

Summary

Steps in Model Building

Identify the response (dependent) variable y .

Classify each potential predictor (independent) variable as quantitative or qualitative.

Define qualitative dummy variables. Select one to be the base level, setup dummy variables for rest.

Consider higher-order terms (e.g., x^2, x^3) for quantitative variables.

Possibly code the quantitative variables in higher-order polynomials.

Consider interaction terms for both quantitative and qualitative independent variables.

Compare nested models using partial F-tests to arrive at a final model.

General Form of the Multiple Regression Model:

$$E(y) = \beta_0 + \beta_1 x_1 + \beta_2 x_1^2 + \beta_3 x_2 + \beta_4 x_3 + \beta_5 x_1 x_2 + \beta_6 x_1 x_3 + \beta_7 x_1^2 x_2 + \beta_8 x_1^2 x_3 \quad (1 \text{ Quantitative, 1 Qualitative at 3 levels})$$

Coefficient and Residual Variance Estimation:

$Y = X\beta + E$ $\hat{\beta} = (X'X)^{-1}X'y$ $s^2 = \frac{(y - X\hat{\beta})'(y - X\hat{\beta})}{n - k - 1}$ $MSE = s^2, s = \sqrt{s^2}$	$Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}, X = \begin{bmatrix} 1 & x_{11} & x_{21} & \cdots & x_{kn} \\ 1 & x_{12} & x_{22} & \cdots & x_{k2} \\ 1 & x_{13} & x_{23} & \cdots & x_{k3} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{1n} & x_{2n} & \cdots & x_{kn} \end{bmatrix}, \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix}, E = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{bmatrix}$
--	---

Assumptions About the Random Error ε :

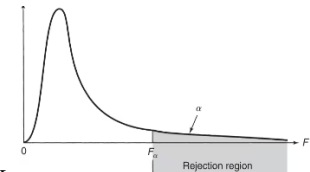
1. For any given $x_1, \dots, x_k$, the error ε has a normal distribution with, $E(\varepsilon)=0$ and $\text{var}(\varepsilon)=\sigma^2$.

2. The random errors are independent, $f(\varepsilon_i, \varepsilon_j) = f(\varepsilon_i)f(\varepsilon_j)$. Normal only needed for CIs and HTs.

Model Test: $H_0: \beta_1 = \beta_2 = \dots = \beta_k = 0$ vs. H_a : At least one $\beta_i \neq 0$.

$$F = \frac{(SS_{yy} - SSE) / k}{SSE / (n - k - 1)} = \frac{\text{Mean Square Model}}{MSE} = \frac{R^2 / k}{(1 - R^2) / (n - k - 1)}$$

Reject if $F > F_{\alpha, k, n-k-1}$ or $\alpha > p\text{-value} = P(F > F_{\alpha, k, n-k-1})$.

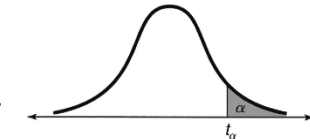


Individual Coefficient Test: $t = \hat{\beta}_i / s_{\hat{\beta}_i}$, $s_{\hat{\beta}_i} = s \sqrt{W_{ii}}$, W_{ii} is the i^{th} diagonal of $W = (X'X)^{-1}$.

One Tailed: $H_0: \beta_i \geq 0$ vs. $H_a: \beta_i < 0$ w/ RR $t < -t_{\alpha, n-k-1}$ or $\alpha > p\text{-value} = P(t < -t_{\alpha, n-k-1})$,

One Tailed: $H_0: \beta_i \leq 0$ vs. $H_a: \beta_i > 0$ w/ RR $t > t_{\alpha, n-k-1}$ or $\alpha > p\text{-value} = P(t > t_{\alpha, n-k-1})$,

Two Tailed: $H_0: \beta_i = 0$ vs. $H_a: \beta_i \neq 0$ w/ RR $|t| > t_{\alpha/2, n-k-1}$ or $\alpha > p\text{-value} = 2P(|t| > t_{\alpha/2, n-k-1})$.



