

R と RStudio の使い方

芳賀敏郎 (2014) 医薬品開発のための統計解析 第2部 実験計画法
6 多因子実験
6.3 応答曲面計画

テキストと利用上の注意

●テキスト

芳賀敏郎（2011）医薬品開発のための統計解析

第2部 実験計画法 改訂版、サイエンティスト社、p.294

（サイトへアップすることに対して、サイエンティスト社の了解を得ています）

●Rによる解析事例を紹介

R スクリプトの出力結果を紹介します（tidyverse 系には次期バージョンで対応します）

R スクリプト（文字コードUTF-8に設定）を、このサイトから[ダウンロード](#)できます

R スクリプトを [Compile Report] することにより、Word または HTML で見ることができます

R と RStudio の設定と基本的な使い方は「[R と RStudio の使い方](#)」を参照してください

R の出力結果の見方は、テキストとそれを解説した [PDF ファイル](#) を参照してください

グラフ表示は、解析手段として、必要最小限の表現に止めています

●自己責任で利用

上記のことを理解した上で、自己責任により利用してください

第2部 実験計画法

- 1 因子実験 質的因子
 - 1.1 繰り返し数が等しい場合、1.2 繰り返し数が異なる場合
 - 1.3 多重比較、1.4 ばらつきを特性値とする実験
 - 1.5 ノンパラメトリック検定
- 量的因子
 - 2.1 直線関係の場合、2.2 非直線関係の場合
 - 2.3 ダミー変数による質的因子の効果の推定
- 乱塊法 3.1 質的因子の乱塊法、3.2 量的因子の乱塊法、3.3 欠測値のある場合
- 共分散分析 4.1 共分散分析の目的、4.2 解析手順、4.3 医薬品開発における共分散分析の例
- 2 因子実験 5.1 2 因子実験の基礎、5.2 質的因子×質的因子、5.3 質的因子×量的因子
- 5.4 質的因子×量的因子（変形）、5.5 量的因子×量的因子
- 多因子実験 6.1 多因子実験の基礎、6.2 スクリーニング計画、6.3 応答曲面計画**
- 変量モデルほか . . . 7.1 1 因子実験、7.2 枝分れ実験、7.3 乱塊法の拡張、7.4 経時データ、7.5 交差試験



応答曲面：Box-Behnken計画

A : x1
B : x2
C : x3

●表示6.3.3 Box-Behnken データの解析

スクリプトファイル：Green2-6-3a.R

方法

提供されている JMP ファイルの中から、
Box-Behnken データを読み込んで表示
(1 ~ 15 の順番を改変してある)

テキストでは、3 因子の列名は A、B、C
これ以降、コード化した列名は x1、x2、x3

提供されている JMP ファイルの内容
(1 ~ 15 の順番を改変)

##		A	B	C	y
##	1	-1	-1	0	1.66
##	2	1	-1	0	1.30
##	3	-1	1	0	1.08
##	4	1	1	0	1.54
##	5	-1	0	-1	1.45
##	6	1	0	-1	0.973
##	7	-1	0	1	1.40
##	8	1	0	1	1.79
##	9	0	-1	-1	1.41
##	10	0	1	-1	1.17
##	11	0	-1	1	1.75
##	12	0	1	1	1.51
##	13	0	0	0	1.03
##	14	0	0	0	0.967
##	15	0	0	0	1.06

●表示6.3.3 Box-Behnken データの解析

スクリプトファイル：Green2-6-3a.R

利用した関数：rsm::bbd

```
bbd_out <- bbd(k = 3, n0 = 3, randomize = FALSE,  
              coding = list(x1 ~ (A - 8) / 6,  
                            x2 ~ (B - 20)/10,  
                            x3 ~ (C - 7.8) / 0.7))
```

	濃度	温度	pH
x_1, x_2, x_3	A	B	C
-1	2	10	7.1
0	8	20	7.8
1	14	30	8.5

基準化 (コード化)

$$x_1 = \frac{A - 8}{6}$$

$$x_2 = \frac{B - 20}{10}$$

$$x_3 = \frac{C - 7.8}{0.7}$$

生成したBox-Behnken計画

#	run.order	A	B	C
## 1	1	2	10	7.8
## 2	2	14	10	7.8
## 3	3	2	30	7.8
## 4	4	14	30	7.8
## 5	5	2	20	7.1
## 6	6	14	20	7.1
## 7	7	2	20	8.5
## 8	8	14	20	8.5
## 9	9	8	10	7.1
## 10	10	8	30	7.1
## 11	11	8	10	8.5
## 12	12	8	30	8.5
## 13	13	8	20	7.8
## 14	14	8	20	7.8
## 15	15	8	20	7.8

