The General Linear Model (Part 2)
Lecture 11

- *Estimation in the general linear model.*
  - *Estimation in specific models*
  - *Time as class versus continuous*
  - *Properties of 'good' estimators*
  - *Standard errors and confidence intervals*

4 Estimation

4.1 Computing estimates – methods and application

4.1.1 Myostatin data in a one-way effects model

The Myostatin data came from a 2×3 factorial treatment structure in a CRD, and hence was analyzed with a 2-way ANOVA. We can also fit it using the one-way effects model, where there are 2×3 = 6 levels of this new 'composite' factor. This is considered next.

Write the vectors $\mathbf{Y}$ and $\boldsymbol{\beta}$ and matrix $\mathbf{X}$ associated with this data, in the one-way effects model:

$$\mathbf{Y} = (Y_{11} \;\; Y_{12} \;\; Y_{13} \;\; Y_{14} \;\; Y_{21} \ldots Y_{61} \;\; Y_{62} \;\; Y_{63} \;\; Y_{64})^t$$

$$\boldsymbol{\beta} = (\mu \;\; \kappa_1 \;\; \kappa_2 \;\; \kappa_3 \;\; \kappa_4 \;\; \kappa_5 \;\; \kappa_6)^t$$

$$\mathbf{X} = \begin{pmatrix} 1 & 1 & 0 & 0 & \ldots & 0 \\ 1 & 1 & 0 & 0 & \ldots & 0 \\ 1 & 1 & 0 & 0 & \ldots & 0 \\ 1 & 1 & 0 & 0 & \ldots & 0 \\ 1 & 0 & 1 & 0 & \ldots & 0 \\ 1 & 0 & 1 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & 0 & 0 & 0 & \ldots & 1 \\ 1 & 0 & 0 & 0 & \ldots & 1 \\ 1 & 0 & 0 & 0 & \ldots & 1 \\ 1 & 0 & 0 & 0 & \ldots & 1 \end{pmatrix}$$

Note that $\mathbf{X}$ does not have full rank (e.g., first column is the sum of the next 6). Thus, $(\mathbf{X}'\mathbf{X})^{-1}$ does not exist. We'll need to use a generalized inverse.

The fit of this model with the data application can be obtained as follows:

```
*One-way version;
proc glm data=myostatin; class group time; model y = group*time / solution; run;
```

```
The GLM Procedure
Dependent Variable: y

                                 Sum of
Source                     DF       Squares     Mean Square    F Value    Pr > F
Model                       5    23.12640221      4.62528044       8.02    0.0004
Error                      18    10.37454375      0.57636354
Corrected Total            23    33.50094596

      R-Square     Coeff Var      Root MSE        y Mean
      0.690321      14.04979      0.759186      5.403542

                                                Standard
Parameter                 Estimate                 Error    t Value    Pr > |t|
Intercept            3.777750000 B      0.37959305       9.95      <.0001
group*time c 24      3.184250000 B      0.53682564       5.93      <.0001
group*time c 48      1.673000000 B      0.53682564       3.12      0.0060
group*time c 72      1.133500000 B      0.53682564       2.11      0.0490
group*time m 24      2.290500000 B      0.53682564       4.27      0.0005
group*time m 48      1.473500000 B      0.53682564       2.74      0.0133
group*time m 72      0.000000000 B       .                .         .

NOTE: The X'X matrix has been found to be singular, and a generalized inverse was used to solve the
normal equations.  Terms whose estimates are followed by the letter 'B' are not uniquely estimable.
```

- A least squares solution to $\boldsymbol{\beta} = (\mu \ \kappa_1 \ \kappa_2 \ \kappa_3 \ \kappa_4 \ \kappa_5 \ \kappa_6)^t$ for the Myostatin data is given in the "Parameter estimates" of the previous SAS output. To solve for $\hat{\boldsymbol{\beta}}$, SAS uses a generalized inverse for $\mathbf{X}^t\mathbf{X}$ due to linear dependency ($\mathbf{X}$ and thus $\mathbf{X}^t\mathbf{X}$ do not have full rank).

- The fact that $\mathbf{X}$ is not of full rank is easy to see, since the first column of $\mathbf{X}$ is the sum of the other columns, i.e., the columns are not linearly independent. SAS uses the generalized inverse for $\mathbf{X}^t\mathbf{X}$ that is equivalent to setting the highest level of *group*×*time* to 0.

