Fitting the Additive Model in R

```
> o=lm(weightgain~diet+drug,data=d)
> model.matrix(o)
```

$X_R =$

```
   (Intercept) diet2 drug2 drug3
1            1     0     0     0
2            1     0     0     0
3            1     0     1     0
4            1     0     1     0
5            1     0     0     1
6            1     0     0     1
7            1     1     0     0
8            1     1     0     0
9            1     1     1     0
10           1     1     1     0
11           1     1     0     1
12           1     1     0     1
```

R: The $\hat{\beta}$ Vector

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

```
> #betahat vector:
>
> coef(o)
(Intercept)       diet2       drug2       drug3
  41.616667   -5.033333   -2.100000   -2.550000
```

$\hat{\mu}$ $\hat{\alpha}_2$ $\hat{\beta}_2$ $\hat{\beta}_3$

R: $\widehat{\text{Var}}(\hat{\beta})$ and Error Degrees of Freedom

```
> #Estimated variance of betahat:
>
> vcov(o)
```

$= \widehat{\text{VAR}}(\hat{\beta}_R) = \hat{\sigma}^2 (X_R' X_R)^{-1}$

```
            (Intercept)         diet2         drug2         drug3
(Intercept)   0.8186111 -4.093056e-01 -6.139583e-01 -6.139583e-01
diet2        -0.4093056  8.186111e-01 -6.759159e-17 -6.759159e-17
drug2        -0.6139583 -6.759159e-17  1.227917e+00  6.139583e-01
drug3        -0.6139583 -6.759159e-17  6.139583e-01  1.227917e+00
```

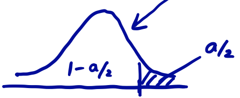
0

```
> #The degrees of freedom for error:
>
> o$df
[1] 8
```

$= n - \text{RANK}(X_R) = n - r = 12 - 4 = 8$

$c_{(k)}' \vee c_{(k)} \quad \forall\ k=1,\ldots,q \qquad c_{(i)}' \vee c_{(j)}$

R: A Function for Point and Interval Estimation




```
> estimate=function(lmout,C,a=0.05)
+ {
+   b=coef(lmout)
+   V=vcov(lmout)
+   df=lmout$df
+   Cb=C%*%b
+   se=sqrt(diag(C%*%V%*%t(C)))
+   tval=qt(1-a/2,df)
+   low=Cb-tval*se
+   up=Cb+tval*se
+   m=cbind(C,Cb,se,low,up)
+   dimnames(m)[[2]]=c(paste("c",1:ncol(C),sep=""),
+                   "estimate","se",
+                   paste(100*(1-a),"% Conf.",sep=""),
+                   "limits")
+   m
+ }
```

$= \hat{\beta}_R$

$= \hat{\sigma}^2 (X_R' X_R)^{-1}$

$= n - r$

$= C\hat{\beta}_R$

$= \sqrt{\hat{\sigma}^2 c_{(k)}' (X_R' X_R)^{-1} c_{(k)}}\ , \quad k=1,\ldots,q$

$c_{(k)}'\hat{\beta}_R - \text{tval}\sqrt{\hat{\sigma}^2 c_{(k)}'(X_R'X_R)^{-1}c_{(k)}}\ , \quad k=1,\ldots,q$

$c_{(k)}'\hat{\beta}_R + \text{tval}\sqrt{\hat{\sigma}^2 c_{(k)}'(X_R'X_R)^{-1}c_{(k)}}\ , \quad k=1,\ldots,q$

$$C = \begin{bmatrix} c_{(1)}' \\ c_{(2)}' \\ \vdots \\ c_{(q)}' \end{bmatrix} \qquad C\beta \qquad c_{(k)}'\beta$$

