STAT 571A — Advanced Statistical Regression Analysis

Chapter 6 NOTES

Multiple Regression – I

Multiple Linear Regression Model

- If p–1 > 1 predictor variables are under study, we expand the SLR model into a ("first-order") **Multiple Linear Regression (MLR)** model:

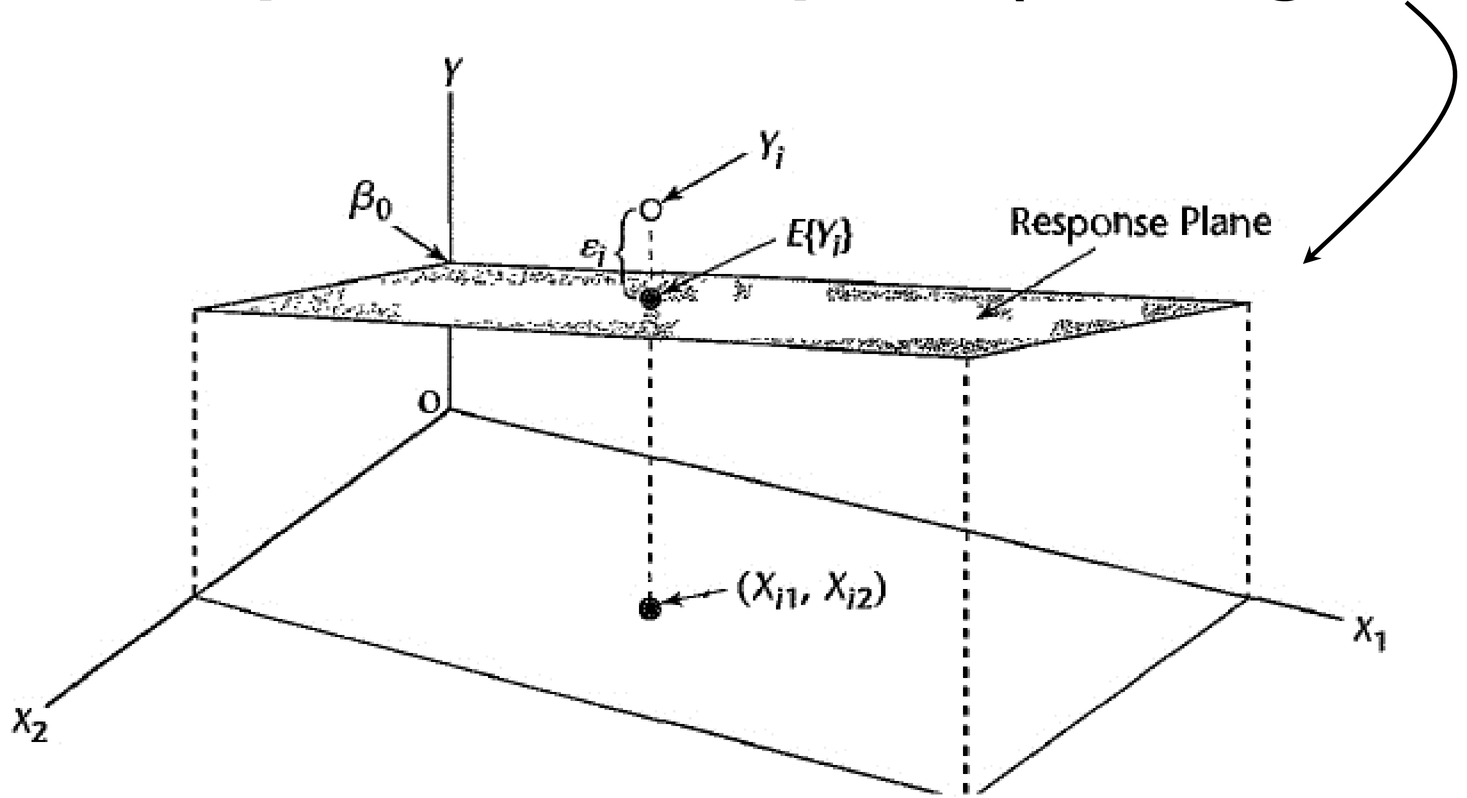
  $$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_{p-1} X_{i,p-1} + \varepsilon_i$$

  where $\varepsilon_i \sim$ i.i.d.$N(0,\sigma^2)$; $i = 1, \ldots, n$.

- One can also write this as $Y_i = \beta_0 X_{i0} + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_{p-1} X_{i,p-1} + \varepsilon_i$, where $X_{i0} \equiv 1$.

MLR with p=3

- For instance take the case of p = 3 (two predictors): $E\{Y_i\} = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2}$
- This is a plane in 3-D space (see Fig. 6.1).



MLR with p=3 (cont'd)

- Two predictors: $E\{Y_i\} = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2}$
- Interpret the β_ks as:
  - β_1 is change in E{Y} for unit change in X_1, when all other X_k's (here, just X_2) are held fixed.
  - β_2 is change in E{Y} for unit change in X_2, when all other X_k's (here, just X_1) are held fixed.
  - β_0 is the "Y-intercept," as before.