- Consequently, the $\hat{\boldsymbol{\beta}}$ solution is not unique (which relates to the NOTE at the end of the output). This is easy to see since we could have picked any other level to be the reference level, which would in turn alter the estimates. The 'NOTE' does not indicate an error; we just need to be careful about which functions of parameters to consider.

Finally, here is the R fit with the one-way data. Differences are again due to use of reference group; R uses ‘Control at 24 hours’, the lowest level of each factor:

```
#1-way effects model
class_fit2=lm(y ~ gt,data=myostatin)
summary(class_fit2)

Call:  lm(formula = y ~ gt, data = myostatin)

Residuals:
    Min      1Q  Median      3Q     Max
-0.8193 -0.5470 -0.1629  0.2788  1.5862

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   6.9620     0.3796  18.341 4.27e-13 ***
gtc48        -1.5113     0.5368  -2.815  0.01146 *
gtc72        -2.0508     0.5368  -3.820  0.00125 **
gtm24        -0.8938     0.5368  -1.665  0.11325
gtm48        -1.7108     0.5368  -3.187  0.00511 **
gtm72        -3.1843     0.5368  -5.932 1.30e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7592 on 18 degrees of freedom
Multiple R-squared: 0.6903,     Adjusted R-squared: 0.6043
F-statistic: 8.025 on 5 and 18 DF,  p-value: 0.0003960
```

4.1.2 Myostatin data in a two-way effects model

Now we get back to the analytical model that is more consistent with the actual treatment structure. The output on page 3 relates to the estimates that we will be deriving now.

<u>Model with interaction</u>

Write **X**, **Y** and **β** (for the two-way effects model).

The model: $Y_{ijk} = \mu + \alpha_i + \tau_j + \gamma_{ij} + \varepsilon_{ijk}$

$$\mathbf{X} = \begin{pmatrix} 1&1&0&1&0&0&1&0&0&0&0&0 \\ 1&1&0&1&0&0&1&0&0&0&0&0 \\ 1&1&0&1&0&0&1&0&0&0&0&0 \\ 1&1&0&1&0&0&1&0&0&0&0&0 \\ 1&1&0&0&1&0&0&1&0&0&0&0 \\ 1&1&0&0&1&0&0&1&0&0&0&0 \\ 1&1&0&0&1&0&0&1&0&0&0&0 \\ 1&1&0&0&1&0&0&1&0&0&0&0 \\ .&.&.& & & & & & & & & \\ 1&0&1&0&0&1&0&0&0&0&0&1 \\ 1&0&1&0&0&1&0&0&0&0&0&1 \\ 1&0&1&0&0&1&0&0&0&0&0&1 \\ 1&0&1&0&0&1&0&0&0&0&0&1 \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} y_{111} \\ y_{112} \\ y_{113} \\ y_{114} \\ y_{121} \\ y_{122} \\ y_{123} \\ y_{124} \\ \dots \\ y_{231} \\ y_{232} \\ y_{233} \\ y_{234} \end{pmatrix} = \begin{pmatrix} 6568 \\ 6802 \\ 7198 \\ 7280 \\ 4992 \\ 5242 \\ 5285 \\ 6284 \\ \dots \\ 3076 \\ 3209 \\ 3462 \\ 5364 \end{pmatrix} \quad \boldsymbol{\beta} = \begin{pmatrix} \mu \\ \alpha_1 \\ \alpha_2 \\ \tau_1 \\ \tau_2 \\ \tau_3 \\ \gamma_{11} \\ \gamma_{12} \\ \gamma_{13} \\ \gamma_{21} \\ \gamma_{22} \\ \gamma_{23} \end{pmatrix}$$

Once again, the **X** matrix does not have full rank (i.e., at least one column is a linear combination of the others), so we can't use a regular inverse of $\mathbf{X}^t\mathbf{X}$ in calculating Beta hat. We will let SAS compute the generalized inverse. Again, the way SAS computes the generalized inverse is equivalent to setting the highest levels of factors to 0 (including levels of interactions that involve the highest level of at least one of the factors). Here is a review of the two-way model fit from SAS PROC GLM:

```
proc glm data=myostatin; class group time; model y = group|time / solution; run;
```

