

# Proposition de correction du TD sur les plans continus

Module de plan d'expériences - F. Husson - Institut Agro

## Exercice 1 : DÉTERMINATION D'UN OPTIMUM : LA RÉACTION DE MAILLARD

1. Le modèle classique comporte, pour chaque variable explicative, un effet linéaire et un effet quadratique, auxquels s'ajoute un terme d'interaction entre  $X_1$  et  $X_2$  :

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{11} X_1^2 + \beta_{22} X_2^2 + \beta_{12} X_1 X_2 + \varepsilon$$

À ce modèle, s'ajoutent les hypothèses usuelles sur les résidus  $\varepsilon$ .

2. Voici les lignes de code :

```
CR.rsm <- rsm(Y~S0(x1,x2),data=plan)
summary(CR.rsm)
```

Call:

```
rsm(formula = Y ~ S0(x1, x2), data = plan)
```

	Estimate	Std. Error	t value	Pr(> t )	
(Intercept)	1.2000e+01	1.0000e+00	12.0000	2.031e-05	***
x1	-6.0000e+00	7.0711e-01	-8.4853	0.0001465	***
x2	4.0000e+00	7.0711e-01	5.6569	0.0013107	**
x1:x2	2.7964e-15	1.0000e+00	0.0000	1.0000000	
x1^2	3.0000e+00	7.9057e-01	3.7947	0.0090232	**
x2^2	2.0000e+00	7.9057e-01	2.5298	0.0446904	*

---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Multiple R-squared: 0.953, Adjusted R-squared: 0.9138  
F-statistic: 24.33 on 5 and 6 DF, p-value: 0.0006455

Analysis of Variance Table

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
F0(x1, x2)	2	416.00	208.001	52.0002	0.0001623
TWI(x1, x2)	1	0.00	0.000	0.0000	1.0000000
PQ(x1, x2)	2	70.67	35.333	8.8333	0.0162946
Residuals	6	24.00	4.000		
Lack of fit	3	8.00	2.667	0.5000	0.7082086
Pure error	3	16.00	5.333		

Stationary point of response surface:

x1	x2
1.000001	-1.000004

Stationary point in original units:

ratio	pH
40.000011	4.999992

Eigenanalysis:

\$values

[1] 3 2

\$vectors

	[,1]	[,2]
x1	-1.000000e+00	1.398187e-15
x2	-1.398187e-15	-1.000000e+00

```
CR.rsm <- rsm(Y~F0(x1,x2)+PQ(x1,x2),data=plan)
summary(CR.rsm)
```

