```

Compute the test statistic and the p-value:

```
> 1.66^2*8
[1] 22.045
> 1-pchisq(22.045,8)
[1] 0.0048332
```

We conclude that there is a lack of fit. Just because R^2 is large does not mean that you can not do better. Add a quadratic term to the model and test again:

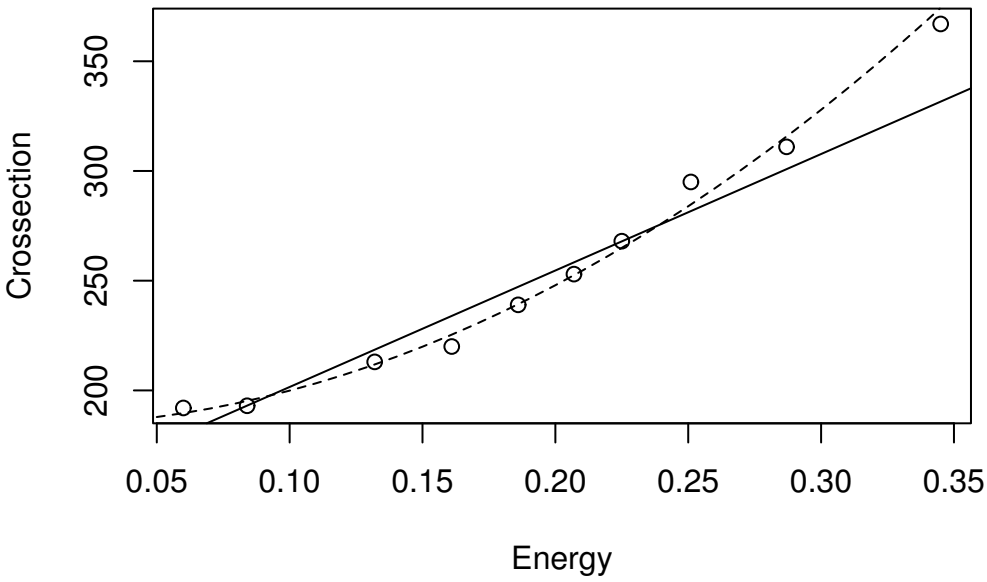

Figure 6.2: Linear and quadratic fits to the physics data

```
> g2 <- lm(crossx ~ energy + I(energy^2), weights=sd^-2, strongx)
> summary(g2)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  183.830      6.459   28.46  1.7e-08
energy         0.971     85.369    0.01  0.99124
I(energy^2) 1597.505    250.587    6.38  0.00038

Residual standard error: 0.679 on 7 degrees of freedom
Multiple R-Squared: 0.991,      Adjusted R-squared: 0.989
F-statistic:  391 on 2 and 7 degrees of freedom,        p-value: 6.55e-08
> 0.679^2*7
[1] 3.2273
> 1-pchisq(3.32273,7)
[1] 0.85363
```

This time we cannot detect a lack of fit. Plot the fit:

```
> x <- seq(0.05,0.35,by=0.01)
> lines(x,g2$coef[1]+g2$coef[2]*x+g2$coef[3]*x^2,lty=2)
```

The curve is shown as a dotted line on the plot (thanks to `lty=2`). This seems clearly more appropriate than the linear model.

6.2 σ^2 unknown

The $\hat{\sigma}^2$that is based in the chosen regression model needs to be compared to some model-free estimate of σ^2. We can do this if we have repeated y for one or more fixed x. These replicates do need to be

truly independent. They cannot just be repeated measurements on the same subject or unit. Such repeated measures would only reveal the within subject variability or the measurement error. We need to know the between subject variability — this reflects the σ^2 described in the model.

The "pure error" estimate of σ^2 is given by SS_{pe}/df_{pe} where

$$SS_{pe} = \sum_{\text{distinct x}} \sum_{given x} (y_i - \bar{y})^2$$

Degrees of freedom $df_{pe} = \sum_{distinct x}(\#replicates - 1)$

If you fit a model that assigns one parameter to each group of observations with fixed x then the $\hat{\sigma}^2$ from this model will be the pure error $\hat{\sigma}^2$. This model is just the one-way anova model if you are familiar with that. Comparing this model to the regression model amounts to the lack of fit test. This is usually the most convenient way to compute the test but if you like we can then partition the RSS into that due to lack of fit and that due to the pure error as in Table 6.1.

	df	SS	MS	F
Residual	n-p	RSS		
Lack of Fit	$n-p-df_{pe}$	$RSS - SS_{pe}$	$\frac{RSS-SS_{pe}}{n-p-df_{pe}}$	Ratio of MS
Pure Error	df_{pe}	SS_{pe}	SS_{pe}/df_{pe}	

Table 6.1: ANOVA for lack of fit

Compute the F-statistic and compare to $F_{n-p-df_{pe},df_{pe}}$ and reject if the statistic is too large.

Another way of looking at this is a comparison between the model of interest and a saturated model that assigns a parameter to each unique combination of the predictors. Because the model of interest represents a special case of the saturated model where the saturated parameters satisfy the constraints of the model of interest, we can use the standard F-testing methodology.

The data for this example consist of thirteen specimens of 90/10 Cu-Ni alloys with varying iron content in percent. The specimens were submerged in sea water for 60 days and the weight loss due to corrosion was recorded in units of milligrams per square decimeter per day. The data come from Draper and Smith (1998).

We load in and print the data