Parameter		Estimate		Std. Error	t Value	Pr > |t|
Intercept		3.777750000	B	0.37959305	9.95	<.0001
group	c	1.133500000	B	0.53682564	2.11	0.0490
group	m	0.000000000	B	.	.	.
time	24	2.290500000	B	0.53682564	4.27	0.0005
time	48	1.473500000	B	0.53682564	2.74	0.0133
time	72	0.000000000	B	.	.	.
group*time	c 24	-0.239750000	B	0.75918610	-0.32	0.7558
group*time	c 48	-0.934000000	B	0.75918610	-1.23	0.2344
group*time	c 72	0.000000000	B	.	.	.
group*time	m 24	0.000000000	B	.	.	.
group*time	m 48	0.000000000	B	.	.	.
group*time	m 72	0.000000000	B	.	.	.

```
NOTE: The X'X matrix has been found to be singular...
```

SAS's generalized inverse is only one possibility; choosing another one will yield different Beta estimates.

An alternative approach is to define a full-rank model beforehand. To do this, we can employ SAS PROC REG, creating our own indicator variables. We can also use the LM function in R, as shown below. In this approach, I have not used the 'factor' function as shown previously, but rather, I have created my own indicators so that the results match those of SAS (in which the highest levels of factors are set to 0).

Fitting the model in R – in this case instead of using the factor function as before, I manually create dummy variables so that the estimates are the same as those obtained from SAS.

```
leucine =c(6568,6802,7198,7280,4992,5242,5285,6284,4092,4331,5135,6087,5516,6023,
   6334,6400,4512,4706,5175,6612,3076,3209,3462,5364)/1000;
group=c(1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0)
time=c(24,24,24,24,48,48,48,48,72,72,72,72,24,24,24,24,48,48,48,48,72,72,72,72)
time_24=c(1,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0)
time_48=c(0,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,0,0,0,0)
class_fit=lm(leucine ~ group + time_24 + time_48 + group*time_24 + group*time_48)
summary(class_fit)

Coefficients:
```

```
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     3.7778     0.3796   9.952 9.62e-09 ***
group           1.1335     0.5368   2.111 0.048975 *
time_24         2.2905     0.5368   4.267 0.000464 ***
time_48         1.4735     0.5368   2.745 0.013318 *
group:time_24  -0.2397     0.7592  -0.316 0.755788
group:time_48  -0.9340     0.7592  -1.230 0.234435
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7592 on 18 degrees of freedom
Multiple R-squared: 0.6903,     Adjusted R-squared: 0.6043
F-statistic: 8.025 on 5 and 18 DF,  p-value: 0.0003960
```

The beta estimates are the same as with the ‘less-than-full-rank’ approach since we used the highest levels of factors as reference points.

Summary:

- In order to fit a two-way effects model, we need to deal with linear dependencies in **X**. Another approach is to let software do this automatically by using generalized inverses in the calculation. However, in the end, the fitted model will be the same when we consider functions of parameters that are uniquely estimable.

- In some cases, the two approaches will even yield the same individual beta estimates. For example, using the highest levels of factors as the reference group is equivalent to the way that SAS finds a generalized inverse.

4.1.3 Myostatin data in the means model

If we simply remove the intercept from the one-way effects model, we have the means model. In SAS or R, you can remove the intercept easily. Below is the means model fit in SAS and R, plus some extra tests in R. Here, results are the same since the model already has full rank.

SAS code

```
proc glm
data=myostatin;
class group time;
model y = group*time
   / solution noint;
run;
```

SAS output

```
Parameter          Estimate  Std. Err. t Value Pr>|t|
group*time c 24       6.962   0.380    18.34    <.0001
group*time c 48       5.451   0.380    14.36    <.0001
group*time c 72       4.911   0.380    12.94    <.0001
group*time m 24       6.068   0.380    15.99    <.0001
group*time m 48       5.251   0.380    13.83    <.0001
group*time m 72       3.778   0.380     9.95    <.0001
```