Call:

```
rsm(formula = Y ~ F0(x1, x2) + PQ(x1, x2), data = plan)
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	12.00000	0.92582	12.9615	3.785e-06 ***
x1	-6.00001	0.65465	-9.1652	3.789e-05 ***
x2	4.00002	0.65465	6.1101	0.0004862 ***
x1^2	3.00000	0.73193	4.0988	0.0045801 **
x2^2	2.00000	0.73193	2.7325	0.0292321 *

---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Multiple R-squared: 0.953, Adjusted R-squared: 0.9261  
F-statistic: 35.49 on 4 and 7 DF, p-value: 9.756e-05

Analysis of Variance Table

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
F0(x1, x2)	2	416.00	208.001	60.667	3.79e-05
PQ(x1, x2)	2	70.67	35.333	10.306	0.008204
Residuals	7	24.00	3.429		
Lack of fit	4	8.00	2.000	0.375	0.816497
Pure error	3	16.00	5.333		

Stationary point of response surface:

x1	x2
1.000001	-1.000004

Stationary point in original units:

ratio	pH
40.000011	4.999992

Eigenanalysis:

\$values

[1] 3 2

\$vectors

	[,1]	[,2]
x1	-1	0
x2	0	-1

L'ajustement global du modèle est excellent ( $R^2 = 0.953$ ). Il est hautement significatif (probabilité critique = 0.0006455).

On peut parler d'erreur pure en présence de répétitions, c'est-à-dire lorsque l'on dispose de plusieurs mesures faites dans les mêmes conditions expérimentales. Dans le plan composite centré, c'est le cas des points au centre et d'eux seuls. L'erreur pure est alors la variance de la réponse pour ces points. D'après le tableau, les valeurs de la réponse au centre sont {14, 10, 14, 10}. Pour ces 4 valeurs, la variance vaut 16/3 et correspond à l'erreur pure.

L'erreur (ou déficit) d'ajustement mesure l'écart entre les points moyens par condition expérimentale et les valeurs prédites correspondantes. Ainsi, figure ??, il y a deux observations au point  $x_i$  qui ont conduit à une réponse moyenne  $\bar{y}_i$ . L'erreur pure mesure l'écart entre les  $y_{ij}$  et  $\bar{y}_i$ ; l'erreur d'ajustement mesure l'écart entre  $\bar{y}_i$  et la valeur  $\hat{y}_i$  prédite par le modèle (en  $x_i$ ). En revanche, au point  $x_k$ , il n'y a qu'une seule observation. On ne peut formellement distinguer l'erreur pure et l'erreur d'ajustement pour cette condition expérimentale. En pratique, on peut inclure ce résidu dans l'erreur d'ajustement : ce faisant, on surestime cette erreur.

Finalement, à partir d'une variance résiduelle de 24 et d'une erreur pure de 16, on en déduit, par différence, l'erreur d'ajustement.

On observe nécessairement une erreur d'ajustement : les points moyens par condition expérimentale ne vérifient jamais exactement le modèle. Question : cet écart est-il du même ordre de grandeur que ce à quoi on peut s'attendre du seul fait de la variabilité résiduelle ? Si oui, on n'a aucune raison de ne pas faire confiance au modèle. Ou, au contraire, cet écart est-il « significativement » plus grand auquel cas le modèle est remis en question.

Pour cela on compare les deux carrés moyens (de l'erreur pure et du déficit d'ajustement) par un test de Fisher. La probabilité critique (0.816) indique un déficit d'ajustement non significatif : on n'a aucune raison de remettre le modèle en question.

3. Tous les coefficients sont significatifs excepté celui de l'interaction. On retient donc le modèle sans interaction. Il n'est pas utile de ré-estimer les coefficients car, avec un plan composite centré, l'interaction est orthogonale à tous les autres effets.

$$Y = 12 - 6X_1 + 4X_2 + 3X_1^2 + 2X_2^2$$

4. On annule les dérivées partielles de  $y$  :

$$dy/dx_1 = -6 + 6x_1 = 0 \quad dy/dx_2 = 4 + 4x_2 = 0$$

Soit  $x_1 = 1$  et  $x_2 = -1$  (ce résultat est donné dans le listage). Pour ces valeurs,  $y$  vaut : 7.

5. On voit qu'il s'agit d'un minimum

```
contour(CR.rsm,~x1+x2,image=TRUE)
persp(CR.rsm,~x1+x2,contours="colors")
```

## Exercice 2 : OPTIMISATION DE CONCENTRATION CELLULAIRE

1. On construit un plan composite centré. Ici,  $\alpha = 2$  et  $n_\alpha = 10$  et enfin, le nombre de points au centre est  $n_0 = 10$ .
2. 