応答曲面：Box-Behnken計画

● 表示6.3.3 Box-Behnken データの解析

スクリプトファイル：Green2-6-3a.r

利用した関数：rsm::bbd

k : 因子数
n0 : 中心点の繰り返し数

```
bbd_out <- bbd(k = 3, n0 = 3, randomize = FALSE,  
              coding = list(x1 ~ (A - 8) / 6,  
                             x2 ~ (B - 20) / 10,  
                             x3 ~ (C - 7.8) / 0.7))
```

x_1, x_2, x_3	濃度 A	温度 B	pH C
-1	2	10	7.1
0	8	20	7.8
1	14	30	8.5

基準化 (コード化)

$$x_1 = \frac{A - 8}{6}$$

$$x_2 = \frac{B - 20}{10}$$

$$x_3 = \frac{C - 7.8}{0.7}$$

生成したBox-Behnken計画

#	run.order	A	B	C
## 1	1	2	10	7.8
## 2	2	14	10	7.8
## 3	3	2	30	7.8
## 4	4	14	30	7.8
## 5	5	2	20	7.1
## 6	6	14	20	7.1
## 7	7	2	20	8.5
## 8	8	14	20	8.5
## 9	9	8	10	7.1
## 10	10	8	30	7.1
## 11	11	8	10	8.5
## 12	12	8	30	8.5
## 13	13	8	20	7.8
## 14	14	8	20	7.8
## 15	15	8	20	7.8

応答曲面：Box-Behnken計画

●表示6.3.3 Box-Behnken データの解析

スクリプトファイル：Green2-6-3a.R

利用した関数：rsm::bbd

as.data.frame(bbd_out)
により出力

```
bbd_out <- bbd(k = 3, n0 = 3, randomize = FALSE,  
              coding = list(x1 ~ (A - 8) / 6,  
                             x2 ~ (B - 20)/10,  
                             x3 ~ (C - 7.8) / 0.7))
```

	濃度	温度	pH
x_1, x_2, x_3	A	B	C
-1	2	10	7.1
0	8	20	7.8
1	14	30	8.5

基準化 (コード化)

$$x_1 = \frac{A - 8}{6}$$

$$x_2 = \frac{B - 20}{10}$$

$$x_3 = \frac{C - 7.8}{0.7}$$

生成したBox-Behnken計画

##	std.order	x1	x2	x3
## 1	1	-1	-1	0
## 2	2	1	-1	0
## 3	3	-1	1	0
## 4	4	1	1	0
## 5	5	-1	0	-1
## 6	6	1	0	-1
## 7	7	-1	0	1
## 8	8	1	0	1
## 9	9	0	-1	-1
## 10	10	0	1	-1
## 11	11	0	-1	1
## 12	12	0	1	1
## 13	13	0	0	0
## 14	14	0	0	0
## 15	15	0	0	0

●表示6.3.3 Box-Behnken データの解析

スクリプトファイル：Green2-6-3a.R

利用した関数：rsm::bbd、rsm::coded.data

```
df <- read_excel("Green2-6.xlsx", sheet = "6-bbd")
df1 <- coded.data(df, x1 ~ (A - 8) / 6,
                  x2 ~ (B - 20) / 10,
                  x3 ~ (C - 7.8) / 0.7)
```

	濃度	温度	pH
x_1, x_2, x_3	A	B	C
-1	2	10	7.1
0	8	20	7.8
1	14	30	8.5

基準化 (コード化)

$$x_1 = \frac{A - 8}{6}$$

$$x_2 = \frac{B - 20}{10}$$

$$x_3 = \frac{C - 7.8}{0.7}$$

Excel で提供される計画と観測値

##	A	B	C	y
## 1	2	10	7.8	1.659
## 2	14	10	7.8	1.298
## 3	2	30	7.8	1.075
## 4	14	30	7.8	1.536
## 5	2	20	7.1	1.453
## 6	14	20	7.1	0.973
## 7	2	20	8.5	1.398
## 8	14	20	8.5	1.790
## 9	8	10	7.1	1.413
## 10	8	30	7.1	1.174
## 11	8	10	8.5	1.754
## 12	8	30	8.5	1.507
## 13	8	20	7.8	1.029
## 14	8	20	7.8	0.967
## 15	8	20	7.8	1.060