Coefficient Confidence Interval:

$$\hat{\beta}_i \pm t_{\alpha/2, n-k-1} s \sqrt{W_{ii}}, s^2 = MSE \text{ and } W_{ii} \text{ is the } i^{\text{th}} \text{ diagonal element of } W = (X'X)^{-1}.$$

Coefficient of determination R^2 and adjusted coefficient of determination R_a^2 . Fit quality.

$$R^2 = 1 - SSE / SS_{yy}, \quad 0 \leq R^2 \leq 1, \quad SSE = \sum (y_i - \hat{y}_i)^2, \quad SS_{yy} = \sum (y_i - \bar{y})^2$$

$$R_a^2 = 1 - [SSE / (n - k - 1)] / [SS_{yy} / (n - 1)] = 1 - [(n - 1) / (n - k - 1)] (1 - R^2), \quad R_a^2 \leq R^2$$

Estimated mean function at x_0 : $\hat{y}(x_0) = x_0' \hat{\beta}$, $SE(\hat{y}_{x_0}) = \sqrt{MSE(x_0'(X'X)^{-1}x_0)}$

Mean function confidence interval at x_0 : $CI = \hat{y}(x_0) \pm t_{\alpha/2, n-k-1} SE(\hat{y}_{x_0})$

Mean function prediction interval at x_0 : $PI = \hat{y}(x_0) \pm t_{\alpha/2, n-k-1} \cdot \sqrt{MSE + (SE(\hat{y}_{x_0}))^2}$

F-Test for Comparing Nested Models

Reduced Model: $E(y|x) = \beta_0 + \beta_1 x_1 + \dots + \beta_g x_g$

Complete Model: $E(y|x) = \beta_0 + \beta_1 x_1 + \dots + \beta_g x_g + \beta_{g+1} x_{g+1} + \dots + \beta_k x_k$

$H_0: \beta_{g+1} = \dots = \beta_k = 0$ vs. H_a : At least one tested $\beta_i \neq 0$.

$$F = \frac{(SSE_R - SSE_C) / (k - g)}{SSE_C / (n - k - 1)}, \text{ Reject if } F > F_{\alpha, k-g, n-k-1} \text{ or } \alpha > p\text{-value} = P(F > F_{\alpha, k-g, n-k-1}).$$

MATH 2780 Chapter5 Worksheet

Example: Solving a system of equations by inverting a matrix.

In algebra we learned that we can solve a system of equations to find x_1 and x_2 .

$$\begin{aligned} x_1 + x_2 &= 2 \\ x_1 - x_2 &= 4 \end{aligned} \quad \text{add} \rightarrow 2x_1 = 6 \rightarrow x_1 = 3, x_2 = -1.$$

Which can be written using arrays (matrices) as $\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \end{bmatrix}$.

We learned that $ax=b$ can solve for x as $x=b/a=a^{-1}b$. We are going to do something similar.

Define $A = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$, $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, and $b = \begin{bmatrix} 2 \\ 4 \end{bmatrix}$.

Then, we can write the above system of equations as

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \text{ or } Ax = b.$$

We are going to multiply both sides by the inverse of A . (Uppercase letter for matrices and lower case for vectors and scalars. Matrix has more than one column while vector only has one column.)

$$A^{-1}Ax = A^{-1}b \rightarrow x = A^{-1}b.$$

We can find that $A^{-1} = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & -1/2 \end{bmatrix} \rightarrow x = A^{-1}b = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & -1/2 \end{bmatrix} \begin{bmatrix} 2 \\ 4 \end{bmatrix} = \begin{bmatrix} 3 \\ -1 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$.

When A is not square,

$$A^{-1} = \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 2 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix}, \text{ we need to multiply by transpose of } A \text{ first, } A'Ax = A'b, \text{ then}$$

multiply by the inverse of $A'A$. That is, $(A'A)^{-1}(A'A)x = (A'A)^{-1}A'b$, $x = (A'A)^{-1}A'b$.

$$A'A = \begin{bmatrix} 6 & 6 \\ 6 & 11 \end{bmatrix}, \rightarrow (A'A)^{-1} = \begin{bmatrix} 0.3667 & -0.2000 \\ -0.2000 & 0.2000 \end{bmatrix} \rightarrow$$

$$(A'A)^{-1}A = \begin{bmatrix} 0.3667 & -0.2000 \\ -0.2000 & 0.2000 \end{bmatrix} \begin{bmatrix} 1 & 1 & 2 \\ 1 & -1 & 3 \end{bmatrix} = \begin{bmatrix} 0.1667 & 0.5667 & 0.1333 \\ 0.0000 & -0.4000 & 0.2000 \end{bmatrix}$$

$$b = (A'A)^{-1}Ax = \begin{bmatrix} 0.1667 & 0.5667 & 0.1333 \\ 0.0000 & -0.4000 & 0.2000 \end{bmatrix} \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix} = \begin{bmatrix} 3.4000 \\ -0.4000 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