```
plan <- ccd(basis=~A+B+C+D,generators = E~A*B*C*D,randomize=FALSE, alpha="rotatable",
coding=list(A~(Temp-35)/5, B~(pH-7)/1, C~(Vitesse-150)/50,
D~(taux-20)/10,E~(durée-3)/1))
plan
```

	run.order	std.order	Temp	pH	Vitesse	taux	durée	Block
1	1	1	30	6	100	10	4	1
2	2	2	40	6	100	10	2	1
3	3	3	30	8	100	10	2	1
4	4	4	40	8	100	10	4	1
5	5	5	30	6	200	10	2	1
6	6	6	40	6	200	10	4	1
7	7	7	30	8	200	10	4	1
8	8	8	40	8	200	10	2	1
9	9	9	30	6	100	30	2	1
10	10	10	40	6	100	30	4	1
11	11	11	30	8	100	30	4	1
12	12	12	40	8	100	30	2	1
13	13	13	30	6	200	30	4	1
14	14	14	40	6	200	30	2	1
15	15	15	30	8	200	30	2	1
16	16	16	40	8	200	30	4	1
17	17	17	35	7	150	20	3	1
18	18	18	35	7	150	20	3	1
19	19	19	35	7	150	20	3	1
20	20	20	35	7	150	20	3	1
21	1	1	25	7	150	20	3	2
22	2	2	45	7	150	20	3	2
23	3	3	35	5	150	20	3	2
24	4	4	35	9	150	20	3	2
25	5	5	35	7	50	20	3	2
26	6	6	35	7	250	20	3	2
27	7	7	35	7	150	0	3	2

28	8	8	35	7	150	40	3	2
29	9	9	35	7	150	20	1	2
30	10	10	35	7	150	20	5	2
31	11	11	35	7	150	20	3	2
32	12	12	35	7	150	20	3	2
33	13	13	35	7	150	20	3	2
34	14	14	35	7	150	20	3	2

3. Seul le pH est significatif, pas d'autres effets linéaires. Mais le test global sur les effets linéaires permet de considérer que tous les effets linéaires sont nuls. Donc on élimine tous les effets linéaires même si le pH seul semble significatif. Toutes les interactions sont négligeables, tous les effets quadratiques sont significatifs.
4. L'erreur d'ajustement du modèle montre qu'on ne peut pas rejeter l'hypothèse que le modèle a une mauvaise qualité d'ajustement.
5. le modèle est bon.

### Exercice 3 : PLAN DE DOEHLERT POUR DEUX FACTEURS : UNE BONNE GALETTE BRETONNE

```
plan <- matrix(c(0,0,1,0,-1,0,0.5,0.866,-0.5,-0.866,0.5,-0.866,-0.5,0.866),ncol=2,byrow=TRUE)
colnames(plan) <- c("X1","X2")
plan <- as.data.frame(plan)
X <- model.matrix(~X1+X2+I(X1^2)+I(X2^2)+I(X1*X2),data=plan)
cor(X[,-1])
```

	X1	X2	I(X1^2)	I(X2^2)	I(X1 * X2)
X1	1	0	0.0000000	0.0000000	0
X2	0	1	0.0000000	0.0000000	0
I(X1^2)	0	0	1.0000000	-0.5555556	0
I(X2^2)	0	0	-0.5555556	1.0000000	0
I(X1 * X2)	0	0	0.0000000	0.0000000	1

```
solve(t(X)%*%X)
```

	(Intercept)	X1	X2	I(X1^2)	I(X2^2)	I(X1 * X2)
(Intercept)	1.000000	0.0000000	0.0000000	-1.0000000	-1.0000587	0.0000000
X1	0.000000	0.3333333	0.0000000	0.0000000	0.0000000	0.0000000
X2	0.000000	0.0000000	0.3333529	0.0000000	0.0000000	0.0000000
I(X1^2)	-1.000000	0.0000000	0.0000000	1.5000000	0.8333822	0.0000000
I(X2^2)	-1.000059	0.0000000	0.0000000	0.8333822	1.5001760	0.0000000
I(X1 * X2)	0.000000	0.0000000	0.0000000	0.0000000	0.0000000	1.333412

1. Les confusions ou orthogonalités se lisent dans la matrice des corrélations entre effets et dans la matrice  $(X'X)^{-1}$ .
  - Les effets linéaires et l'interaction sont centrés et donc orthogonaux avec la constante. Les effets quadratiques ne sont pas centrés et sont donc confondus partiellement avec la constante.
  - Les effets linéaires et l'interaction sont orthogonaux entre eux et avec les effets quadratiques.
  - Les effets quadratiques ne sont pas orthogonaux entre eux.
2. Au coefficient  $\sigma^2$  près, dont nous ne tenons pas compte ici car il ne dépend pas du plan, les variances des estimateurs des effets se lisent sur la diagonale de la matrice  $(X'X)^{-1}$  : elles sont hétérogènes. Les variances des effets linéaires sont les plus faibles et identiques entre elles. Les variances des autres effets sont élevées (supérieures à 1). La variance de la constante est égale à 1. L'estimateur de la constante s'obtient en calculant la première ligne de la matrice  $(X'X)^{-1}X'$ . Cette ligne comporte un 1 en première position et des 0 ailleurs. La constante est donc estimée à partir du seul point au centre. D'où sa variance égale à 1.
3. 