応答曲面：Box-Behnken計画

p.232

●表示6.3.3 Box-Behnken データの解析

スクリプトファイル：Green2-6-3a.R

利用した関数

rsm::rsm、

rsm::SO、

summary

方法

rsm の使い方は、

[§5.5](#) 2 因子実験

(量的因子×量的因子)

を参照

[§5.5](#) 2 因子実験
(質的因子×質的因子)
を参照

A,B,C ではないことに注意

```
rsm_out <- rsm(y ~ SO(x1, x2, x3), data = df1)
summary(rsm_out)
```

##	Estimate	Std. Error	t value	Pr(> t)	
## (Intercept)	1.018667	0.028660	35.5435	3.318e-07	***
## x1	0.001500	0.017550	0.0855	0.9352061	
## x2	-0.104000	0.017550	-5.9258	0.0019517	**
## x3	0.179500	0.017550	10.2277	0.0001534	***
## x1:x2	0.205500	0.024820	8.2796	0.0004194	***
## x1:x3	0.218000	0.024820	8.7832	0.0003173	***
## x2:x3	-0.002000	0.024820	-0.0806	0.9389018	
## x1^2	0.157417	0.025833	6.0935	0.0017225	**
## x2^2	0.215917	0.025833	8.3580	0.0004012	***
## x3^2	0.227417	0.025833	8.8032	0.0003140	***

●表示6.3.3 Box-Behnken データの解析

スクリプトファイル：Green2-6-3a.R

利用した関数

rsm::rsm、
canonical、
predict

方法

滞留点 ([§5.5](#), p.201) から
最小値を計算

滞留点

滞留点での y の値
(最小値)

```
## Stationary point of response surface:
##           x1           x2           x3
## 0.31519838 0.08831199 -0.54533519
##
## Stationary point in original units:
##           A           B           C
## 9.891190 20.883120 7.418265
##
# 滞留点 (Stationary point) と最小値
opt_x1 <- canonical(rsm_out)$xs[1]
opt_x2 <- canonical(rsm_out)$xs[2]
opt_x3 <- canonical(rsm_out)$xs[3]
opt_xx <- data.frame(x1 = opt_x1, x2 = opt
_x2, x3 = opt_x3)
predict(rsm_out, opt_xx)
##           x1
## 0.965367
```

応答曲面：Box-Behnken計画

p.232

●表示6.3.3 Box-Behnken データの解析

スクリプトファイル：Green2-6-3a.R

利用した関数

rsm::rsm、rsm::SO、par、parsp

```
par(mfcol = c(1, 3))
```

```
par(oma = c(0, 0, 0, 0))
```