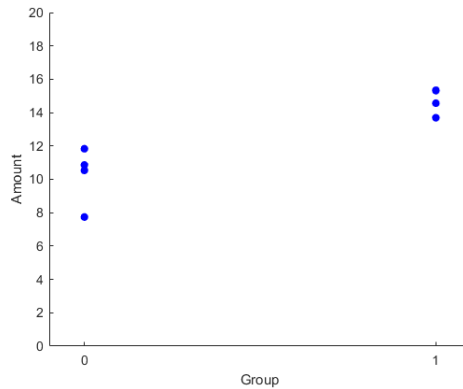
This is our most important equation in statistics $\hat{\beta} = (X'X)^{-1}X'y$!

MATH 2780 Chapter5 Worksheet

Example 1: One quantitative response variable y depending on one qualitative variable x with two levels.

a) Develop a coding scheme for the qualitative variable x and an expectation model $E(y)$.

Person	Level	y
1	Low	10.5
2	Low	11.8
3	Low	7.7
4	Low	10.9
5	High	15.3
6	High	13.7
7	High	14.6
8	High	15.3



Person	X	y
1	1	0
2	1	0
3	1	0
4	1	0
5	1	1
6	1	1
7	1	1
8	1	1

$y = \beta_0 + \beta_1 x + \varepsilon$, $x=0$ if low and $x=1$ if high.

$E(y(0)) = \beta_0 + \beta_1(0) = \beta_0$, $E(y(1)) = \beta_0 + \beta_1(1) = \beta_0 + \beta_1$, so $\beta_0 = \mu_L$ and $\beta_1 = \mu_H - \mu_L$.

b) Estimate the regression coefficients and MSE. $\hat{\beta} = (X'X)^{-1}X'y$

Step through code.

c) Compute the mean function at $x_0 = [1, 0]$ and $x_0 = [1, 1]$. $\hat{y}(x_0) = x_0 \hat{\beta}$, $SE(\hat{y}_{x_0}) = \sqrt{MSE(x_0(X'X)^{-1}x_0')}$

Step through code.

d) Compute the mean function CI at x_0 . $CI = \hat{y}(x_0) \pm t_{\alpha/2, n-k-1} SE(\hat{y}_{x_0})$

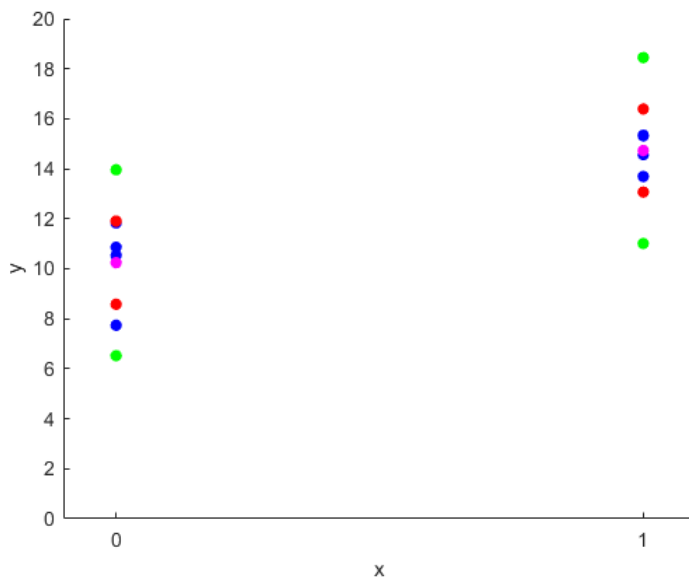
Step through code.

e) Compute the mean function PI at x_0 . $PI = \hat{y}(x_0) \pm t_{\alpha/2, n-k-1} \cdot \sqrt{MSE + (SE(\hat{y}_{x_0}))^2}$

Step through code.

f) Test $H_0: \beta_2 = 0$ vs. $H_a: \beta_2 \neq 0$. $t = \hat{\beta}_i / s_{\hat{\beta}_i}$, $s_{\hat{\beta}_i} = s \sqrt{W_{ii}}$, W_{ii} is the i^{th} diagonal of $W = (X'X)^{-1}$.