```
library(rsm)
PCC <- ccd(2)
colnames(PCC)[3:4] <- c("X1","X2")
X <- model.matrix(~X1+X2+I(X1^2)+I(X2^2)+I(X1*X2),data=PCC)
solve(t(X)%*%X)
```

	(Intercept)	X1	X2	I(X1^2)	I(X2^2)	I(X1 * X2)
(Intercept)	0.1250	0.000	0.000	-0.0625	-0.0625	0.00
X1	0.0000	0.125	0.000	0.0000	0.0000	0.00
X2	0.0000	0.000	0.125	0.0000	0.0000	0.00
I(X1^2)	-0.0625	0.000	0.000	0.1250	0.0000	0.00
I(X2^2)	-0.0625	0.000	0.000	0.0000	0.1250	0.00
I(X1 * X2)	0.0000	0.000	0.000	0.0000	0.0000	0.25

Le plan de Doehlert comporte moins d'essais que le plan composite centré ; hors points au centre, dans le cas de deux facteurs : 6 essais au lieu de 8.

Les deux facteurs n'ont pas le même nombre de modalités ce qui peut s'avérer un avantage ou un inconvénient selon les situations.

Du point de vue de l'orthogonalité entre facteurs, les propriétés des deux plans sont comparables.

```
4. plan2 <- matrix(c(0,0,0,0,0,0,1,0,-1,0,0.5,0.866,-0.5,-0.866,0.5,-0.866,-0.5,0.866),
  ncol=2,byrow=TRUE)
colnames(plan2)=c("X1","X2")
plan2 <- as.data.frame(plan2)
X <- model.matrix(~X1+X2+I(X1^2)+I(X2^2)+I(X1*X2),data=plan2)
solve(t(X)%*%X)
```

	(Intercept)	X1	X2	I(X1^2)	I(X2^2)	I(X1 * X2)
(Intercept)	0.3333333	0.0000000	0.0000000	-0.3333333	-0.3333529	0.0000000
X1	0.0000000	0.3333333	0.0000000	0.0000000	0.0000000	0.0000000
X2	0.0000000	0.0000000	0.3333529	0.0000000	0.0000000	0.0000000
I(X1^2)	-0.3333333	0.0000000	0.0000000	0.8333333	0.1666764	0.0000000
I(X2^2)	-0.3333529	0.0000000	0.0000000	0.1666764	0.8334311	0.0000000
I(X1 * X2)	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	1.333412

Les variances des estimateurs de la constante et des effets quadratiques ont largement diminué. Les autres sont inchangées.

