Reminder
Figure 11-1  Scatter diagram of oxygen purity versus hydrocarbon level from Table 11-1.

\[ Y = \beta_0 + \beta_1 X + \varepsilon \]
$Y = \beta_0 + \beta_1 X + \epsilon$  \quad $E(\epsilon | x) = 0 \quad \forall \ x$

How does one find $\beta_0$ & $\beta_1$?

$\text{Cov}(Y, x) = \text{Cov}((\beta_0 + \beta_1 X + \epsilon), x) = \text{Cov}(\beta_0, x) + \beta_1 \text{Cov}(x, x) + \text{Cov}(\epsilon, x)$

$\text{Cov}(\beta_0, x) = 0 \quad \text{since } \beta_0 \text{ is constant}$

$\text{Cov}(x, x) = E(x^2) - E(x)^2 = \text{Var}(x)$

$\text{Cov}(\epsilon, x) = E(\epsilon \cdot x) - E(\epsilon) \cdot E(x) = E(\epsilon \cdot x) = \sum x \cdot E(\epsilon | x) = 0$

Thus $\beta_1 = \frac{\text{Cov}(x, y)}{\text{Var}(x)}$; $\beta_0 = E(y) - \beta_1 E(x)$
Method of least squares

- The *method of least squares* is used to estimate the parameters, $\beta_0$ and $\beta_1$ by minimizing the sum of the squares of the vertical deviations in Figure 11-3.

**Figure 11-3** Deviations of the data from the estimated regression model.
Multiple Linear Regression
(Chapters 12-13 in Montgomery, Runger)
12-1: Multiple Linear Regression Model

12-1.1 Introduction

• Many applications of regression analysis involve situations in which there are more than one regressor variable $X_k$ used to predict $Y$.

• A regression model then is called a multiple regression model.
Multiple Linear Regression Model

\[ Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \ldots + \beta_k x_k + \varepsilon \]

One can also use powers and products of other variables or even non-linear functions like \( \exp(x_i) \) or \( \log(x_i) \) instead of \( x_3, \ldots, x_k \).

Example: the general two-variable quadratic regression has 6 constants:

\[ Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 (x_1)^2 + \beta_4 (x_2)^2 + \beta_5 (x_1 x_2) + \varepsilon \]
Logistic Regression

\[ P(y=1) = \sigma(x_1w_1 + x_2w_2 + b) \]
12-1: Multiple Linear Regression Model

12-1.3 Matrix Approach to Multiple Linear Regression

Suppose the model relating the regressors to the response is

\[ y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} + \epsilon_i \quad i = 1, 2, \ldots, n \]

In matrix notation this model can be written as

\[ y = X\beta + \epsilon \quad (12-6) \]
12-1: Multiple Linear Regression Model

12-1.3 Matrix Approach to Multiple Linear Regression

where

\[
\begin{align*}
\mathbf{y} &= \begin{bmatrix}
y_1 \\
y_2 \\
\vdots \\
y_n \\
\end{bmatrix}, \\
\mathbf{X} &= \begin{bmatrix}
1 & x_{11} & x_{12} & \cdots & x_{1k} \\
1 & x_{21} & x_{22} & \cdots & x_{2k} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & x_{n1} & x_{n2} & \cdots & x_{nk} \\
\end{bmatrix}, \\
\mathbf{\beta} &= \begin{bmatrix}
\beta_0 \\
\beta_1 \\
\vdots \\
\beta_k \\
\end{bmatrix}, \quad \text{and} \quad \mathbf{\varepsilon} &= \begin{bmatrix}
\varepsilon_1 \\
\varepsilon_2 \\
\vdots \\
\varepsilon_n \\
\end{bmatrix}
\end{align*}
\]
12-1.3 Matrix Approach to Multiple Linear Regression

We wish to find the vector $\hat{\beta}$ that minimizes the sum of squares of error terms:

$$L = \sum_{i=1}^{n} \varepsilon_i^2 = \varepsilon' \varepsilon = (y - X\beta)' (y - X\beta)$$

$$0 = \frac{\partial L}{\partial \beta} = -X' (y - X\beta) = -X' y + (X'X)\beta$$

The resulting least squares estimate is

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

(12-7)
Multiple Linear Regression Model

\[ \hat{\beta} = (X'X)^{-1} X'y \]

\[ \hat{y} = X\hat{\beta} = X(X'X)^{-1}X'y, \]

\[ \hat{y} = Hy, \quad \text{and} \quad e = (I - H)y. \]

\[ H = H' = H^2 = X(X'X)^{-1}X'X(X'X)^{-1}X = X(X'X)^{-1}X'X = H \]