Step through code.



MATH 2780 Chapter5 Worksheet

<pre> #R Code # set seed to same default value set.seed(NULL) # yi=b0+b1*xi+ei, y=X*b+e nA<-4 nB<-4 n <-nA+nB x <-c(rep(0,nA),rep(1,nB)) X <-cbind(rep(1,n),x) k <-ncol(X)-1 beta <-c(10,5) sigma<-1 y<-X%%beta+rnorm(n) cbind(X,y) plot(x,y,xlab="Level",ylab="Amount",pch=19) # estimate coefficients W <-solve(t(X)%*%X) b <- W%%t(X)%*%y SSE<-t(y-X%%b)%*%(y-X%%b) MSE<-SSE/(n-k-1) # plot x0<-rbind(c(1,0),c(1,1)) m <-nrow(x0) # calculate fitted, CIs, and PIs yfit <- x0%%b; yfitCIL<-c(rep(0,m)) yfitCIU<-c(rep(0,m)) yfitPIL<-c(rep(0,m)) yfitPIU<-c(rep(0,m)) tcrit<-qt(1-alpha,n-k-1) for(i in 1:m) { yfitCIL[i]<-x0[i,]%*%b -tcrit*sqrt(MSE* t(x0[i,])%%W%%x0[i,]) yfitCIU[i]<-x0[i,]%*%b +tcrit*sqrt(MSE* t(x0[i,])%%W%%x0[i,]) yfitPIL[i]<-x0[i,]%*%b -tcrit*sqrt(MSE*(1+t(x0[i,])%%W%%x0[i,])) yfitPIU[i]<-x0[i,]%*%b +tcrit*sqrt(MSE*(1+t(x0[i,])%%W%%x0[i,])) } </pre>	<pre> # plot fit, CIs, and PIs plot (X[,2],y ,xlab="Level",ylab="Amount", pch=19, cex = .5, col = "blue",xlim = c(0,1),ylim = c(0,16)) points(x0[,2],yfit,xlab="Level",ylab="Amount", pch=19,cex = .5,col = "magenta",xlim = c(0,1), ylim = c(0,16)) points(x0[,2],yfitCIL,xlab="Level",ylab="Amount", pch=19,cex = .5,col = "red",xlim = c(0,1), ylim = c(0,16)) points(x0[,2],yfitCIU,xlab="Level",ylab="Amount", pch=19, cex = .5,col = "red",xlim = c(0,1), ylim = c(0,16)) points(x0[,2],yfitPIL,xlab="Level",ylab="Amount", ,pch=19, cex = .5,col = "green",xlim = c(0,1), ylim = c(0,16)) points(x0[,2],yfitPIU,xlab="Level",ylab="Amount", pch=19, cex = .5,col = "green",xlim = c(0,1), ylim = c(0,16)) # t-test if beta1=0 alph<-0.05; tb0 <-b[1]/sqrt(MSE*W[1,1]) tb1 <-b[2]/sqrt(MSE*W[2,2]) </pre>
---	---

```

% Matlab Code
% set seed to same default value
rng('default')

% yi=b0+b1*xi+ei, y=X*b+e
nA=4; nB=4; n=nA+nB;
x=[zeros(nA,1);ones(nA,1)];
X=[ones(n,1),x];
k=size(X,2)-1;
beta=[10;5];
sigma=1;

y=X*beta+randn(n,1);
[X,y]

figure;
scatter(X(:,2),y, 'filled', 'blue')
xlabel('Level'), ylabel('Amount')
set(gca, 'xtick', [0:1])
xlim([-0.1,1.1]),ylim([0,20])

% estimate coefficients
W =inv(X'*X)
b=W*X'*y
SSE=(y-X*b)'*(y-X*b);
MSE=SSE/(n-k-1)

% plot
x0=[1,0;1,1]
m=size(x0,1);
yfit = x0*b;
yfitCIL=zeros(m,1);
yfitCIU=zeros(m,1);
yfitPIL=zeros(m,1);
yfitPIU=zeros(m,1);