```
5. library(rsm)
PCC <- ccd(2,randomize=F,n0=1)
colnames(PCC)[3:4] <- c("X1","X2")
X <- model.matrix(~X1+X2+I(X1^2)+I(X2^2)+I(X1*X2),data=PCC)
solve(t(X)%*%X)
```

	(Intercept)	X1	X2	I(X1^2)	I(X2^2)	I(X1 * X2)
(Intercept)	0.50	0.000	0.000	-0.25000	-0.25000	0.00
X1	0.00	0.125	0.000	0.00000	0.00000	0.00
X2	0.00	0.000	0.125	0.00000	0.00000	0.00
I(X1^2)	-0.25	0.000	0.000	0.21875	0.09375	0.00
I(X2^2)	-0.25	0.000	0.000	0.09375	0.21875	0.00
I(X1 * X2)	0.00	0.000	0.000	0.00000	0.00000	0.25

Pour deux facteurs, le plan de Doehlert a des propriétés tout à fait comparables à celles du plan composite centré. Il nécessite 1 essai de moins, ce qui est un avantage dans notre expérience sur la pâte à galette.

## Exercice 4 : ETUDE DES PROPRIÉTÉS DES PLANS DE BOX-BENHKEN

1. On utilise `randomize=FALSE` pour voir comment le plan est construit

```
library(rsm)
Benhken <- bbd(3, n0=2, randomize=FALSE)
Benhken
```

	run.order	std.order	x1.as.is	x2.as.is	x3.as.is
1	1	1	-1	-1	0
2	2	2	1	-1	0
3	3	3	-1	1	0
4	4	4	1	1	0
5	5	5	-1	0	-1
6	6	6	1	0	-1
7	7	7	-1	0	1

8	8	8	1	0	1
9	9	9	0	-1	-1
10	10	10	0	1	-1
11	11	11	0	-1	1
12	12	12	0	1	1
13	13	13	0	0	0
14	14	14	0	0	0

2. Pour évaluer la qualité du plan, on calcule  $(X'X)^{-1}$ .

```
colnames(Benhken)[3:5] <- c("X1","X2","X3")
X <- model.matrix(~S0(X1,X2,X3),data=Benhken)
solve(t(X)%*%X)
```

	(Intercept)	x1	x2	x3	I(x1^2)	I(x2^2)	I(x3^2)	x1:x2	x1:x3	x2:x3
(Intercept)	0.50	0.000	0.000	0.000	-0.2500	-0.2500	-0.2500	0.00	0.00	0.00
x1	0.00	0.125	0.000	0.000	0.0000	0.0000	0.0000	0.00	0.00	0.00
x2	0.00	0.000	0.125	0.000	0.0000	0.0000	0.0000	0.00	0.00	0.00
x3	0.00	0.000	0.000	0.125	0.0000	0.0000	0.0000	0.00	0.00	0.00
I(x1^2)	-0.25	0.000	0.000	0.000	0.3125	0.0625	0.0625	0.00	0.00	0.00
I(x2^2)	-0.25	0.000	0.000	0.000	0.0625	0.3125	0.0625	0.00	0.00	0.00
I(x3^2)	-0.25	0.000	0.000	0.000	0.0625	0.0625	0.3125	0.00	0.00	0.00
x1:x2	0.00	0.000	0.000	0.000	0.0000	0.0000	0.0000	0.25	0.00	0.00
x1:x3	0.00	0.000	0.000	0.000	0.0000	0.0000	0.0000	0.00	0.25	0.00
x2:x3	0.00	0.000	0.000	0.000	0.0000	0.0000	0.0000	0.00	0.00	0.25

3. On peut voir que le nb de points au centre :

- n'influe pas sur la qualité de l'estimation des effets linéaires et des interactions
- joue sur la qualité de l'estimation de  $\beta_0$  (plus  $n_0$  grand, plus  $V(\beta_0)$  petite)
- les effets linéaires et les interactions sont toujours orthogonales à tous les autres effets
- les effets quadratiques sont orthogonaux entre eux si  $n_0 = 4$  seulement
- les effets quadratiques ne sont jamais orthogonaux avec la constante