Special MLR Models

- If one (or more) of the X_k's is an indicator (=0 or =1), E{Y} has a simplified interpreta-tion. See equ. (6.10).
- If $X_k = X^k$, this is a **polynomial regression** (discussed in §8.1).
- Say X_1 and X_2 interact in how they affect E{Y}. Then we include a second-order **interaction** term:

$$E\{Y_i\} = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \beta_3 X_{i1} X_{i2}$$

Response Surface Model

- **We combine the second-order polynomial with second-order interactions to create a response surface model:**

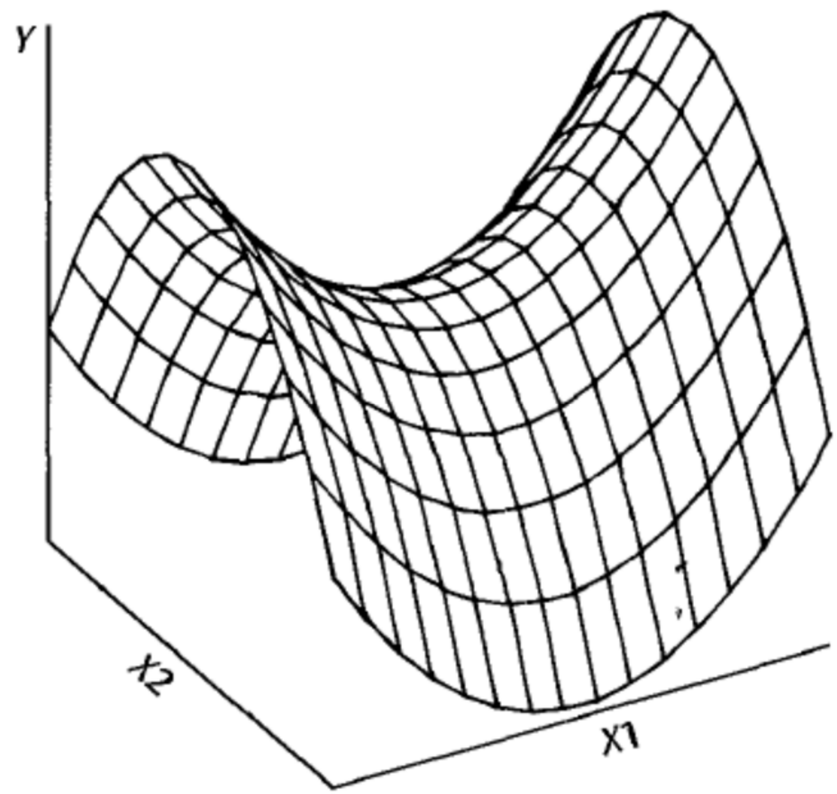
$$E\{Y_i\} = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i1}^2 + \beta_3 X_{i2} + \beta_4 X_{i2}^2 + \beta_5 X_{i1} X_{i2}$$

- **(Why is this "linear"? Because all the β_k's enter into E{Y} at first-order!)**

Response Surface (cont'd)

- The second-order response surface model produces a smoothly arcing surface in 3-D space.

- See Fig. 6.2.



§6.2: MLR Matrix Formulation

The MLR model (with any $p - 1 > 1$) is a straightforward extension of SLR, so the matrix equations are of essentially identical form.

Recall $$\mathbf{Y}_{n\times 1} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix},\quad E\{\mathbf{Y}\} = \begin{bmatrix} E[Y_1] \\ E[Y_2] \\ \vdots \\ E[Y_n] \end{bmatrix},\quad \boldsymbol{\epsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

and now take $$\mathbf{X}_{n\times p} = \begin{bmatrix} 1 & X_{11} & X_{12} & \cdots & X_{1,p-1} \\ 1 & X_{21} & X_{22} & \cdots & X_{2,p-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & X_{n1} & X_{n2} & \cdots & X_{n,p-1} \end{bmatrix}$$

Matrix Formulation (cont'd)

If we let $\beta_{p\times 1} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{p-1} \end{bmatrix}$, we can write

the MLR model as a matrix expression:

$$Y_{n\times 1} = X_{n\times p}\beta_{p\times 1} + \epsilon_{n\times 1}$$

where $\epsilon \sim N_n(0, \sigma^2 I)$.

The mean response vector is $E\{Y\} = X\beta$ and the covariance matrix is $\sigma^2\{Y\} = \sigma^2 I$.

§6.3: LS Estimation

The LS normal equations can again be written simply as

$$(\mathbf{X}'\mathbf{X})\mathbf{b} = \mathbf{X}'\mathbf{Y},$$

for $\mathbf{b} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{p-1} \end{bmatrix}$. The solution for $\mathbf{b}$ is clearly

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$$

whenever $(\mathbf{X}'\mathbf{X})^{-1}$ exists.

(Here again, these correspond to the MLE for β.)

§6.4: Fitted Values

The fitted values for the MLR are $\hat{\mathbf{Y}} = [\hat{Y}_1 \ldots \hat{Y}_n]'$, which in matrix notation is again

$$\hat{\mathbf{Y}} = \mathbf{X}\mathbf{b} = \mathbf{H}\mathbf{Y}.$$

The hat matrix remains $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$.

Also, the residual vector is still

$$\mathbf{e} = \mathbf{Y} - \hat{\mathbf{Y}} = \mathbf{Y} - \mathbf{H}\mathbf{Y} = (\mathbf{I} - \mathbf{H})\mathbf{Y},$$

with $\boldsymbol{\sigma}^2\{\mathbf{e}\} = (\mathbf{I} - \mathbf{H})\sigma^2$ estimated via

$$\mathbf{s}^2\{\mathbf{e}\} = (\mathbf{I} - \mathbf{H})\times\text{MSE}.$$

§6.5: MLR ANOVA

The MLR ANOVA table also looks similar to its SLR counterpart:

Source	d.f.	SS	MS
Regr.	p–1	SSR=$\mathbf{Y}'(\mathbf{H}-\frac{1}{n}\mathbf{J})\mathbf{Y}$	MSR=$\frac{SSR}{p-1}$
Error	n–p	SSE=$\mathbf{Y}'(\mathbf{I}-\mathbf{H})\mathbf{Y}$	MSE=$\frac{SSE}{n-p}$
Total	n–1	SSTO=$\mathbf{Y}'(\mathbf{I}-\frac{1}{n}\mathbf{J})\mathbf{Y}$	

The expected means squares are E{MSE} = σ^2 and E{MSR} = $\sigma^2 + \theta^2(\beta)$ (see next slide).

E{MSR}

The expected mean square for MSR involves the expression $\theta^2(\beta)$, which is a complicated function of β such that $\theta^2(\mathbf{0}) = 0$.

