Printout

Wednesday, March 29, 2023 2:49 PM

Section 1

MANE 6313

Subsection 1

Week 12, Module A

Student Learning Outcome

- Select an appropriate experimental design with one or more factors,
- Select an appropriate model with one or more factors,
- Evaluate statistical analyses of experimental designs,
- Assess the model adequacy of any experimental design, and
- Interpret model results.

Module Learning Outcome

Describe linear regression.

simple linear (1x, 1y)

Introduction to Linear Regression

$y = \beta_0 + \beta_1 x + \varepsilon$

- We are interested in a relationship between a single *dependent variable* or *response* y that depends on k *independent* or *regressor variables*.
- We assume that there is some mathematical function (unknown) $y = \phi(x_1, x_2, \ldots, x_k)$. In general, we don't know this function
- We'll use low order polynomial equations as an approximating function. This is called *empirical modeling*. → neural networks
- What are methods that we can determine if there is a relationship between two (or more) variables?
  1) graphical analysis
  2) correlation

Relationship between two or more variables

Example 12.8 Suppose that a scientist takes experimental data on the radius of a propellant grain Y as a function of powder temperature x_1, extrusion rate x_2, and die temperature x_3. Fit a linear regression model for predicting grain radius, and determine the effectiveness of each variable in the model. The data are as follows:

Y **Grain radius**	X_1 **Powder temperature**	X_2 **Extrusion rate**	X_3 **Die temperature**
82	150 (−1)	12 (−1)	220 (−1)
93	190 (1)	12 (−1)	220 (−1)
114	150 (−1)	24 (1)	220 (−1)
124	150 (−1)	12 (−1)	250 (1)
111	190 (1)	24 (1)	220 (−1)
129	190 (1)	12 (−1)	250 (1)
157	150 (−1)	24 (1)	250 (1)
164	190 (1)	24 (1)	250 (1)

Figure 1: Example 12.8

uncoded - raw

$\beta_{11} x_1^2$, let $z = x_1^2$

$\beta_{11} z_1$

Linear regression models

- In general they look like

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2 + \varepsilon$$

- This model is *linear* in the parameters β

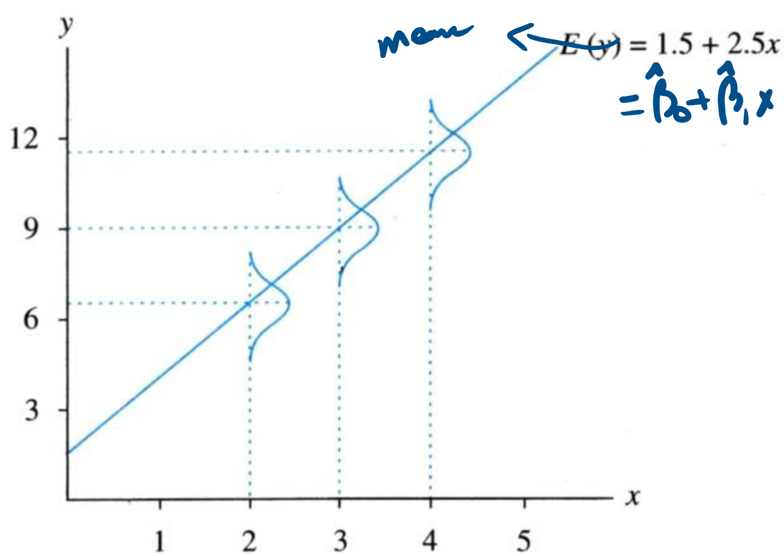
non linear

$e^{-\beta x}$

- See graphical explanation from Ott.

measurement to another.

FIGURE 11.2
Theoretical distribution of y in regression



① Normality is not required to fit a linear regression
② Normality is used for hypothesis

Figure 2: Figure 11.2

Taken from ... Ott (2001)[1]

Estimation of Parameters

- Parameter estimates are derived using least squares. Goal is to minimize the squared error.
- Parameter estimation can be done algebraically or using linear algebra. Montgomery focuses on linear algebra formulation (matrices)
- In general, the matrix formulation is used. Model is defined to be

$$\mathbf{y} = \mathbf{X}\beta + \varepsilon$$

- The least squares estimates are found by minimizing

$$L = \sum_i \varepsilon_i^2 = \varepsilon'\varepsilon = (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta)$$

- The least squares estimates must satisfy

$$\frac{\partial L}{\partial \beta}\Big|_{\widehat{\beta}} = -2\mathbf{X}'\mathbf{y} + 2\mathbf{X}'\mathbf{X}\widehat{\beta} = 0$$

- Which leads to the solution

$$\widehat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

$$X = \begin{bmatrix} 1 & x_{11} & x_{21} & x_{31} \\ 1 & x_{12} & x_{22} & x_{32} \\ \vdots & x_{1n} & x_{2n} & x_{3n} \end{bmatrix}$$

- We can define the predicted response to be

$$\widehat{\mathbf{y}} = \mathbf{X}\widehat{\beta}$$

- The residuals are defined to be

actual predicted

$$\mathbf{e} = \mathbf{y} - \widehat{\mathbf{y}}$$

- Thus the sum of squares errors can be shown to be

$$\begin{aligned} SS_E = & \ (\mathbf{y} - \mathbf{X}\widehat{\beta})'(\mathbf{y} - \mathbf{X}\widehat{\beta}) \\ = & \ \mathbf{y}'\mathbf{y} - \widehat{\beta}'\mathbf{X}'\mathbf{y} \end{aligned}$$