```
Benhken <- bbd(3, n0=2)
colnames(Benhken)[3:5] <- c("X1","X2","X3")
X <- model.matrix(~S0(X1,X2,X3),data=Benhken)
solve(t(X)%*%X)
```

	(Intercept)	x1	x2	x3	I(x1^2)	I(x2^2)	I(x3^2)	x1:x2	x1:x3	x2:x3
(Intercept)	0.50	0.000	0.000	0.000	-0.2500	-0.2500	-0.2500	0.00	0.00	0.00
x1	0.00	0.125	0.000	0.000	0.0000	0.0000	0.0000	0.00	0.00	0.00
x2	0.00	0.000	0.125	0.000	0.0000	0.0000	0.0000	0.00	0.00	0.00
x3	0.00	0.000	0.000	0.125	0.0000	0.0000	0.0000	0.00	0.00	0.00
I(x1^2)	-0.25	0.000	0.000	0.000	0.3125	0.0625	0.0625	0.00	0.00	0.00
I(x2^2)	-0.25	0.000	0.000	0.000	0.0625	0.3125	0.0625	0.00	0.00	0.00
I(x3^2)	-0.25	0.000	0.000	0.000	0.0625	0.0625	0.3125	0.00	0.00	0.00
x1:x2	0.00	0.000	0.000	0.000	0.0000	0.0000	0.0000	0.25	0.00	0.00
x1:x3	0.00	0.000	0.000	0.000	0.0000	0.0000	0.0000	0.00	0.25	0.00
x2:x3	0.00	0.000	0.000	0.000	0.0000	0.0000	0.0000	0.00	0.00	0.25

```
Benhken <- bbd(3, n0=4)
colnames(Benhken)[3:5] <- c("X1","X2","X3")
X <- model.matrix(~S0(X1,X2,X3),data=Benhken)
solve(t(X)%*%X)
```

	(Intercept)	x1	x2	x3	I(x1^2)	I(x2^2)	I(x3^2)	x1:x2	x1:x3	x2:x3
(Intercept)	0.250	0.000	0.000	0.000	-0.125	-0.125	-0.125	0.00	0.00	0.00
x1	0.000	0.125	0.000	0.000	0.000	0.000	0.000	0.00	0.00	0.00
x2	0.000	0.000	0.125	0.000	0.000	0.000	0.000	0.00	0.00	0.00
x3	0.000	0.000	0.000	0.125	0.000	0.000	0.000	0.00	0.00	0.00
I(x1^2)	-0.125	0.000	0.000	0.000	0.250	0.000	0.000	0.00	0.00	0.00
I(x2^2)	-0.125	0.000	0.000	0.000	0.000	0.250	0.000	0.00	0.00	0.00
I(x3^2)	-0.125	0.000	0.000	0.000	0.000	0.000	0.250	0.00	0.00	0.00
x1:x2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.25	0.00	0.00

```
x1:x3      0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.00 0.25 0.00
x2:x3      0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.00 0.00 0.25
```

```
Benhken <- bbd(3, n0=10)
colnames(Benhken)[3:5] <- c("X1","X2","X3")
X <- model.matrix(~S0(X1,X2,X3),data=Benhken)
solve(t(X)%*%X)
```

```
      (Intercept)      x1      x2      x3 I(x1^2) I(x2^2) I(x3^2) x1:x2 x1:x3 x2:x3
(Intercept)      0.10 0.000 0.000 0.000 -0.0500 -0.0500 -0.0500 0.00 0.00 0.00
x1                0.00 0.125 0.000 0.000 0.0000 0.0000 0.0000 0.00 0.00 0.00
x2                0.00 0.000 0.125 0.000 0.0000 0.0000 0.0000 0.00 0.00 0.00
x3                0.00 0.000 0.000 0.125 0.0000 0.0000 0.0000 0.00 0.00 0.00
I(x1^2)          -0.05 0.000 0.000 0.000 0.2125 -0.0375 -0.0375 0.00 0.00 0.00
I(x2^2)          -0.05 0.000 0.000 0.000 -0.0375 0.2125 -0.0375 0.00 0.00 0.00
I(x3^2)          -0.05 0.000 0.000 0.000 -0.0375 -0.0375 0.2125 0.00 0.00 0.00
x1:x2            0.00 0.000 0.000 0.000 0.0000 0.0000 0.0000 0.25 0.00 0.00
x1:x3            0.00 0.000 0.000 0.000 0.0000 0.0000 0.0000 0.00 0.25 0.00
x2:x3            0.00 0.000 0.000 0.000 0.0000 0.0000 0.0000 0.00 0.00 0.25
```

On peut construire une fonction pour voir l'évolution (pour ceux qui veulent aller plus loin) :