```
> data(corrosion)
> corrosion
     Fe  loss
1  0.01 127.6
2  0.48 124.0
3  0.71 110.8
4  0.95 103.9
5  1.19 101.5
6  0.01 130.1
7  0.48 122.0
8  1.44  92.3
9  0.71 113.1
10 1.96  83.7
11 0.01 128.0
12 1.44  91.4
13 1.96  86.2
```

We fit a straight line model:

```
> g <- lm(loss ~ Fe, data=corrosion)
> summary(g)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   129.79       1.40    92.5  < 2e-16
Fe            -24.02       1.28   -18.8  1.1e-09

Residual standard error: 3.06 on 11 degrees of freedom
Multiple R-Squared: 0.97,       Adjusted R-squared: 0.967
F-statistic:  352 on 1 and 11 degrees of freedom,       p-value: 1.06e-09
```

Check the fit graphically — see Figure 6.3.

```
> plot(corrosion$Fe,corrosion$loss,xlab="Iron content",ylab="Weight loss")
> abline(g$coef)
```

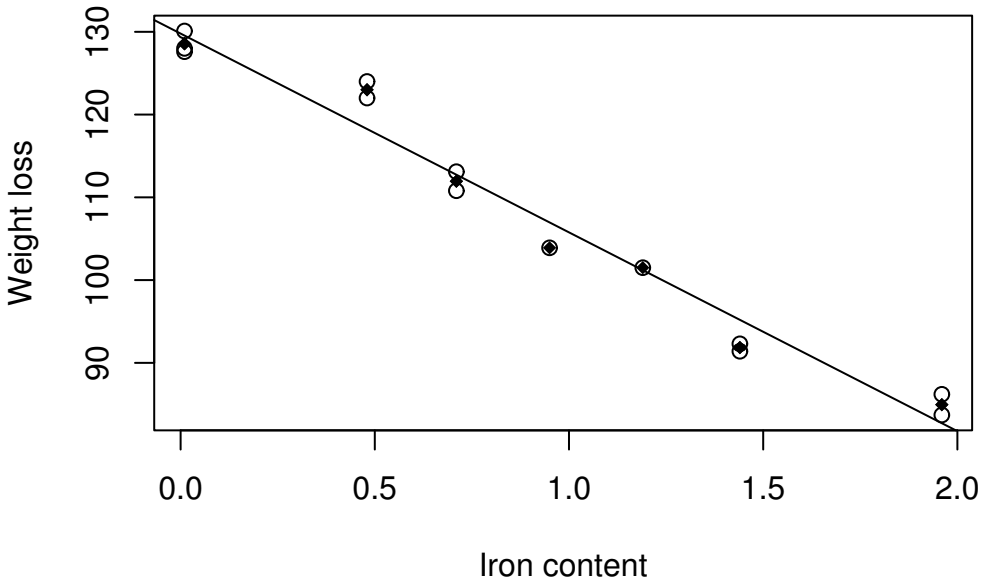

Figure 6.3: Linear fit to the Cu-Ni corrosion data. Group means denoted by black diamonds

We have an R^2 of 97% and an apparently good fit to the data. We now fit a model that reserves a parameter for each group of data with the same value of x. This is accomplished by declaring the predictor to be a factor. We will describe this in more detail in a later chapter

```
> ga <- lm(loss ~ factor(Fe), data=corrosion)
```

The fitted values are the means in each group - put these on the plot:

```
> points(corrosion$Fe,ga$fit,pch=18)
```

We can now compare the two models in the usual way:

```
> anova(g,ga)
Analysis of Variance Table

Model 1: loss ~ Fe
Model 2: loss ~ factor(Fe)
  Res.Df Res.Sum Sq Df Sum Sq F value Pr(>F)
1     11      102.9
2      6       11.8  5   91.1    9.28 0.0086
```

The low p-value indicates that we must conclude that there is a lack of fit. The reason is that the pure error sd $\sqrt{(11.8/6)} = 1.4$ is substantially less than the regression standard error of 3.06. We might investigate models other than a straight line although no obvious alternative is suggested by the plot. Before considering other models, I would first find out whether the replicates are genuine — perhaps the low pure error SD can be explained by some correlation in the measurements. Another possible explanation is unmeasured third variable is causing the lack of fit.

When there are replicates, it is impossible to get a perfect fit. Even when there is parameter assigned to each group of x-values, the residual sum of squares will not be zero. For the factor model above, the R^2 is 99.7%. So even this saturated model does not attain a 100% value for R^2. For these data, it's a small difference but in other cases, the difference can be substantial. In these cases, one should realize that the maximum R^2 that may be attained might be substantially less than 100% and so perceptions about what a good value for R^2 should be downgraded appropriately.

These methods are good for detecting lack of fit, but if the null hypothesis is accepted, we cannot conclude that we have the true model. After all, it may be that we just did not have enough data to detect the inadequacies of the model. All we can say is that the model is not contradicted by the data.

When there are no replicates, it may be possible to group the responses for similar x but this is not straightforward. It is also possible to detect lack of fit by less formal, graphical methods.

A more general question is how good a fit do you really want? By increasing the complexity of the model, it is possible to fit the data more closely. By using as many parameters as data points, we can fit the data exactly. Very little is achieved by doing this since we learn nothing beyond the data itself and any predictions made using such a model will tend to have very high variance. The question of how complex a model to fit is difficult and fundamental. For example, we can fit the mean responses for the example above exactly using a sixth order polynomial:

```
> gp <- lm(loss ~ Fe+I(Fe^2)+I(Fe^3)+I(Fe^4)+I(Fe^5)+I(Fe^6),corrosion)
```

Now look at this fit:

```
> plot(loss ~ Fe, data=corrosion,ylim=c(60,130))
> points(corrosion$Fe,ga$fit,pch=18)
> grid <- seq(0,2,len=50)
> lines(grid,predict(gp,data.frame(Fe=grid)))
```

as shown in Figure 6.4. The fit of this model is excellent — for example:

```
> summary(gp)$r.squared
[1] 0.99653
```

but it is clearly riduculous. There is no plausible reason corrosion loss should suddenly drop at 1.7 and thereafter increase rapidly. This is a consequence of overfitting the data. This illustrates the need not to become too focused on measures of fit like R^2.

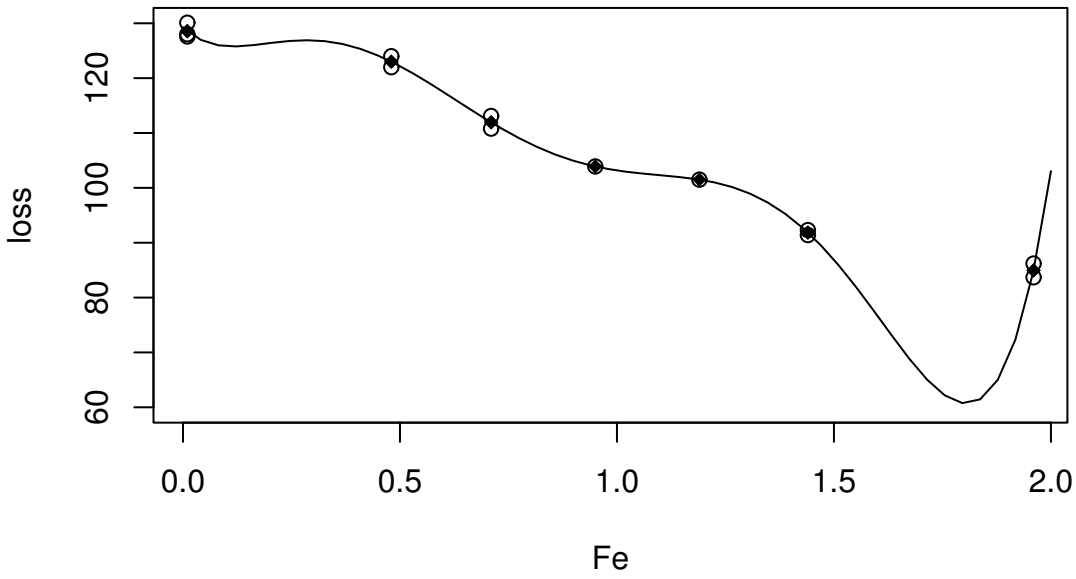

Figure 6.4: Polynomial fit to the corrosion data