Chapter 5

Generalized Least Squares

5.1 The general case

Until now we have assumed that var $\varepsilon = \sigma^2 I$ but it can happen that the errors have non-constant variance or are correlated. Suppose instead that var $\varepsilon = \sigma^2 \Sigma$ where σ^2 is unknown but Σ is known — in other words we know the correlation and relative variance between the errors but we don't know the absolute scale.

Generalized least squares minimizes

$$(y - X\beta)^T \Sigma^{-1} (y - X\beta)$$

which is solved by

$$\hat{\beta} = (X^T \Sigma^{-1} X)^{-1} X^T \Sigma^{-1} y$$

Since we can write $\Sigma = SS^T$, where S is a triangular matrix using the Choleski Decomposition, we have

$$(y - X\beta)^T S^{-T} S^{-1} (y - X\beta) = (S^{-1}y - S^{-1}X\beta)^T (S^{-1}y - S^{-1}X\beta)$$

So GLS is like regressing $S^{-1}X$ on $S^{-1}y$. Furthermore

$$\begin{aligned} y &= X\beta + \varepsilon \\ S^{-1}y &= S^{-1}X\beta + S^{-1}\varepsilon \\ y' &= X'\beta + \varepsilon' \end{aligned}$$

So we have a new regression equation $y' = X'\beta + \varepsilon'$ where if we examine the variance of the new errors, ε' we find

$$\text{var } \varepsilon' = \text{var } (S^{-1}\varepsilon) = S^{-1}(\text{var } \varepsilon)S^{-T} = S^{-1}\sigma^2 SS^T S^{-T} = \sigma^2 I$$

So the new variables y' and X' are related by a regression equation which has uncorrelated errors with equal variance. Of course, the practical problem is that Σ may not be known.

We find that

$$\text{var } \hat{\beta} = (X^T \Sigma^{-1} X)^{-1}\sigma^2.$$

To illustrate this we'll use a built-in R dataset called Longley's regression data where the response is number of people employed, yearly from 1947 to 1962 and the predictors are GNP implicit price deflator (1954=100), GNP, unemployed, armed forces, noninstitutionalized population 14 years of age and over, and year. The data originally appeared in Longley (1967)

Fit a linear model.

```
> data(longley)
> g <- lm(Employed ~ GNP + Population, data=longley)
> summary(g,cor=T)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  88.9388    13.7850    6.45  2.2e-05
GNP           0.0632     0.0106    5.93  5.0e-05
Population   -0.4097     0.1521   -2.69    0.018

Residual standard error: 0.546 on 13 degrees of freedom
Multiple R-Squared: 0.979,      Adjusted R-squared: 0.976
F-statistic:  304 on 2 and 13 degrees of freedom,      p-value: 1.22e-11

Correlation of Coefficients:
            (Intercept)     GNP
GNP             0.985
Population     -0.999 -0.991
```

Compare the correlation between the variables gnp, pop and their corresponding coefficients. What do you notice?

In data collected over time such as this, successive errors could be correlated. Assuming that the errors take a simple autoregressive form:

$$\varepsilon_{i+1} = \rho\varepsilon_i + \delta_i$$

where $\delta_i \sim N(0, \tau^2)$. We can estimate this correlation ρ by

```
> cor(g$res[-1],g$res[-16])
[1] 0.31041
```

Under this assumption $\Sigma_{ij} = \rho^{|i-j|}$. For simplicity, lets assume we know that $\rho = 0.31041$. We now construct the Σ matrix and compute the GLS estimate of β along with its standard errors.

```
> x <- model.matrix(g)
> Sigma <- diag(16)
> Sigma <- 0.31041^abs(row(Sigma)-col(Sigma))
> Sigi <- solve(Sigma)
> xtxi <- solve(t(x) %*% Sigi %*% x)
> beta <- xtxi %*% t(x) %*% Sigi %*% longley$Empl
> beta
          [,1]
[1,] 94.89889
[2,]  0.06739
[3,] -0.47427
> res <- longley$Empl - x %*% beta
> sig <- sqrt(sum(res^2)/g$df)
> sqrt(diag(xtxi))*sig
[1] 14.157603  0.010867  0.155726
```

Compare with the model output above where the errors are assumed to be uncorrelated.

Another way to get the same result is to regress $S^{-1}y$ on $S^{-1}x$ as we demonstrate here:

```
> sm <- chol(Sigma)
> smi <- solve(t(sm))
> sx <- smi %*% x
> sy <- smi %*% longley$Empl
> lm(sy ~ sx-1)$coef
sx(Intercept)          sxGNP   sxPopulation
     94.89889        0.06739       -0.47427
```

In practice, we would not know that the $\rho = 0.31$ and we'd need to estimate it from the data. Our initial estimate is 0.31 but once we fit our GLS model we'd need to re-estimate it as

```
> cor(res[-1],res[-16])
[1] 0.35642
```

and then recompute the model again with $\rho = 0.35642$. This process would be iterated until convergence.

The `nlme` library contains a GLS fitting function. We can use it to fit this model:

```
> library(nlme)
> g <- gls(Employed ~ GNP + Population,
  correlation=corAR1(form= ~Year), data=longley)
> summary(g)
Correlation Structure: AR(1)
 Formula: ~Year
 Parameter estimate(s):
    Phi
0.64417

Coefficients:
             Value Std.Error t-value p-value
(Intercept) 101.858   14.1989  7.1736  <.0001
GNP           0.072    0.0106  6.7955  <.0001
Population   -0.549    0.1541 -3.5588  0.0035

Residual standard error: 0.68921
Degrees of freedom: 16 total; 13 residual
```

We see that the estimated value of ρ is 0.64. However, if we check the confidence intervals for this:

```
> intervals(g)
Approximate 95% confidence intervals

 Coefficients:
               lower        est.       upper
(Intercept) 71.183204 101.858133 132.533061
GNP          0.049159   0.072071   0.094983
Population  -0.881491  -0.548513  -0.215536
```

```
 Correlation structure:
       lower     est.    upper
Phi -0.44335 0.64417 0.96451


 Residual standard error:
   lower     est.    upper
0.24772 0.68921 1.91748
```

we see that it is not significantly different from zero.