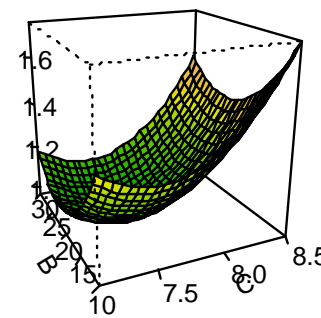
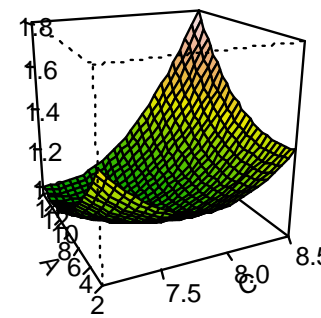
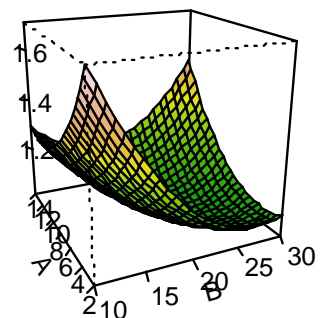
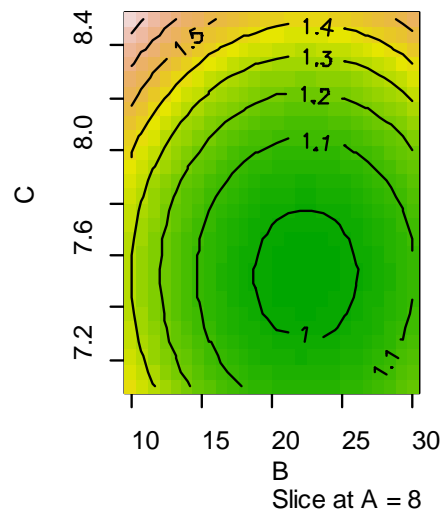
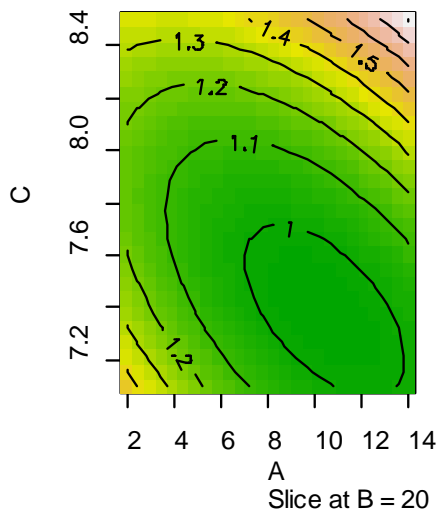
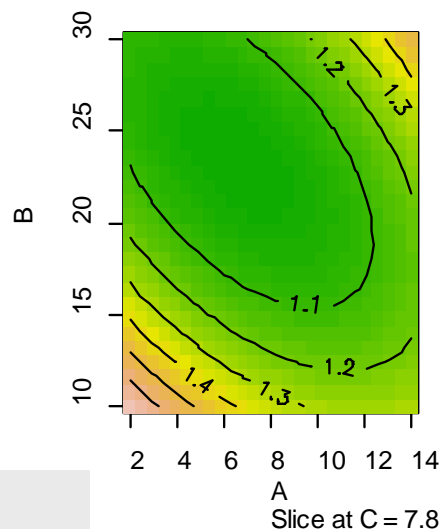
```
par(mar = c(5, 4, 1, 1))
```

```
contour(rsm_out, ~ x1 + x2 + x3, image = TRUE)
```

```
persp(rsm_out, x1 ~ x2, col = terrain.colors(50))
```

```
persp(rsm_out, x1 ~ x3, col = terrain.colors(50))
```

```
persp(rsm_out, x2 ~ x3, col = terrain.colors(50))
```