For instance, at p=3:

$$\theta^2(\beta) = \tfrac{1}{2}\Big\{\sum\sum \beta_k^{\,2}(X_{ik} - \bar{X}_k)^2$$

$$+ 2\beta_1\beta_2 \sum (X_{i1} - \bar{X}_1)(X_{i2} - \bar{X}_2)\Big\}$$

This suggests that an F-test is available for testing H_o: $\beta_1 = \beta_2 = \cdots = \beta_{p-1} = 0$ $\longrightarrow$

Full MLR F-Test

- To test H_o: $\beta_1 = \beta_2 = \ldots = \beta_{p-1} = 0$ vs. H_a: any departure, construct the "full" F-statistic $F^* = MSR/MSE$ and reject H_o when $F^* > F(1-\alpha;\ p-1, n-p)$.
- (At $p=2$ we clearly recover the SLR F-test of H_o: $\beta_1=0$.)
- The P-value is $P = P[F(p-1,n-p) > F^*]$.

Multiple R^2

For the MLR model, the **Coefficient of Multiple Determination** mimics its SLR progenitor:

$$R^2 = 1 - \frac{SSE}{SSTO} = \frac{SSR}{SSTO}$$

Again, $0 \le R^2 \le 1$.

Interpretation: R^2 is still the % of total variation in the Y_i's explained by the X_j's in the regression model.

Adjusted R^2

- With multiple X_k's, however, R^2 exhibits irregularities.
- Notice that by adding a new X_k to the model, SSE cannot increase. Thus we can drive $R^2 \rightarrow 1$ simply by pushing $p \rightarrow n$.
- An **adjusted R^2** compensates by replacing the SS terms with MS terms:

$$R_a^2 = 1 - \{MSE/MSTO\}.$$

- Interpretation is essentially similar.

§6.6: MLR Inferences

The LS estimator $\mathbf{b}$ is again unbiased:

$$E\{\mathbf{b}\} = \boldsymbol{\beta}.$$

Its sample covariance mtx. is again

$s^2\{\mathbf{b}\} = MSE(\mathbf{X}'\mathbf{X})^{-1}$. Take a closer look:

$$s^2\{\mathbf{b}\} = \begin{bmatrix} s^2\{b_0\} & s\{b_0,b_1\} & \cdots & s\{b_0,b_{p-1}\} \\ s\{b_1,b_0\} & s^2\{b_1\} & \cdots & s\{b_1,b_{p-1}\} \\ \vdots & \vdots & \vdots & \vdots \\ s\{b_{p-1},b_0\} & s\{b_{p-1},b_1\} & \cdots & s^2\{b_{p-1}\} \end{bmatrix}$$

The covar. of b_0 and b_{p-1}

The var. of b_{p-1}

MLR inferences (cont'd)

So, each indiv. b_k has

$$T_k = \frac{b_k - \beta_k}{s\{b_k\}} \sim t(n-p)$$

($k = 0,\ldots,p-1$). From this, a (pointwise) $1-\alpha$ conf. int. on β_k has the familiar form

$$b_k \pm t(1-\tfrac{\alpha}{2};\, n-p)s\{b_k\}.$$

Or, to test H_o: $\beta_k = 0$ vs. H_a: $\beta_k \neq 0$ find
$t^* = b_k/s\{b_k\}$ & reject H_o when $|t^*| > t(1-\frac{\alpha}{2};\, n-p)$.

(One-sided tests are similar.)

Bonferroni Adjustment

- But, WATCH IT! The t-based conf. int's and hypoth. tests are pointwise. If multiple b_k's are assessed, need a multiplicity adjustment.

- For instance, Bonferroni-adjusted simul-taneous limits on g>1 different β_k's are

$$b_k \pm B\, s\{b_k\}$$

for $B = t(1 - \tfrac{1}{2}\{\alpha/g\}; n-p)$ and $k = 1,\ldots,g$.

§6.7: Inference on $E\{Y_h\}$

Given a future predictor vector $\mathbf{X}_h = \begin{bmatrix} 1 \\ X_{h1} \\ \vdots \\ X_{h,p-1} \end{bmatrix}$,

an estimate of $E\{Y_h\}$ at this $\mathbf{X}_h$ is $\hat{Y}_h = \mathbf{X}_h'\mathbf{b}$.

We find $E\{\hat{Y}_h\} = \mathbf{X}_h'\boldsymbol{\beta}$ (unbiased!) with std. error $s\{\hat{Y}_h\} = \sqrt{MSE\left(\mathbf{X}_h'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}_h\right)}$.

A $1-\alpha$ conf. int. on $E\{Y_h\}$ then has the familiar form

$$\hat{Y}_h \pm t(1-\tfrac{\alpha}{2};\, n-p)s\{\hat{Y}_h\}.$$

Multiplicity Adjustment

Here again these are <u>pointwise</u> conf. int's. If more than a single X_h is under study, must apply a multiplicity adjustment.

Over a finite, pre-specified set of g > 1 X_h's, use the Bonferroni-adjusted intervals

$$\hat{Y}_h \pm B\, s\{\hat{Y}_h\}$$

where $B = t(1 - \frac{1}{2}\{\alpha/g\};\ n-p)$ and h = 1,...,g.

Multiplicity Adjustment (cont'd)

Or, for an exact, simultaneous 1–α confidence (hyper-)band on E{Y} over <u>all</u> <u>possible</u> vectors $\mathbf{X}_h$, use the WHS method:

$$\hat{Y}_h \pm W s\{\hat{Y}_h\}$$

for $W^2 = p F(1-\alpha;\ p, n-p)$.

WHS also applies (conservatively) for any g>1 $\mathbf{X}_h$'s, so always check: if W ≤ B, use the WHS limits instead of Bonferroni.

(Can also use the WHS band, and <u>only</u> the WHS band, for *post hoc* intervals on E{Y_h}.)