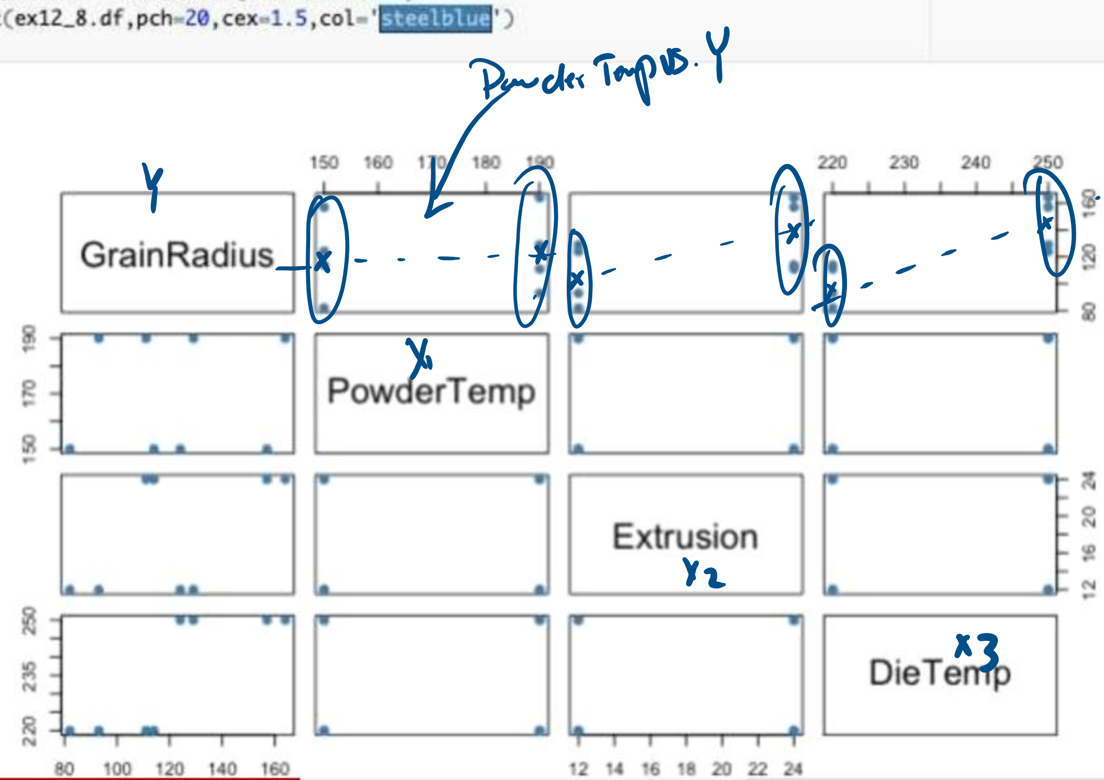
Coding Variables

- From the example, there were two ways to represent the same problem, coded and uncoded
- Why use coded variables[a]
  - Computational ease and increased accuracy in estimating the model coefficients
  - Enhanced interpretability of the coefficient estimates in the model.
- Internally most statistical software codes for estimating parameters

[a] Khuri, and Cornell (1987). Response Surfaces: Designs and Analyses. Dekker

Plot: Example 12.8

```
```{r}
library(readxl)
ex12_8.df <- read_excel("ex12_8.xlsx")
plot(ex12_8.df,pch=20,cex=1.5,col='steelblue')
```
```



Regression: Example 12.8

```
```{r}
ex12_8.model <- lm(GrainRadius~PowderTemp+Extrusion+DieTemp,data=ex12_8.df)
summary(ex12_8.model)
```
```

```
Call:
lm(formula = GrainRadius ~ PowderTemp + Extrusion + DieTemp,
    data = ex12_8.df)

Residuals:
    1     2     3     4     5     6     7     8
-0.75  5.25  1.75 -2.25 -6.25 -2.25  1.25  3.25

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -284.50000   30.77729  -9.244 0.000761 ***
PowderTemp     0.12500    0.08501   1.470 0.215398
Extrusion      2.45833    0.28336   8.676 0.000972 ***
DieTemp        1.45000    0.11335  12.793 0.000215 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.809 on 4 degrees of freedom
Multiple R-squared:  0.9837,	Adjusted R-squared:  0.9714
F-statistic: 80.36 on 3 and 4 DF,  p-value: 0.0004967
```

$\hat{y} = -284.5 + .125\, x_1 + 2.458\, x_2 + 1.45\, x_3$

β_0

Not Significant

Anova — Model 3 — p-value .0005

Model as a whole is significant

Values from lm() function

Source: https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/lm

Value

`lm` returns an object of `class` `"lm"` or for multiple responses of class `c("mlm", "lm")`.

The functions `summary` and `anova` are used to obtain and print a summary and analysis of variance table of the results. The generic accessor functions `coefficients`, `effects`, `fitted.values` and `residuals` extract various useful features of the value returned by `lm`.

An object of class `"lm"` is a list containing at least the following components:

`coefficients`	a named vector of coefficients
`residuals`	the residuals, that is response minus fitted values.
`fitted.values`	the fitted mean values.
`rank`	the numeric rank of the fitted linear model.
`weights`	(only for weighted fits) the specified weights.
`df.residual`	the residual degrees of freedom.
`call`	the matched call.
`terms`	the `terms` object used.
`contrasts`	(only where relevant) the contrasts used.
`xlevels`	(only where relevant) a record of the levels of the factors used in fitting.
`offset`	the offset used (missing if none were used).
`y`	if requested, the response used.