応答曲面：中心複合計画

- 中心複合計画（Central Composite Design,, CCD）と観測値

スクリプトファイル

Green2-6-3b.R

方法

提供されている JMP ファイルの中から、

中心複合計画（CCD）のデータを読み込んで表示

（1～16の順番を改変してある）

提供されている JMP I ファイルの内容

（1～16の順番を改変）

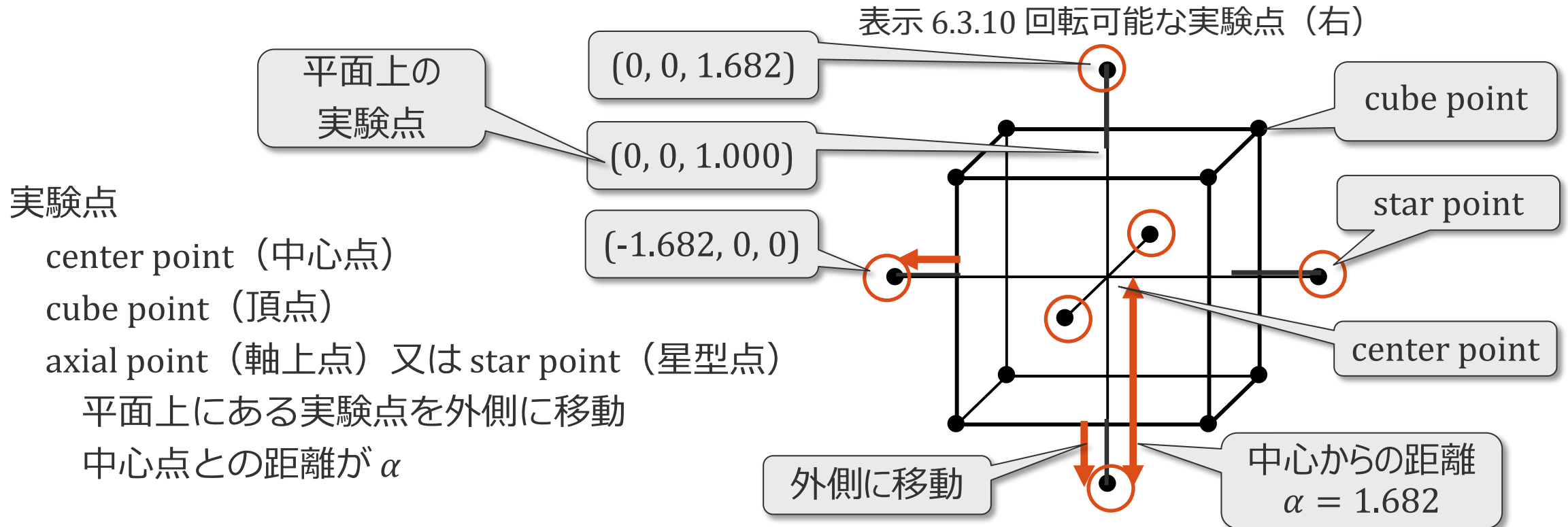
##	A	B	C	y
## 1	-1	-1	-1	1.887860
## 2	1	-1	-1	1.033515
## 3	-1	1	-1	1.365979
## 4	1	1	-1	1.259969
## 5	-1	-1	1	1.888449
## 6	1	-1	1	1.910722
## 7	-1	1	1	1.293220
## 8	1	1	1	2.163193
## 9	0	0	0	1.028984
## 10	-1	0	0	1.211358
## 11	1	0	0	1.249906
## 12	0	-1	0	1.387848
## 13	0	1	0	1.127830
## 14	0	0	-1	1.092206
## 15	0	0	1	1.427091
## 16	0	0	0	1.060342

●回転可能性のある中心複合計画

中心複合計画で回転可能性を持たせる

平面上の中心の点（6個）を外側に移動（表示6.3.10）

中心からの距離は $\alpha = \sqrt{3} = 1.732$ ではなく、 $\alpha = \sqrt[4]{8} = 1.682$ (回転可能性を満たす理論値)





応答曲面：中心複合計画

- 中心複合計画 (Central Composite Design, CCD) の生成

スクリプトファイル：Green2-6-3b.R

利用した関数：rsm::ccd

```
ccd_out <- ccd (basis = 3, n0 = c(1, 1),
               randomize = FALSE,
               alpha = "faces",
               oneblock = TRUE, coding =省略)
```

alpha : "orthogonal" 直交 = 1.789
 "rotatable" 回転可能 = $\sqrt[4]{8} = 1.682$
 "spherical" 球状 = $\sqrt{3} = 1.732$
 "faces" 平面上 = 1

bbd と同様に coding 引数の設定が可

cube 関数とstar 関数を組み合わせて生成することも可

##	run.order	A	B	C
## 1	1	2	10	7.1
## 2	2	14	10	7.1
## 3	3	2	30	7.1
## 4	4	14	30	7.1
## 5	5	2	10	8.5
## 6	6	14	10	8.5
## 7	7	2	30	8.5
## 8	8	14	30	8.5
## 9	9	8	20	7.8
## 10	1	2	20	7.8
## 11	2	14	20	7.8
## 12	3	8	10	7.8
## 13	4	8	30	7.8
## 14	5	8	20	7.1
## 15	6	8	20	8.5
## 16	7	8	20	7.8

● 中心複合計画 (Central Composite Design, CCD) の生成

スクリプトファイル：Green2-6-3b.R

利用した関数：rsm::ccd

```
ccd_out <- ccd (basis = 3, n0 = c(1, 1),
               randomize = FALSE,
               alpha = "faces",
               oneblock = TRUE, coding =省略)
```

alpha : "orthogonal"	直交	= 1.789
"rotatable"	回転可能	= $\sqrt[4]{8} = 1.682$
"spherical"	球状	= $\sqrt{3} = 1.732$
"faces"	平面上	= 1

bbd と同様に coding 引数の設定が可

cube 関数とstar 関数を組み合わせて生成することも可

as.data.faram(ccd_out)
により出力

##	run.order	x1	x2	x3
## 1	1	-1	-1	-1
## 2	2	1	-1	-1
## 3	3	-1	1	-1
## 4	4	1	1	-1
## 5	5	-1	-1	1
## 6	6	1	-1	1
## 7	7	-1	1	1
## 8	8	1	1	1
## 9	9	0	0	0
## 10	1	-1	0	0
## 11	2	1	0	0
## 12	3	0	-1	0
## 13	4	0	1	0
## 14	5	0	0	-1
## 15	6	0	0	1
## 16	7	0	0	0