MLR Prediction

For <u>prediction</u> of a future observation $Y_{h(new)}$ at some $\mathbf{X}_{h(new)}$, use

$$\hat{Y}_{h(new)} = \mathbf{X}'_{h(new)}\mathbf{b},$$

with

$$s\{pred\} = \sqrt{MSE\left(1 + \mathbf{X}'_{h(new)}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}_{h(new)}\right)}\ .$$

The corresp. (pointwise) 1–α **prediction interval** is then

$$\hat{Y}_{h(new)} \pm t(1-\tfrac{\alpha}{2};\ n-p)s\{pred\}.$$

S-Method Prediction Intervals

As previously, the pointwise prediction interval is valid at only one $X_{h(new)}$. For simultaneous prediction intervals at g>1 future $X_{h(new)}$'s, apply a modification of the WHS method due to Scheffé (called the "S-method"):

$$\hat{Y}_{h(new)} \pm S\, s\{pred\}$$

where $S = \sqrt{g\, F(1-\alpha;\, g, n-p)}$,

for h = 1,...,g.

Bonferroni Prediction Intervals

Can also use Bonferroni intervals for multiplicity-adjusted predictions:

$$\hat{Y}_{h(new)} \pm B\, s\{pred\}$$

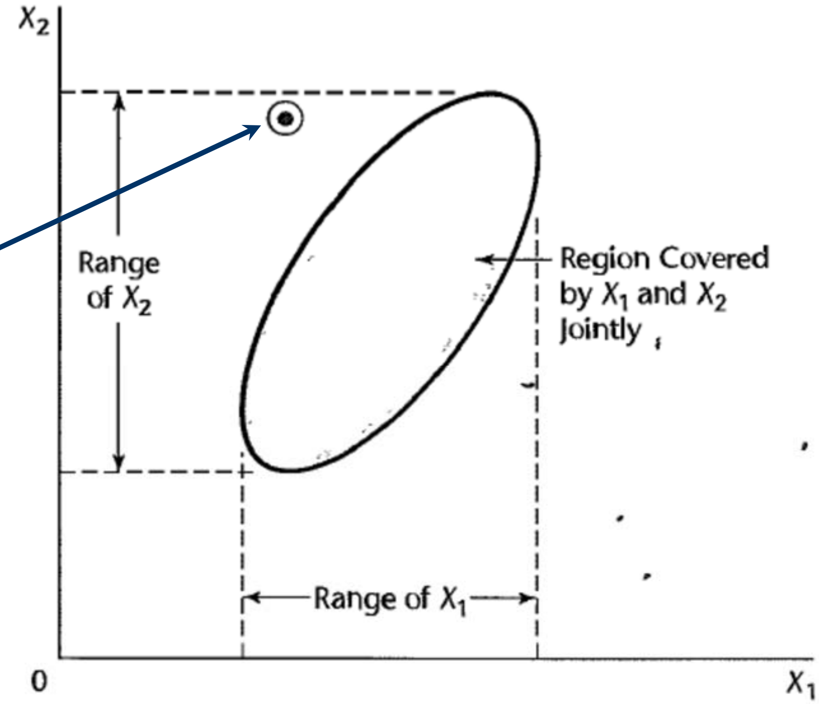
with $B = t(1 - \frac{1}{2}\{\alpha/g\};\, n-p)$ over $h = 1,\ldots,g$.

Both the Scheffé and Bonferroni crit. points are conservative for any finite g, so check first: if $S \le B$, use Scheffé's *S*-method, otherwise use Bonferroni.

Extrapolation

- As always, be careful with extrapolated $\mathbf{X}_h$ vectors.
- Ensure that the entire vector is within the range of the data.
- See Fig. 6.3 at p=3:

This point extrapolates, but that's not clear with-out a careful look at the data.



§6.8: Diagnostics

Preliminary diagnostics to assess an MLR fit include:

- **Quick check of pairwise correlations among Y and each/all X_k's: make sure no surprises are hiding (also see 'multicollinearity' discussion in §7.6).**
- **Should always plot the data! Try plotting Y vs. each X_k. Use a scatterplot matrix (see Fig. 6.4).**
- **Also try 3-D scatterplots of Y vs. pairs of X_k's. If available, apply real-time rotation. (In R, use `plot3d()` function from external *rgl* package.)**

MLR Residual Plots

Residual plots remain a mainstay:

- **plot e_i vs. $\hat{Y}_i$ (the usual resid. plot)**
- **plot $e_i = Y_i - \hat{Y}_i$ vs. X_{ik} at every $k = 1,\ldots,p-1$**
- **graph histograms/boxplots of the e_i's**
- **check NP plot of the e_i's**

(Same interpretations apply as in the SLR case.)

Brown-Forsythe Test

The Brown-Forsythe test for constant σ^2 remains valid with the MLR model:

- **(a)** Divide e_i's into two groups: group 1 has e_i's from small fitted values, $\hat{Y}_i$,
- **(b)** and group 2 has e_i's from large fitted values, $\hat{Y}_i$.
- **(c)** Then construct the t^*_{BF}-statistic as in §3.6. Conclude significant departure from homogeneous variance if

$$|t^*_{BF}| > t(1 - \tfrac{\alpha}{2};\ n-2).$$

- **(d)** P-value is $2P[\ t(n-2) > |t^*_{BF}|\ ]$.

Other Diagnostics/Remediation

- To test for Lack of Fit (LOF), can apply F-tests similar to those seen in §3.7.
  - Need to have appropriate form(s) of replication among the X_k's.
  - Can get tricky! See p. 235.
- If serious departures from normality or from variance homogeneity are observed, can apply Box-Cox power transformation to Y_i, as in §3.9.