```
library(rsm)
n0 <- 1:10
res <- NULL
for (i in 1:length(n0)){
  Benhken <- bbd(3, n0=n0[i])
  colnames(Benhken)[3:5] <- c("X1","X2","X3")
  X <- model.matrix(~S0(X1,X2,X3),data=Benhken)
  res <- c(res,solve(t(X)%*%X)[5,6])
}
plot(res~n0)
```

## Exercice 5 : CONSTRUCTION DE SURFACE DE RÉPONSE

1. On simule des données avec le modèle  $Y = X\beta + E$ .
2. On retrouve que tous les effets linéaires, quadratiques et interactions sont significatifs dans le modèle.

```
mod <- rsm (Y~S0(x1,x2,x3),data=PCC)
summary(mod)
```

Call:

```
rsm(formula = Y ~ S0(x1, x2, x3), data = PCC)
```

```
      Estimate Std. Error  t value Pr(>|t|)
(Intercept)  0.78253    0.36703   2.1320 0.0543590 .
x1           -0.82084    0.27172  -3.0209 0.0106458 *
x2           -2.79600    0.27172 -10.2898 2.628e-07 ***
x3           -0.91313    0.27172  -3.3605 0.0056688 **
x1:x2        0.17957    0.36792   0.4881 0.6342981
x1:x3        3.25634    0.36792   8.8508 1.317e-06 ***
x2:x3        0.78438    0.36792   2.1320 0.0543643 .
x1^2        -0.45583    0.23208  -1.9642 0.0731008 .
x2^2        -1.82168    0.23208  -7.8495 4.565e-06 ***
x3^2        -1.16240    0.23208  -5.0087 0.0003049 ***
---

```

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```
Multiple R-squared:  0.9603,      Adjusted R-squared:  0.9305
F-statistic: 32.23 on 9 and 12 DF,  p-value: 4.711e-07
```

Analysis of Variance Table

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
F0(x1, x2, x3)	3	136.770	45.590	42.0999	1.204e-06
TWI(x1, x2, x3)	3	90.010	30.003	27.7065	1.117e-05
PQ(x1, x2, x3)	3	87.352	29.117	26.8884	1.304e-05
Residuals	12	12.995	1.083		
Lack of fit	5	4.228	0.846	0.6753	0.6561
Pure error	7	8.766	1.252		

Stationary point of response surface:

x1	x2	x3
0.7921999	-0.6190089	0.5080007

Stationary point in original units:

x1.as.is	x2.as.is	x3.as.is
0.7921999	-0.6190089	0.5080007

Eigenanalysis:

eigen() decomposition

\$values

[1] 0.8939749 -1.7715969 -2.5622938

\$vectors

	[,1]	[,2]	[,3]
x1	0.7677584	0.2957298	0.5684109
x2	0.1163788	-0.9367234	0.3301594
x3	0.6300818	-0.1873316	-0.7535939

```
3. par(mfrow=c(2,3))  
contour(mod, ~x1+x2+x3, image=TRUE)  
contour(mod, ~x1+x2+x3, image=TRUE, at=canonical(mod)$xs)  
par(mar=c(0,0,0,0))  
persp(mod, ~x1+x2+x3, col=rainbow(50), contours="colors")  
persp(mod, ~x1+x2+x3, col=rainbow(50), contours="colors", at=c(x1=0.8, x2=-0.62, x3=0.51))
```