応答曲面：中心複合計画

p.233

- 中心複合計画 (Central Composite Design, CCD) の生成

スクリプトファイル：Green2-6-3b.R

利用した関数：rsm::ccd

```
df <- read_excel("Green2-6.xlsx", sheet = "6-ccd")
df1 <- coded.data(df, x1 ~ (A - 8) / 6,
                  x2 ~ (B - 20) / 10,
                  x3 ~ (C - 7.8) / 0.7)
```

alpha : "orthogonal"	直交	= 1.789
"rotatable"	回転可能	= $\sqrt[4]{8} = 1.682$
"spherical"	球状	= $\sqrt{3} = 1.732$
"faces"	平面上	= 1

bbd と同様に coding 引数の設定が可

cube 関数とstar 関数を組み合わせて生成することも可

Excel で提供される計画と観測値

##	A	B	C	y
## 1	2	10	7.8	1.659
## 2	14	10	7.8	1.298
## 3	2	30	7.8	1.075
## 4	14	30	7.8	1.536
## 5	2	20	7.1	1.453
## 6	14	20	7.1	0.973
## 7	2	20	8.5	1.398
## 8	14	20	8.5	1.790
## 9	8	10	7.1	1.413
## 10	8	30	7.1	1.174
## 11	8	10	8.5	1.754
## 12	8	30	8.5	1.507
## 13	8	20	7.8	1.029
## 14	8	20	7.8	0.967
## 15	8	20	7.8	1.060

- 中心複合計画 (Central Composite Design, CCD) の解析

スクリプトファイル：Green2-6-3b.R

利用した関数

rsm::rsm、
summary

方法

rsm の使い方は、
[§5.5](#) 2 因子実験
を参照

[§5.5](#) 2 因子実験
(質的因子×質的因子)
を参照

```
rsm_out <- rsm(y ~ S0(x1, x2, x3), data = df1)
summary(rsm_out)
```

	Estimate	Std. Error	t value	Pr(> t)	
## (Intercept)	1.0607817	0.0205820	51.5394	3.580e-09	***
## x1	-0.0029562	0.0137477	-0.2150	0.8368663	
## x2	-0.0898203	0.0137477	-6.5335	0.0006141	***
## x3	0.2043145	0.0137477	14.8617	5.839e-06	***
## x1:x2	0.1995043	0.0153704	12.9798	1.288e-05	***
## x1*x3	0.2315750	0.0153704	15.0663	5.389e-06	***
## x2:x3	-0.0059164	0.0153704	-0.3849	0.7135718	
## x1^2	0.1617908	0.0267748	6.0426	0.0009293	***
## x2^2	0.1889981	0.0267748	7.0588	0.0004047	***
## x3^2	0.1908078	0.0267748	7.1264	0.0003842	***

停留点

- 中心複合計画 (Central Composite Design, CCD) の解析

スクリプトファイル

Green2-6-3b.R

利用した関数

rsm::canonical、

predict

方法

停留点 ([§5.5](#), p.201)

から最小値を計算

```
## Stationary point of response surface:
##           x1           x2           x3
##  1.0844749 -0.3523603 -1.1991672
##
## Stationary point in original units:
##           A           B           C
## 14.506850 16.476397  6.960583
# 停留点 (Stationary point) と最小値
opt_x1 <- canonical(rsm_out, threshold = 0.01)$xs[1]
opt_x2 <- canonical(rsm_out, threshold = 0.01)$xs[2]
opt_x3 <- canonical(rsm_out, threshold = 0.01)$xs[3]
opt_xx <- data.frame(x1 = opt_x1, x2 = opt_x2, x3 = opt_x3)
predict(rsm_out, opt_xx)

##           x1
## 0.9525578
```

$-1 < x < 1$
の範囲外

停留点での y の値
(最小値)