Vectors \( \hat{y} \) & \( e \) are orthogonal since

\[ \hat{y}'e = y'H(I - H)y = 0 \quad \text{since} \]

\[ H(I - H) = H - H^2 = H - H = 0. \]
12-1: Multiple Linear Regression Models

12-1.4 Properties of the Least Squares Estimators

Unbiased estimators:

\[
E(\hat{\beta}) = E[(X'X)^{-1}X'Y] \\
= E[(X'X)^{-1}X'(X\beta + \epsilon)] \\
= E[(X'X)^{-1}X'X\beta + (X'X)^{-1}X'\epsilon] \\
= \beta
\]

Covariance Matrix of Estimators:

\[
C = (X'X)^{-1} = \begin{bmatrix}
    C_{00} & C_{01} & C_{02} \\
    C_{10} & C_{11} & C_{12} \\
    C_{20} & C_{21} & C_{22}
\end{bmatrix}
\]
12-1: Multiple Linear Regression Models

12-1.4 Properties of the Least Squares Estimators

Individual variances and covariances:

\[ V(\hat{\beta}_j) = \sigma^2 C_{jj}, \quad j = 0, 1, 2 \]
\[ \text{cov}(\hat{\beta}_i, \hat{\beta}_j) = \sigma^2 C_{ij}, \quad i \neq j \]

In general,

\[ \text{cov}(\hat{\beta}) = \sigma^2 (X'X)^{-1} = \sigma^2 C \]
12-1: Multiple Linear Regression Models

Estimating error variance $\sigma^2_{\varepsilon}$

An unbiased estimator of error variance $\sigma^2_{\varepsilon}$ is

$$\hat{\sigma}^2_{\varepsilon} = \frac{\sum_{i=1}^{n} e_i^2}{n - p} = \frac{SS_E}{n - p}$$  (12-16)

Here $p = k + 1$ for $k$-variable multiple linear regression
R² and Adjusted R²

The coefficient of multiple determination R²

\[ R^2 = \frac{SS_R}{SS_T} = 1 - \frac{SS_E}{SS_T} \]

The adjusted R² is

\[ R_{adj}^2 = 1 - \frac{SS_E/(n - p)}{SS_T/(n - 1)} \]  (12-23)

- The adjusted R² statistic penalizes adding terms to the MLR model.
- It can help guard against overfitting (including regressors that are not really useful)
How to know where to stop?

• Adding new variables $x_i$ to MLR watch the adjusted $R^2$

• Once the adjusted $R^2$ no longer increases = stop.
Now you did the best you can.
Matlab exercise

• Every group works with
g0=2907; g1=1527; g2=2629; g3=2881;
g4=1144; g5=1066;

• Compute **Multiple Linear Regression (MLR)**: where
  
y=exp_t (g0); x1= exp_t (g1); x2= exp_t (g2);

• **How much better** the MLR did compared to the Single Linear Regression (SLR)?

• **Continue increasing** the number of genes in x until R_adj starts to decrease
How I did it

• \( g_0 = 2907; \ g_1 = 1527; \ g_2 = 2629; \ g_3 = 2881; \ g_4 = 1144; \ g_5 = 1066; \)
• \( y = \text{exp}_t(g_0,:)' \)
• \% first use one x to predict y
  • \( x = \text{exp}_t(g_1,:)' \)
  • figure; plot(x,y,'ko')
  • \( \text{lm} = \text{fitlm}(x,y) \)
  • \( y_{\text{fit}} = \text{lm}.\text{Fitted}; \)
  • hold on;
  • plot(x,lm.Fitted,'r-');
• \% now use 2 x's to predict y
  • \( x = [\text{exp}_t(g_1,:), \ \text{exp}_t(g_2,:)]' \)
  • \( \text{lm}_2 = \text{fitlm}(x,y) \)
  • \( y_{\text{fit}} = \text{lm}_2.\text{Fitted}; \)
  • hold on; plot(x(:,1),y_fit,'gd');
• \% now use m x's to predict y
  • \( \text{corr}_\text{matrix} = \text{corr}([\exp_t(g_1,:)', \ldots]; \)
  • \( g_0 = 2907; \)
  • \[u \ v\] = sort(corr_matrix(g0,:), 'descend');
  • \( x = [\exp_t(v(2:m+1),:])' \)
  • \( \text{lm}_3 = \text{fitlm}(x,y) \)
  • \( y_{\text{fit}} = \text{lm}_3.\text{Fitted}; \)
  • plot(x(:,1),y_fit,'s');