R: Entering a C Matrix

$\mu \quad \alpha_2 \quad \beta_2 \quad \beta_3$

```
> C=matrix(c(
+ 1, 0, 1/3, 1/3,
+ 1, 1, 1/3, 1/3,
+ 1, 1/2, 0, 0,
+ 1, 1/2, 1, 0,
+ 1, 1/2, 0, 1,
+ 0, -1, 0, 0,
+ 0, 0, -1,  0,
+ 0, 0,  0, -1,
+ 0, 0,  1, -1
+ ),ncol=4,byrow=T)
```

DIET 1 LSMEAN
DIET 2 LS MEAN
DRUG 1 LS MEAN
DRUG 2 LS MEAN
DRUG 3 LS MEAN
DIET 1 VS. DIET 2
DRUG 1 VS. DRUG 2
DRUG 1 VS. DRUG 3
DRUG 2 VS. DRUG 3

R: Function for Testing $H_0 : \boldsymbol{C\beta} = \boldsymbol{d}$

`> test=function(lmout,C,d=0){`

`+    b=coef(lmout)` $= \hat{\boldsymbol{\beta}}_R$

`+    V=vcov(lmout)` $= \hat{\sigma}^2 (X_R' X_R)^{-1}$

`+    dfn=nrow(C)` $= q$

`+    dfd=lmout$df` $= n - r$

`+    Cb.d=C%*%b-d` $= C\hat{\boldsymbol{\beta}}_R - \boldsymbol{d}$

`+    Fstat=drop(`

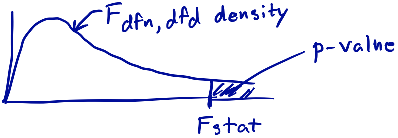
`         t(Cb.d)%*%solve(C%*%V%*%t(C))%*%Cb.d/dfn)`

$$\frac{(C\hat{\boldsymbol{\beta}}_R - \boldsymbol{d})' [C (X_R' X_R)^{-1} C']^{-1} (C\hat{\boldsymbol{\beta}}_R - \boldsymbol{d}) / q}{\hat{\sigma}^2}$$

`+    pvalue=1-pf(Fstat,dfn,dfd)`

`+    list(Fstat=Fstat,pvalue=pvalue)`

`+ }`



OLS Estimates of $m_j - m_{j^*}$ $\forall j \neq j^*$

$$\begin{bmatrix} \mu \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ m_1 \\ m_2 \\ m_3 \end{bmatrix}$$

```
> C=matrix(c(
+ 0,0,0,0,0,1,-1,0,
+ 0,0,0,0,0,1,0,-1,
+ 0,0,0,0,0,0,1,-1
+ ),byrow=T,nrow=3)
>
> Cbhat=C%*%bhat
> Cbhat
      [,1]
[1,]   2.5
[2,]   0.5
[3,]  -2.0
```

$m_1 - m_2$

$m_1 - m_3$

$m_2 - m_3$

Cbhat $= \boxed{C(X'X)^- X'} y$ ✓

$2.5 = \widehat{m_1 - m_2}$

$0.5 = \widehat{m_1 - m_3}$

$-2.0 = \widehat{m_2 - m_3}$

THESE ESTIMATES OBTAINED BY MULTIPLYING $C(X'X)^- X'$ TIMES y. LET'S LOOK AT $C(X'X)^- X'$ TO UNDERSTAND HOW DATA ARE USED TO ESTIMATE MOVIE EFFECT DIFFERENCES.

Response Weights for Estimation of $m_j - m_{j^*} \quad \forall\, j \neq j^*$

$C(X'X)^{-}X'$

```
> round(C%*%XXgi%*%t(X),2)
     [,1] [,2] [,3] [,4] [,5] [,6] [,7]
[1,]  0.5 -0.5    0    0    0  0.5 -0.5
[2,]  0.5 -0.5    1   -1    0  0.5 -0.5
[3,]  0.0  0.0    1   -1    0  0.0  0.0
```

		movie 1	movie 2	movie 3
customer	1	4	1	?
	2	?	3	5
	3	?	?	3
	4	3	1	?