R code

```
#means model
glm2<-glm(y ~ gt-1,
data=myostatin)
summary(glm2)
```

R output

```
      Estimate Std. Error t value Pr(>|t|)
gtc24   6.9620     0.3796  18.341 4.27e-13 ***
gtc48   5.4507     0.3796  14.359 2.67e-11 ***
gtc72   4.9112     0.3796  12.938 1.49e-10 ***
gtm24   6.0682     0.3796  15.986 4.42e-12 ***
gtm48   5.2513     0.3796  13.834 4.95e-11 ***
gtm72   3.7778     0.3796   9.952 9.62e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

4.1.4 Time as a categorical variable versus time as a continuous variable

- So far we've considered modeling time as a class or categorical variable, which allows for separate estimates at each time point. In some cases we may want to model time as a continuous variable, which imposes more constraints in the estimates.

- For example, if we allow for a straight line fit for the outcome versus time, the difference between estimates from 24 to 48 hours is necessarily the same as the difference between estimates from 48 to 72 hours. For estimates using time as a class variable, each estimate is not constrained by estimates for other time points. There may also be higher-order polynomial functions that we could use in modeling time as a continuous variable, but for now we'll just consider the straight line relationship.

- If the linearity assumption holds, there are several potential advantages to modeling time as a continuous variable despite the fact that estimates are more constrained:

  - The relationship can be expressed in a very simple, intuitive way with the slope, which expresses a change in the outcome per unit increase in x.

  - Fewer degrees of freedom are spent on the model, saving more for the error term (for the Myostatin application, time has 1 d.f. instead of (3–1)=2 with the class variable approach.

  - Estimates for values of *x* not observed can be easily obtained (e.g., at 36 hours for the Myostatin application). For application with many time points, using time as a continuous variable may be the only true alternative, since using time as a class variable in those cases may require too many d.f. for the model.

- In SAS PROC GLM, time can be modeled as a continuous variable by simply leaving the *Time* variable out of the CLASS statement.
  - Considering the Myostatin application, *Group* and *Group***Time* and *Time* were all predictors in the model.
  - This allowed for separate estimates for each group-time combination when *Time* was treated as a class variable.
  - When modeling *Time* as a metric variable, separate regression lines can be obtained for each group.
  - The coefficient for *Group* indicates differences between the 2 groups at the y-intercept, and the coefficient of *Group***Time* indicates differences in slopes between the 2 groups.
- When modeling time as a class variable, the linearity assumption can be checked informally by inspecting the plot of outcome versus predictor to see if the patterns look linear, or add higher order terms and see if they are significant.
- In SAS PROC REG or the lm() function in R, variables are treated as continuous variables by default and there are no CLASS statements or options. With R, you can use as.factor(Time) to force lm() to treat the variable Time as categorical; this will use the lowest level as reference by default. The relevel(…,ref=) function allows you to specify an alternative reference group.

Time as continuous for the Myostatin data. For simplicity, we can convert hours to days. Sketch the model, matrix **X**, and vector **β** for this case. For both models, let *i* denote group, *j* denote time, *k* denote replicate. The less-than-full-rank model is given below. Note that the time variable x_j is only one component of **X**.

$$Y_{ijk} = \mu + \alpha_i + \beta x_j + \gamma_i x_j + \varepsilon_{ijk}$$

where x_j are the times in hours; i=1,2; j=1,2,3; k=1,2,3,4.

$$\boldsymbol{\beta} = \begin{pmatrix} \mu \\ \alpha_1 \\ \alpha_2 \\ \beta \\ \gamma_1 \\ \gamma_2 \end{pmatrix}$$

$$\mathbf{X} = \begin{pmatrix}
1 & 1 & 0 & 24 & 24 & 0 \\
1 & 1 & 0 & 24 & 24 & 0 \\
1 & 1 & 0 & 24 & 24 & 0 \\
1 & 1 & 0 & 24 & 24 & 0 \\
1 & 1 & 0 & 48 & 48 & 0 \\
1 & 1 & 0 & 48 & 48 & 0 \\
1 & 1 & 0 & 48 & 48 & 0 \\
1 & 1 & 0 & 48 & 48 & 0 \\
1 & 1 & 0 & 72 & 72 & 0 \\
1 & 1 & 0 & 72 & 72 & 0 \\
1 & 1 & 0 & 72 & 72 & 0 \\
1 & 1 & 0 & 72 & 72 & 0 \\
1 & 0 & 1 & 24 & 0 & 24 \\
1 & 0 & 1 & 24 & 0 & 24 \\
1 & 0 & 1 & 24 & 0 & 24 \\
1 & 0 & 1 & 24 & 0 & 24 \\
1 & 0 & 1 & 48 & 0 & 48 \\
1 & 0 & 1 & 48 & 0 & 48 \\
1 & 0 & 1 & 48 & 0 & 48 \\
1 & 0 & 1 & 48 & 0 & 48 \\
1 & 0 & 1 & 72 & 0 & 72 \\
1 & 0 & 1 & 72 & 0 & 72 \\
1 & 0 & 1 & 72 & 0 & 72 \\
1 & 0 & 1 & 72 & 0 & 72
\end{pmatrix}$$

PROC GLM code and partial output:

```
*time as continuous variable;
proc glm data=myostatin; class group; model y = group|time / solution xpx; run;
```

```
The GLM Procedure

Dependent Variable: y
                                   Sum of
Source                     DF       Squares     Mean Square    F Value    Pr > F
Model                       3    22.20954200      7.40318067      13.11    <.0001
Error                      20    11.29140396      0.56457020
Corrected Total            23    33.50094596

R-Square     Coeff Var      Root MSE        y Mean
0.662953      13.90530      0.751379      5.403542

                                        Standard
Parameter            Estimate              Error    t Value    Pr > |t|
Intercept         7.322916667 B      0.57387509      12.76      <.0001
group      c      0.502500000 B      0.81158193       0.62      0.5428
group      m      0.000000000 B       .                .         .
time              -0.047718750 B      0.01106886      -4.31      0.0003
time*group c      0.004994792 B      0.01565373       0.32      0.7530
time*group m      0.000000000 B       .                .         .

NOTE: The X'X matrix has been found to be singular...
```

- If $\mathbf{X}^t\mathbf{X}$ is singular, then $\tilde{\boldsymbol{\beta}}$ is not unique, but $\mathbf{X}\tilde{\boldsymbol{\beta}}$ is. Also, there may be functions of parameters in $\boldsymbol{\beta}$ that are uniquely estimable despite the fact that $\tilde{\boldsymbol{\beta}}$ is not unique.

- Modeling time as a class versus continuous variable is an important issue that we will discuss throughout the course.

  - Time as a class variable…
    - offers the most flexibility
    - no parametric constraints imposed across levels of time
    - uses more degrees of freedom in the model (e.g., with 4 times there are 3 d.f., 1 d.f. if you have a simple linear term for time as continuous).
    - recommended when there are relatively few times (say, five or less), for which tests for polynomial trends can still be conducted (see course notes for details).

  - Time as continuous recommended when…
    - there are many times of observation, possibly unequally spaced
    - there are different times of measurement for subjects
    - interpolating estimates and predicted values may be of interest

- Here is the fit with time as continuous using R software. Note that the 'myostatin' data is the same as presented in Section 2:

R Code:

```
#Model using time as continuous
contin_fit=lm(y ~ group + time + group*time,data=myostatin)
summary(contin_fit)
```

R Output summary:

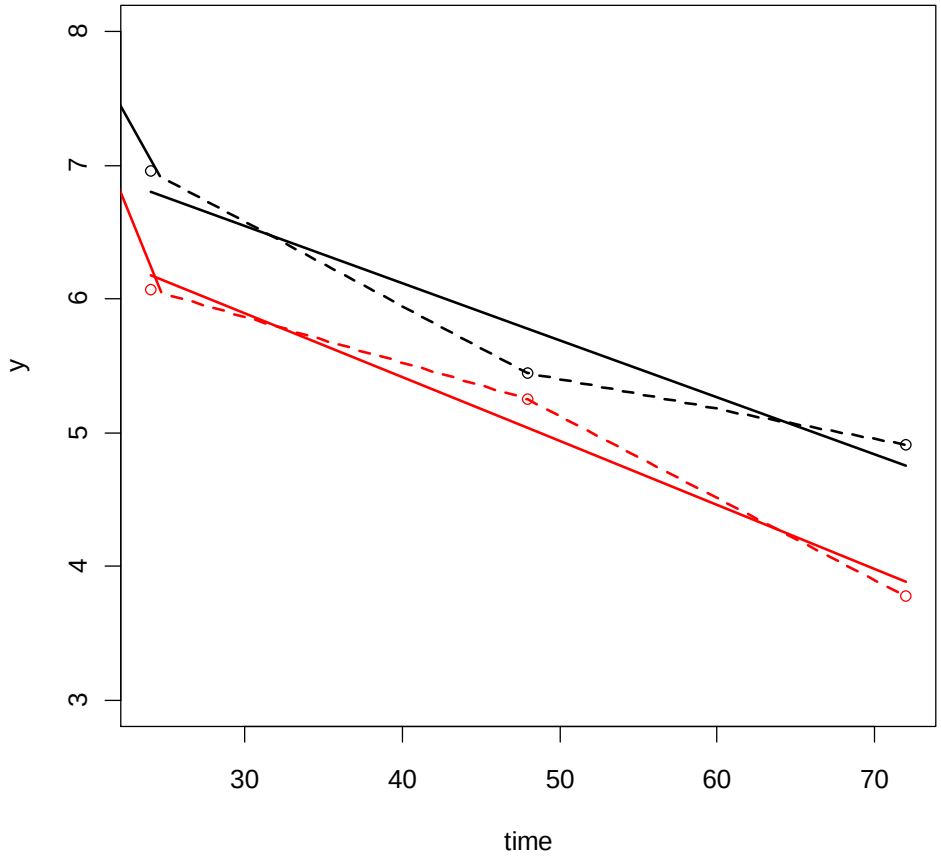
```
Call:
lm(formula = y ~ group + time + group * time, data = myostatin)