```

```

for i=1:m
    yfitCIL(i,1)=x0(i,:)*b...
        -tinv(1-0.05/2,n-k-1)...
        *sqrt(MSE* x0(i,:)*W*x0(i,:)') );
    yfitCIU(i,1)=x0(i,:)*b...
        +tinv(1-0.05/2,n-k-1)...
        *sqrt(MSE* x0(i,:)*W*x0(i,:)') );
    yfitPIL(i,1)=x0(i,:)*b...
        -tinv(1-0.05/2,n-k-1)...
        *sqrt(MSE*(1+x0(i,:)*W*x0(i,:)')));
    yfitPIU(i,1)=x0(i,:)*b...
        +tinv(1-0.05/2,n-k-1)...
        *sqrt(MSE*(1+x0(i,:)*W*x0(i,:)')));
end

figure;
scatter(X(:,2),y, 'filled', 'blue')
hold on
scatter(x0(:,2),yfit, 'filled', 'magenta')
scatter(x0(:,2),yfitCIL, 'filled', 'red')
scatter(x0(:,2),yfitCIU, 'filled', 'red')
scatter(x0(:,2),yfitPIL, 'filled', 'green')
scatter(x0(:,2),yfitPIU, 'filled', 'green')
xlabel('x'), ylabel('y')
set(gca, 'xtick', [0:1])
xlim([-0.1,1.1]),ylim([0,20])

% t-test if beta1=0
alph=0.05;
tb1 =b(2,1)/sqrt(MSE*W(2,2))
tcrit=tinv(1-alph,n-k-1)

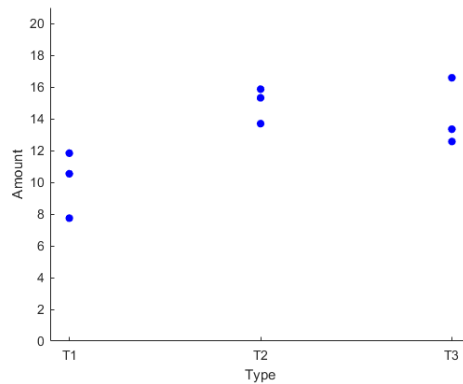
```

MATH 2780 Chapter5 Worksheet

Example 2: One quantitative response variable y depending on one qualitative variable with three types.

a) Develop a coding scheme for the qualitative variable and an expectation model $E(y)$.

Person	Type	y
1	T1	10.5
2	T1	11.8
3	T1	7.7
4	T2	15.9
5	T2	15.3
6	T2	13.7
7	T3	12.6
8	T3	13.3
9	T3	16.6



Person	X			y
1	1	0	0	10.5
2	1	0	0	11.8
3	1	0	0	7.7
4	1	1	0	15.9
5	1	1	0	15.3
6	1	1	0	13.7
7	1	0	1	12.6
8	1	0	1	13.3
9	1	0	1	16.6

Arbitrarily select one level to be the base level, then setup dummy variables for the remaining levels.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon, \quad x_1=1 \text{ if } T_2 \text{ and } x_2=1 \text{ if } T_3, \text{ zero otherwise.}$$

$$E(y(0,0)) = \beta_0 + \beta_1(0) + \beta_2(0) = \beta_0, \quad E(y(0,1)) = \beta_0 + \beta_1(1) + \beta_2(0) = \beta_0 + \beta_1, \quad E(y(1,1)) = \beta_0 + \beta_1(1) + \beta_2(1) = \beta_0 + \beta_1 + \beta_2, \text{ so}$$

$$\beta_0 = \mu_{T1}, \quad \beta_1 = \mu_{T2} - \mu_{T1}, \text{ and } \beta_2 = \mu_{T3} - \mu_{T1}.$$

b) Estimate the regression coefficients and MSE. $\hat{\beta} = (X'X)^{-1}X'y$

Step through code.

c) Compute the mean function at $x_0=[1,0,1]$, $x_0=[1,1,0]$, and $x_0=[1,1,1]$.

$$\hat{y}(x_0) = x_0 \hat{\beta}, \quad SE(\hat{y}_{x_0}) = \sqrt{MSE(x_0(X'X)^{-1}x_0')}$$