§6.9: MLR Example – Dwaine Studios (CH06FI05)

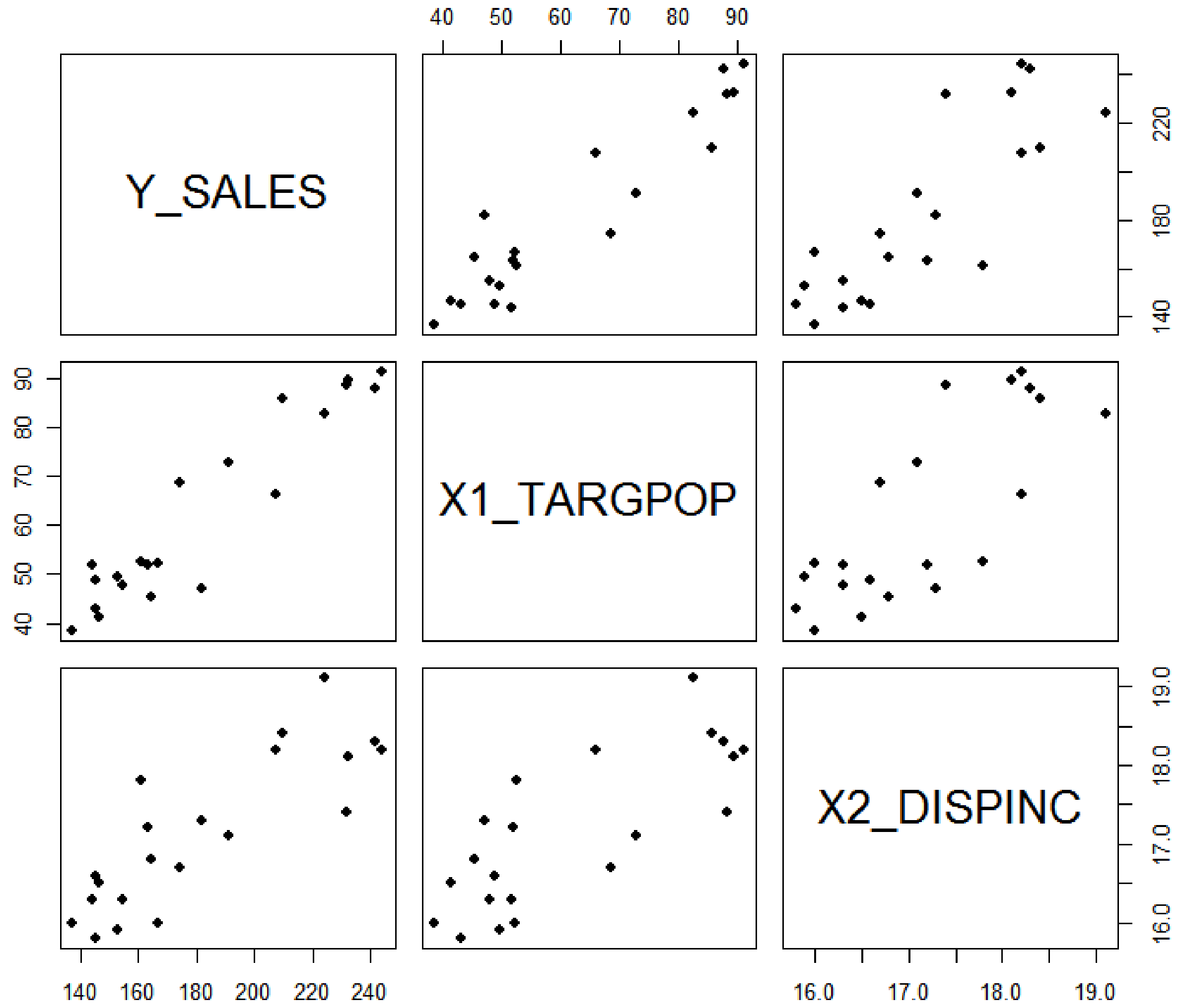
- <u>Example</u>: p=3 (two predictors) with
  - Y = portrait studio sales
  - X_1 = target popl'n below 16 yrs. old
  - X_2 = per cap. disposable income
- Data in Fig. 6.5.
- Start with: (a) scatterplot matrix, and (b) quick check of pairwise correlations.

Dwaine Studios (CH06FI05) Scatterplot Matrix

For the Scatterplot Matrix in R, apply the `pairs()` command to a data frame containing the variables:

```
> CH06FI05.df = data.frame( Y_SALES,
                 X1_TARGPOP, X2_DISPINC )

> pairs( CH06FI05.df )
```

Dwaine Studios Data (CH06FI05)
Scatterplot Matrix



Dwaine Studios (CH06FI05)
Correlation Matrix

For the correlations between Y and the multiple X_k predictor variables, in R apply the `cor()` command to the data frame:

```
> cor( CH06FI05.df )
```

	Y_SALES	X1_TARGPOP	X2_DISPINC
Y_SALES	1.0000000	0.9445543	0.8358025
X1_TARGPOP	0.9445543	1.0000000	0.7812993
X2_DISPINC	0.8358025	0.7812993	1.0000000

Dwaine Studios (CH06FI05) (cont'd)

Fit MLR model with p–1 = 2 predictors:

```
> CH06FI05.lm = lm( Y_SALES ~
          X1_TARGPOP + X2_DISPINC )

> coef( CH06FI05.lm )
   (Intercept)  X1_TARGPOP  X2_DISPINC
    -68.85707      1.45456     9.36550

> summary( CH06FI05.lm )$r.squared
   $r.squared
   [1] 0.9167465

> summary( CH06FI05.lm )$adj.r.squared
   $adj.r.squared
   [1] 0.9074961
```

R^2 (points to 0.9167465)

R_a^2 (points to 0.9074961)

Dwaine Studios (CH06FI05) (cont'd)

Fit MLR model with p–1 = 2 predictors:

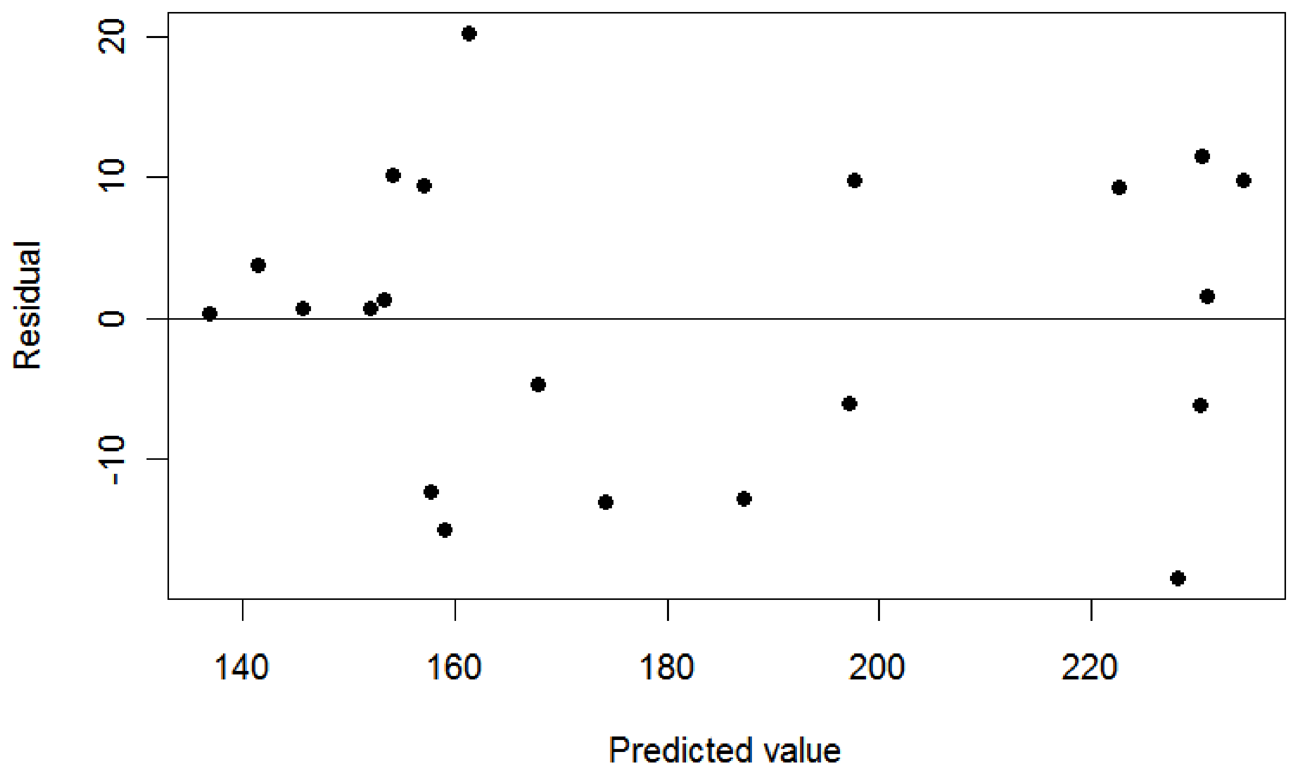
```
> CH06FI05.lm = lm( Y_SALES ~
          X1_TARGPOP + X2_DISPINC )
> coef( CH06FI05.lm )

(Intercept)  X1_TARGPOP  X2_DISPINC
 -68.85707     1.45456     9.36550
```

So, e.g., a unit (\$K) increase in X_2 = {dispos. income} generates a \$9.3655K incr. in sales, when X_1 = target popln. size is held fixed.

Dwaine Studios (CH06FI05) (cont'd)

Residual plot ($e_i = Y_i - \hat{Y}_i$ vs. $\hat{Y}_i$):

```
> plot( resid(CH06FI05.lm) ~ fitted(CH06FI05.lm) )
> abline( h=0 )
```



Dwaine Studios (CH06FI05) (cont'd)