Residuals:
    Min      1Q  Median      3Q     Max
-0.8112 -0.5235 -0.1934  0.3888  1.5796

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.825417   0.573875  13.636 1.38e-11 ***
groupm      -0.502500   0.811582  -0.619 0.542799
time        -0.042724   0.011069  -3.860 0.000976 ***
groupm:time -0.004995   0.015654  -0.319 0.752974
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7514 on 20 degrees of freedom
```

```
Multiple R-squared: 0.663,      Adjusted R-squared: 0.6124
F-statistic: 13.11 on 3 and 20 DF,  p-value: 5.829e-05
```

- You may notice that the signs on some of the estimates are negative, whereas with the SAS analysis they were positive. This is because in SAS, the highest level of Group (Myostatin) was set as the reference group, while with R, the lowest level of Group (Control) is set as the reference group. Make sure to remember these key differences!

- If we only include the linear term for time, then we are forcing straight-line relationships between time and y; including the interaction term allows for different slopes, while inclusion of the group term allows for different y-intercepts.

Predicted values for the Myostatin data using time as continuous (solid), and time as class (dashed, circles).



4.2 Properties of $\hat{\boldsymbol{\beta}}$, $\hat{\sigma}^2$, and linear functions of $\hat{\boldsymbol{\beta}}$

4.2.1 General review of 'good' estimators: MLE, UMVU, BLU, BQU

- Informal definitions:
  - **MLE** – maximum likelihood estimator. Values of the parameters (in terms of statistics) that maximize the likelihood function. The likelihood function is derived from the density of **Y**.
  - **UMVU** – uniformly minimum variance unbiased estimator. A statistic of **Y** that has the smallest variance in estimating a parameter within the class of unbiased estimators for that parameter. The density of **Y** must be known to derive UMVU estimators.
  - **BLU** – best linearly unbiased estimator. A statistic that has the smallest variance in estimating a parameter that is a linear function of the data **Y**. (This and BQU were developed so that 'good' estimators could be determined for the case where the pdf of **Y** is not known.)
  - **BQU** – best quadratic unbiased estimator. A statistic that has the smallest variance in estimating a parameter that is a quadratic function of the data **Y**.

4.2.2 Properties of estimators in the GLM

The $\hat{\sigma}^2$ discussed below is adjusted so that it is unbiased, i.e., (*n*–*k*) used in the denominator instead of *n*.

- For Case I (*iid* normal errors), $\hat{\boldsymbol{\beta}}$ and $\hat{\sigma}^2$ are both MLE and UMVU estimators of $\boldsymbol{\beta}$ and σ^2, respectively.

- For Case II, unknown but identical error distributions, $\hat{\boldsymbol{\beta}}$ is the BLU of $\boldsymbol{\beta}$ and $\hat{\sigma}^2$ is the BQU estimator of σ^2.

4.2.3 Maximum likelihood estimators in the GLM

- The least squares estimator $\hat{\boldsymbol{\beta}}$ is also the maximum likelihood estimator. This is shown easily by taking the partial derivative of log likelihood function with respect to $\boldsymbol{\beta}$, setting it to 0, which yields the normal equations.