(Best to make sense of rows 1 and 3 first and then row 2 follows.)

Alternative Analysis Using the R Full-Rank $\boldsymbol{X}$ Matrix

```
> customer=factor(c(1,1,2,2,3,4,4))
> movie=factor(c(1,2,2,3,3,1,2))
> d=data.frame(customer,movie,y)
>
> d
  customer movie y
1        1     1 4
2        1     2 1
3        2     2 3
4        2     3 5
5        3     3 3
6        4     1 3
7        4     2 1
```

The R Full-Rank $\boldsymbol{X}$ Matrix

μ	$\mu+m_2$	$\mu+m_3$
$\mu+c_2$	$\mu+c_2+m_2$	$\mu+c_2+m_3$
$\mu+c_3$	$\mu+c_3+m_2$	$\mu+c_3+m_3$
$\mu+c_4$	$\mu+c_4+m_2$	$\mu+c_4+m_3$

```
> o=lm(y~customer+movie,data=d)
> model.matrix(o)
```

	μ	c_2	c_3	c_4	m_2	m_3
	(Intercept)	customer2	customer3	customer4	movie2	movie3
1	1	0	0	0	0	0
2	1	0	0	0	1	0
3	1	1	0	0	1	0
4	1	1	0	0	0	1
5	1	0	1	0	0	1
6	1	0	0	1	0	0
7	1	0	0	1	1	0

$\hat{\beta}$, $\hat{y}$, and $y - \hat{y}$

$\hat{\beta}$

```
> coef(o)
(Intercept)   customer2   customer3   customer4     movie2     movie3
       3.75        1.75       -0.25       -0.50      -2.50      -0.50
```

$\hat{\mu}$ $\hat{c}_2$ $\hat{c}_3$ $\hat{c}_4$ $\hat{m}_2$ $\hat{m}_3$

```
> fitted(o)
   1    2    3    4    5    6    7
3.75 1.25 3.00 5.00 3.00 3.25 0.75
```

$P_x y = \hat{y}$

```
> resid(o)
            1             2     3     4     5
 2.500000e-01 -2.500000e-01     0     0     0
            6             7
-2.500000e-01  2.500000e-01
```

$(I - P_x)y = y - \hat{y}$ = VECTOR OF RESIDUALS

$\hat{\beta}$, $\hat{y}$, and $y - \hat{y}$

ALTERNATIVE CODE FOR OBTAINING SAME QUANTITIES

```
> o$coe
(Intercept)  customer2  customer3  customer4   movie2    movie3
       3.75       1.75      -0.25      -0.50    -2.50     -0.50

> o$fit
   1    2    3    4    5    6    7
3.75 1.25 3.00 5.00 3.00 3.25 0.75

> o$res
            1             2     3     4     5
 2.500000e-01 -2.500000e-01     0     0     0
            6             7
-2.500000e-01  2.500000e-01
```

OLS Estimates of $m_j - m_{j^*} \quad \forall\, j \neq j^*$

```
> -o$coe[5]
movie2
   2.5
> -o$coe[6]
movie3
   0.5
> o$coe[5]-o$coe[6]
movie2
    -2
```

$\widehat{m_1 - m_2} \rightarrow 0 - \hat{m}_2 = -\hat{m}_2$

$\widehat{m_1 - m_3} \rightarrow 0 - \hat{m}_3 = -\hat{m}_3$

$\widehat{m_2 - m_3} \rightarrow \hat{m}_2 - \hat{m}_3$

OLS Estimates of $m_j - m_{j^*}$ $\forall\, j \neq j^*$

```
> C=matrix(c(
+ 0,0,0,0,-1,0,
+ 0,0,0,0,0,-1,
+ 0,0,0,0,1,-1
+ ),byrow=T,nrow=3)
>
> C%*%o$coe
     [,1]
[1,]  2.5
[2,]  0.5
[3,] -2.0
```

$= C$

$= C\hat{\underset{\sim}{\beta}}$