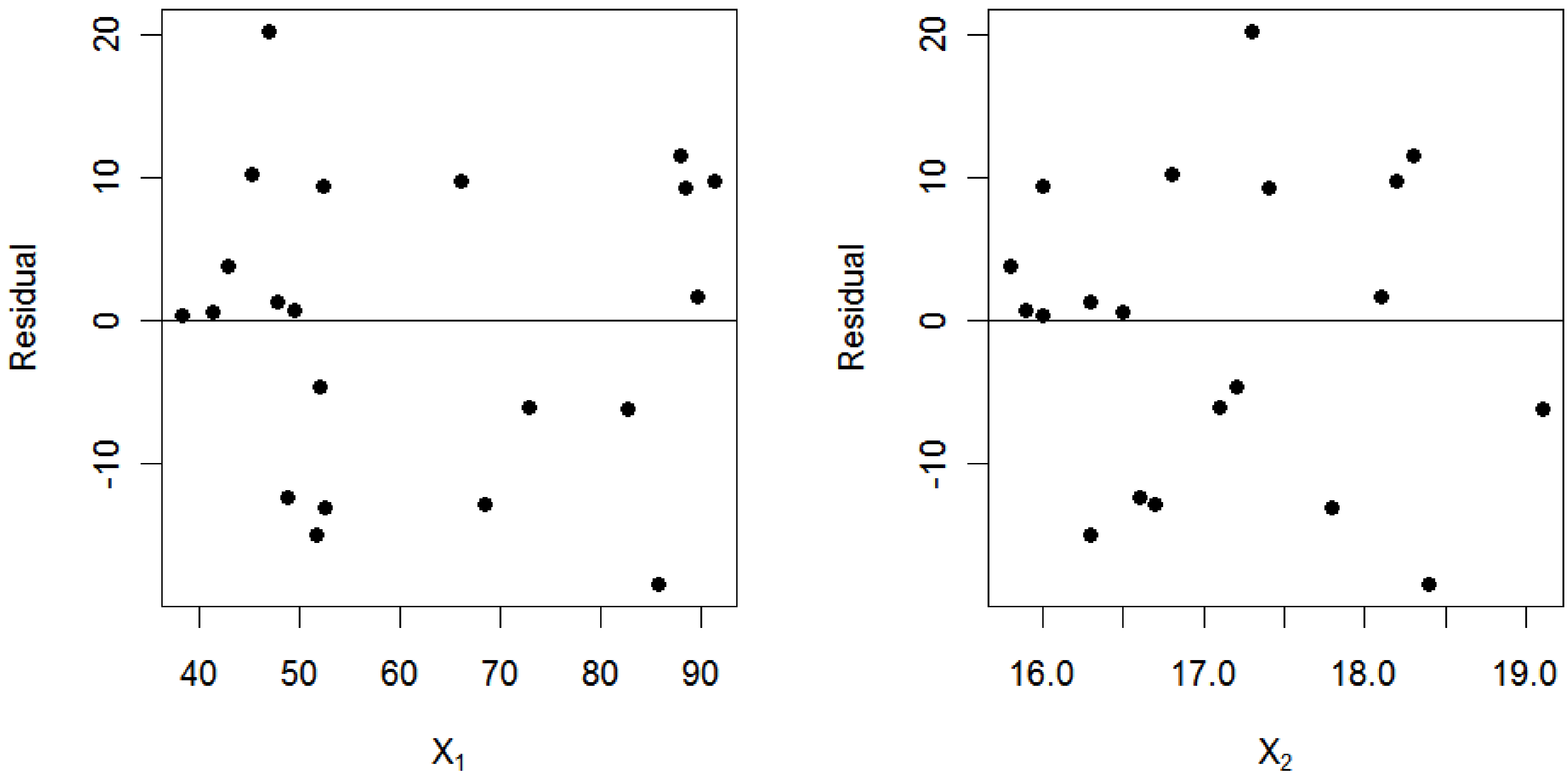
Per-predictor residual plots ($e_i = Y_i - \hat{Y}_i$ vs. each X_{ik}):

```
> par( mfrow=c(1,2) )
> plot( resid(CH06FI05.lm) ~ X1_TARGPOP,
             pch=19, xlab=expression(X[1]),
             ylab='Residual')
> abline( h=0 )
> plot( resid(CH06FI05.lm) ~ X2_DISPINC,
             pch=19, xlab=expression(X[2]),
             ylab='Residual' )
> abline( h=0 )
```

Plots follow ⟶

Dwaine Studios (CH06FI05) (cont'd)

Per-predictor residual plots (cf. Fig. 6.8):



Dwaine Studios (CH06FI05) (cont'd)

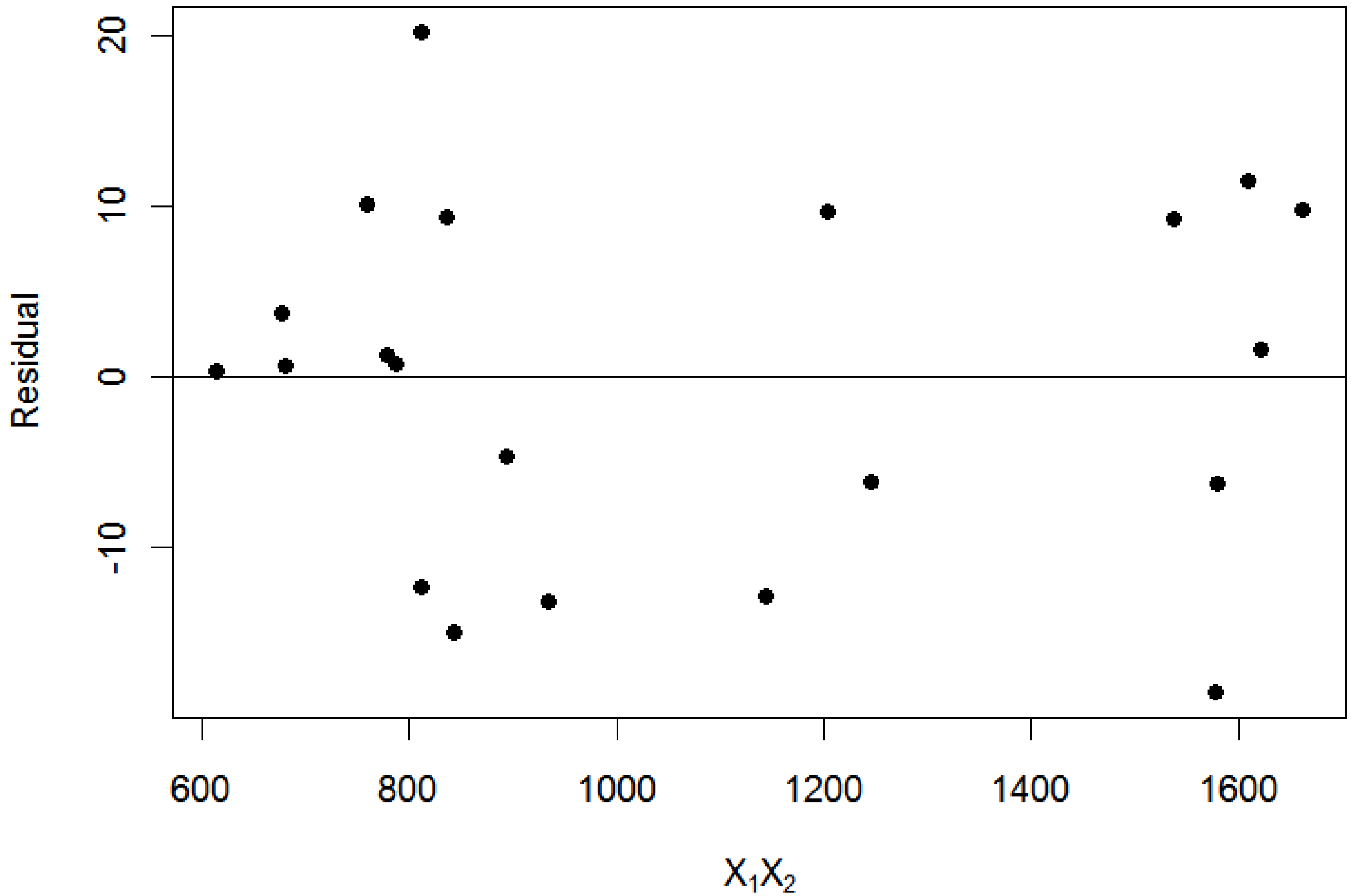
Can also plot residuals against functions of the X_{ik}s; e.g., plot e_i vs. potential interaction term $X_{i1}X_{i2}$:

```
> X3 = X1_TARGPOP * X2_DISPINC

> plot( resid(CH06FI05.lm) ~ X3,
        pch=19, ylab='Residual',
        xlab=expression(X[1]*X[2]) )

> abline( h=0 )
```