- To get the MLE of σ^2, we first substitute $\hat{\boldsymbol{\beta}}$ in for $\boldsymbol{\beta}$ in the log-likelihood function, yielding a profile likelihood function that now only involves the variance parameter. The MLE is obtained by setting to 0 the partial derivative of this quantity with respect to σ^2. The solution is

$$\tilde{\sigma}^2 = (1/n)\mathbf{Y}^t(\mathbf{I}-\mathbf{P}_\mathbf{X})\mathbf{Y}.$$

- We can adjust this estimator so that it is unbiased:

$$\hat{\sigma}^2 = [1/(n-k)]\mathbf{Y}^t(\mathbf{I}-\mathbf{P}_\mathbf{X})\mathbf{Y},$$

where $k=r(\mathbf{X})$. (If $\mathbf{X}$ has full rank, then $k=p$.)

[NOTE: The use of '~' and '^' on the variance estimator is to denote the biased and unbiased estimators, respectively, not to indicate non-unique and unique estimators, as was done for Beta.]

- Note that both variance estimators above are quadratic forms

- Note that $\mathbf{I}-\mathbf{P}_\mathbf{X}$ is symmetric and invariant to choice of $(\mathbf{X}^t\mathbf{X})^-$, since $\mathbf{P}_\mathbf{X}$ also has these qualities. In addition, it is easy to show that $\mathbf{I}-\mathbf{P}_\mathbf{X}$ is idempotent:

$$(\mathbf{I}-\mathbf{P}_\mathbf{X})(\mathbf{I}-\mathbf{P}_\mathbf{X}) = \mathbf{I}-\mathbf{P}_\mathbf{X}-\mathbf{P}_\mathbf{X}+\mathbf{P}_\mathbf{X}\mathbf{P}_\mathbf{X} = \mathbf{I}-\mathbf{P}_\mathbf{X}-\mathbf{P}_\mathbf{X}+\mathbf{P}_\mathbf{X} = \mathbf{I}-\mathbf{P}_\mathbf{X}\,.$$

- It follows that $\mathbf{P}_\mathbf{X}\mathbf{P}_\mathbf{X} = \mathbf{P}_\mathbf{X}$ since $\mathbf{P}_\mathbf{X}$ is in the column space of $\mathbf{X}$, as each column of $\mathbf{P}_\mathbf{X}$ is a linear combination of columns of $\mathbf{X}$

- Using the result for quadratic forms in Section 4, we can show that

$$(n-k)\hat{\sigma}^2 / \sigma^2 \sim \chi^2_{n-k}\ ,$$

I.e., the quantity on the left has a central chi-square distribution with *n–k* degrees of freedom.

4.3 Standard errors and confidence intervals

- Let **C** be a matrix with rows $\mathbf{c}_i^t$, i=1,…,q. We typically denote one $\mathbf{c}_i^t$ as **L**. Concerning the functions of parameters $\boldsymbol{\theta} = \mathbf{C}\boldsymbol{\beta} - \mathbf{h}$, we may be interested in developing confidence intervals. There are two basic types:

- <u>One-at-a-time confidence intervals</u>. This means that each θ_i is treated individually, and a (1–α) CI is determined separately for each $\theta_i = \mathbf{c}_i^t\boldsymbol{\beta}$ (where $\mathbf{c}_i^t$ is the i^{th} row of **C**):

$$\mathbf{c}_i^t\hat{\boldsymbol{\beta}} \mp t_{a/2,n-k}\sqrt{\hat{Var}[\mathbf{c}_i^t\hat{\boldsymbol{\beta}}]}, \qquad i=1,\ldots,q$$

- <u>Simultaneous confidence intervals</u>. This means that all of the θ_i are treated simultaneously, and CIs are determined for each θ_i such that the probability is equal to 1–α that the q intervals simultaneously cover their respective θ_i.

- <u>Standard errors of Beta estimates</u>: For $\mathbf{L\beta}$ that is estimable, previous theoretical results imply that $SE(\mathbf{L}\hat{\boldsymbol{\beta}}) = \sqrt{\mathbf{L}(\mathbf{X}^t\mathbf{X})^{-}\mathbf{L}^t\sigma^2}$. Since σ^2 is typically unknown, the quantity is usually estimated with $\hat{SE}(\mathbf{L}\hat{\boldsymbol{\beta}}) = \sqrt{s^2\mathbf{L}(\mathbf{X}^t\mathbf{X})^{-}\mathbf{L}^t}$, where $s^2 = \left[\mathbf{Y}^t(\mathbf{I}-\mathbf{P}_X)\mathbf{Y}\right] / (n-k)$, the MSE.