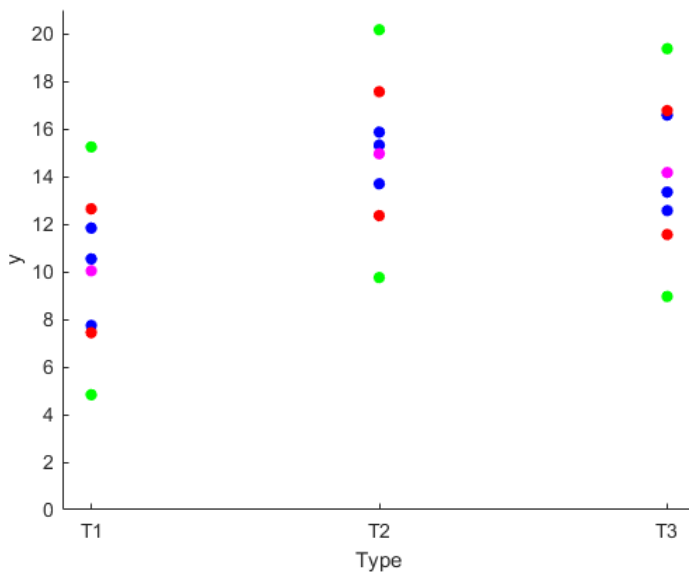
Step through code.

d) Compute the mean function CI at x_0 . $CI = \hat{y}(x_0) \pm t_{\alpha/2, n-k-1} SE(\hat{y}_{x_0})$

Step through code.

e) Compute the mean function PI at x_0 . $PI = \hat{y}(x_0) \pm t_{\alpha/2, n-k-1} \cdot \sqrt{MSE + (SE(\hat{y}_{x_0}))^2}$

Step through code.



MATH 2780 Chapter5 Worksheet

```
% set seed to same default value
rng('default')

% yi=b0+b1*x1i+b2i+ei, y=X*b+e
n1=3; n2=3; n3=3; n=n1+n2+n3;
x1=zeros(n1,1); ones(n2,1); zeros(n3,1)];
x2=zeros(n1,1); zeros(n2,1); ones(n3,1)];
X=[ones(n,1),x1,x2];
k=size(X,2)-1;
beta=[10;5;3];
sigma=1;

y=X*beta+randn(n,1);
[X,y]

X*beta

xx=[1*ones(n1,1);2*ones(n2,1);3*ones(n3,1)];
figure;
scatter(xx,y, 'filled', 'blue')
xlabel('Type'), ylabel('Amount')
set(gca, 'xtick', [1,2,3])
xticklabels({'T1', 'T2', 'T3'})
xlim([0.9,3.1]),ylim([0,21])

% estimate coefficients
W =inv(X'*X)
b=W*X'*y %10.0376, 4.9202, 4.1249
SSE=(y-X*b)'*(y-X*b);
MSE=SSE/(n-k-1)

mean(y(1:3)) %10.0376
mean(y(4:6)) %14.9578
mean(y(7:9)) %14.1625

mean(y(4:6))-mean(y(1:3)) % 4.9202

mean(y(7:9))-mean(y(1:3)) % 4.1249
```

```
% plot
x0=[1,0,0;1,1,0;1,0,1]
m=size(x0,1);
yfit = x0*b;
yfitCIL=zeros(m,1); yfitCIU=zeros(m,1);
yfitPIL=zeros(m,1); yfitPIU=zeros(m,1);
for i=1:m
    yfitCIL(i,1)=x0(i,:)*b...
        -tinv(1-0.05/2,n-k-1)...
        *sqrt(MSE* x0(i,:)*W*x0(i,:)');
    yfitCIU(i,1)=x0(i,:)*b...
        +tinv(1-0.05/2,n-k-1)...
        *sqrt(MSE* x0(i,:)*W*x0(i,:)');
    yfitPIL(i,1)=x0(i,:)*b...
        -tinv(1-0.05/2,n-k-1)...
        *sqrt(MSE*(1+x0(i,:)*W*x0(i,:)));
    yfitPIU(i,1)=x0(i,:)*b...
        +tinv(1-0.05/2,n-k-1)...
        *sqrt(MSE*(1+x0(i,:)*W*x0(i,:)));
end
figure;
scatter(xx,y, 'filled', 'blue')
hold on
scatter([1;2;3],yfit, 'filled', 'magenta')
scatter([1;2;3],yfitCIL, 'filled', 'red')
scatter([1;2;3],yfitCIU, 'filled', 'red')
scatter([1;2;3],yfitPIL, 'filled', 'green')
scatter([1;2;3],yfitPIU, 'filled', 'green')
xlabel('Type'), ylabel('y')
set(gca, 'xtick', [1;2;3])
xticklabels({'T1', 'T2', 'T3'})
xlim([0.9,3.1]),ylim([0,21])
```