A systematic pattern in the plot suggests a need for the interaction term in the full MLR model ⟶

Dwaine Studios (CH06FI05) (cont'd)

Interaction-predictor residual plot shows no systematic pattern (cf. Fig. 6.8d):



Dwaine Studios (CH06FI05) (cont'd)

ANOVA for testing full MLR model with p–1 = 2 predictors:

```
 > anova( lm(Y_SALES ~ 1), CH06FI05.lm )
 Model 1: Y_SALES ~ 1
 Model 2: Y_SALES ~ X1_TARGPOP + X2_DISPINC
 Res.Df     RSS  Df Sum of Sq      F    Pr(>F)
1     20 26196.2
2     18  2180.9  2     24015 99.103 1.921e-10
```

F^* = 99.1 with highly signif. P-value = 1.9×10^{-10}.

Dwaine Studios (CH06FI05) (cont'd)

Pointwise conf. intervals on β_1 and β_2 are available via `confint( CH06FI05.lm )`.

For Bonferroni-corrected simultaneous conf. intervals, adjust via the `level=` option:

```
> g = length( coef(CH06FI05.lm) ) - 1
> alpha = .10
> confint( CH06FI05.lm, level=1-(alpha/g) )[2:3,]

                [lower]    [upper]
  X1_TARGPOP  1.0096226  1.899497
  X2_DISPINC  0.8274411 17.903560
```

(cf. p. 245)

Dwaine Studios (CH06FI05) (cont'd)

- Pointwise **conf. intervals on E{Y_h}** at <u>single</u> future value of predictor vector $X_h' = [X_{h0} \;\; X_{h1} \;\; X_{h2}] = [1 \;\; 65.4 \;\; 17.6]$:
- In R, need to define new data frame containing desired X_h value(s):

```
> newdata.df = data.frame( X1_TARGPOP=65.4,
                            X2_DISPINC=17.6 )
```

- Then, employ `predict.lm()` function:

```
> predict.lm( CH06FI05.lm, newdata=newdata.df,
              se.fit=T, interval='confidence' )
```

Dwaine Studios (CH06FI05) (cont'd)

Output from `predict.lm()`:

```
> predict.lm( CH06FI05.lm, newdata=newdata.df,
              se.fit=T, interval='confidence' )
$fit
        fit        lwr      upr
1  191.1039  185.2911 196.9168

$se.fit
[1] 2.766798

$df
[1] 18
```

Lower and upper 95% limits

$\hat{Y}_h$

std. error, $s\{\hat{Y}_h\}$

Dwaine Studios (CH06FI05) (cont'd)

- Pointwise prediction intervals on Y_h at <u>single</u> future value of predictor vector $\mathbf{X}_h' = [X_{0h} \quad X_{1h} \quad X_{2h}] = [1 \quad 65.4 \quad 17.6]$:
- In R, continue to employ `predict.lm()` function with `newdata.df` data frame, but change `interval=` option:

```
> predict.lm( CH06FI05.lm, newdata=newdata.df,
              se.fit=T, interval='prediction' )
```

Dwaine Studios (CH06FI05) (cont'd)

Prediction from `predict.lm()`:

```
> predict.lm( CH06FI05.lm, newdata=newdata.df,
                              interval='prediction' )
        fit        lwr       upr
1  191.1039  167.2589  214.949
```

$\hat{Y}_h$ (fit)

Lower and upper 95% prediction limits (lwr, upr)

This is a ***pointwise*** prediction interval. For *multiple* X_h's, correction for simultaneity is required.

Dwaine Studios (CH06FI05) (cont'd)

Simultaneous prediction intervals on Y_h at $g = 2$ future values of predictor vector:

$$X_h = \begin{bmatrix} 1 \\ 65.4 \\ 17.6 \end{bmatrix}, \begin{bmatrix} 1 \\ 53.1 \\ 17.7 \end{bmatrix}$$

```
> newdata.df = data.frame( X1_TARGPOP=c(65.4,53.1),
                           X2_DISPINC=c(17.6,17.7) )
> g = nrow( newdata.df )
```

Dwaine Studios (CH06FI05) (cont'd)

- Simultaneous 90% prediction intervals on Y_h
- Begin with check of Scheffé vs. Bonferroni critical points:

```
> alpha = .10
> Spoint = sqrt( g*qf(1-alpha, g,
                           CH06FI05.lm$df.residual) )
 [1] 2.290828

> Bpoint = qt( 1-(.5*(alpha/g)),
                            CH06FI05.lm$df.residual )
 [1] 2.100922
```

- $B = 2.10 \leq S = 2.29$, so apply Bonferroni adjustment

Dwaine Studios (CH06FI05) (cont'd)

Simultaneous 90% prediction intervals on Y_h with Bonferroni critical point at g=2:

```
> newdata.df

   X1_TARGPOP  X2_DISPINC
1        65.4        17.6
2        53.1        17.7

> predict.lm( CH06FI05.lm, newdata=newdata.df,
              interval='prediction',
              level=1-(alpha/g)) )

     fit        lwr        upr
1  191.1039   167.2589   214.9490
2  174.1494   149.0867   199.2121
```