5.2 Weighted Least Squares

Sometimes the errors are uncorrelated, but have unequal variance where the form of the inequality is known. Weighted least squares (WLS) can be used in this situation. When Σ is diagonal, the errors are uncorrelated but do not necessarily have equal variance. We can write $\Sigma = \text{diag}(1/w_1,\ldots,1/w_n)$, where the w_i are the weights so $S = \text{diag}(\sqrt{1/w_1},\ldots,\sqrt{1/w_n})$. So we can regress $\sqrt{w_i}x_i$ on $\sqrt{w_i}y_i$ (although the column of ones in the X-matrix needs to be replaced with $\sqrt{w_i}$. Cases with low variability should get a high weight, high variability a low weight. Some examples

1. Errors proportional to a predictor: $\text{var}(\varepsilon_i) \propto x_i$ suggests $w_i = x_i^{-1}$

2. Y_i are the averages of n_i observations then $\text{var } y_i = \text{var } \varepsilon_i = \sigma^2/n_i$ suggests $w_i = n_i$.

Here's an example from an experiment to study the interaction of certain kinds of elementary particles on collision with proton targets. The experiment was designed to test certain theories about the nature of the strong interaction. The cross-section(crossx) variable is believed to be linearly related to the inverse of the energy(energy - has already been inverted). At each level of the momentum, a very large number of observations were taken so that it was possible to accurately estimate the standard deviation of the response(sd).
Read in and check the data:

```
> data(strongx)
> strongx
   momentum energy  crossx sd
1         4   0.345    367 17
2         6   0.287    311  9
3         8   0.251    295  9
4        10   0.225    268  7
5        12   0.207    253  7
6        15   0.186    239  6
7        20   0.161    220  6
8        30   0.132    213  6
9        75   0.084    193  5
10      150   0.060    192  5
```

Define the weights and fit the model:

```
> g <- lm(crossx ~ energy, strongx, weights=sd^-2)
> 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
```

Try fitting the regression without weights and see what the difference is.

```
> gu <- lm(crossx ~ energy, strongx)
> summary(gu)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    135.0        10.1    13.4  9.2e-07
energy         619.7        47.7    13.0  1.2e-06

Residual standard error: 12.7 on 8 degrees of freedom
Multiple R-Squared: 0.955,       Adjusted R-squared: 0.949
F-statistic:  169 on 1 and 8 degrees of freedom,        p-value: 1.16e-06
```

The two fits can be compared

```
> plot(crossx ~ energy, data=strongx)
> abline(g)
> abline(gu,lty=2)
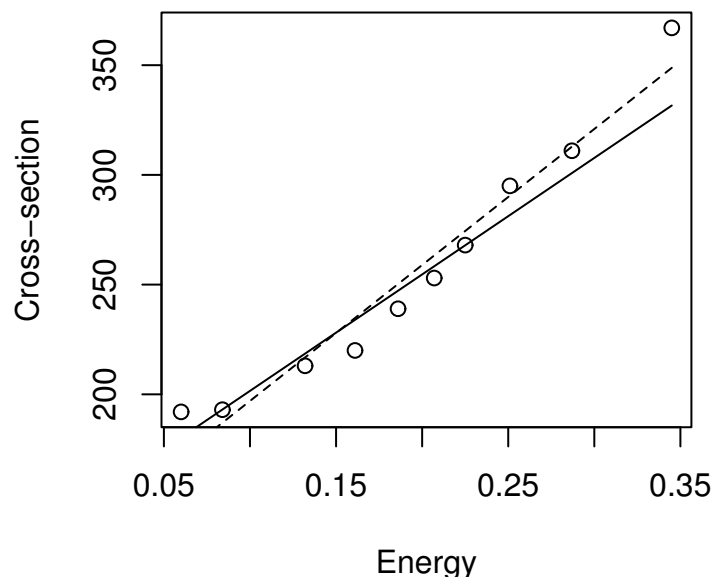```

and are shown in Figure 5.1.



Figure 5.1: Weighted least square fit shown in solid. Unweighted is dashed.

The unweighted fit appears to fit the data better overall but remember that for lower values of energy, the variance in the response is less and so the weighted fit tries to catch these points better than the others.

## 5.3 Iteratively Reweighted Least Squares

In cases, where the form of the variance of $\varepsilon$ is not completely known, we may model $\Sigma$ using a small number of parameters. For example,

$$\text{var } \varepsilon_i = \gamma_0 + \gamma_1 x_1$$

might seem reasonable in a given situation. The IRWLS fitting Algorithm is

1. Start with $w_i = 1$

2. Use least squares to estimate $\beta$.

3. Use the residuals to estimate $\gamma$, perhaps by regressing $\hat{\varepsilon}^2$ on $x$.

4. Recompute the weights and goto 2.

Continue until convergence. There are some concerns about this — how is subsequent inference about $\beta$ affected? Also how many degrees of freedom do we have? More details may be found in Carroll and Ruppert (1988).

An alternative approach is to model the variance and jointly estimate the regression and weighting parameters using likelihood based method. This can be implemented in R using the `gls()` function in the `nlme` library.