応答曲面：中心複合計画

p.233

●表示6.3.3 Box-Behnken データの解析

スクリプトファイル：Green2-6-3b.R

利用した関数

rsm::rsm、rsm::SO、par、parsp

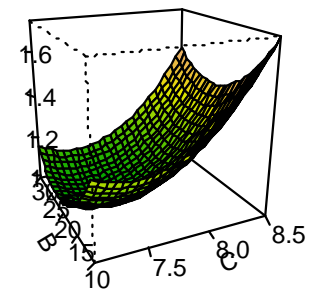
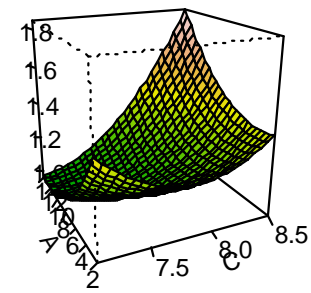
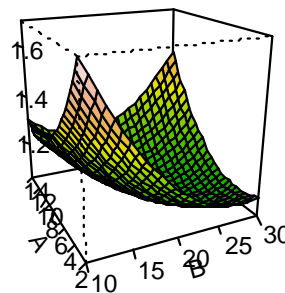
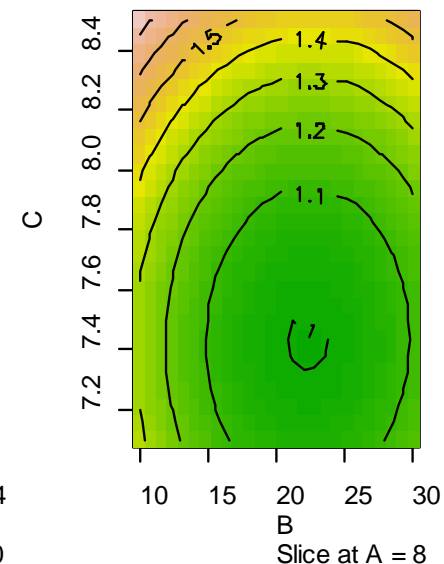
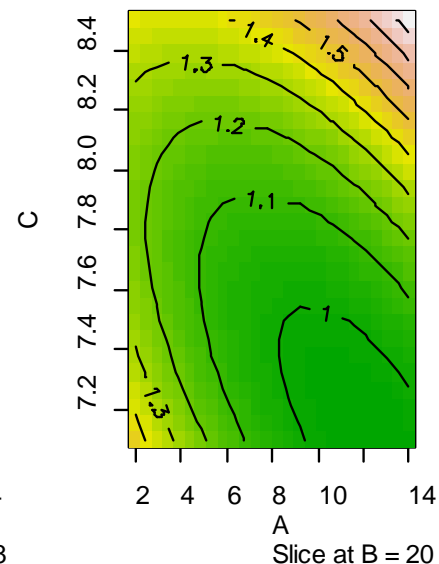
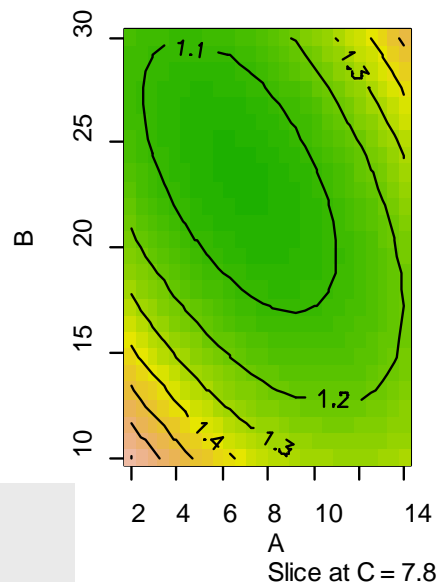
```
par(mfcol = c(1, 3))
par(oma = c(0, 0, 0, 0))
par(mar = c(5, 4, 1, 1))
```

```
contour(rsm_out, ~ x1 + x2 + x3, image = TRUE)
```

```
persp(rsm_out, x1 ~ x2, col = terrain.colors(50))
```

```
persp(rsm_out, x1 ~ x3, col = terrain.colors(50))
```

```
persp(rsm_out, x2 ~ x3, col = terrain.colors(50))
```





- 参考文献

Lenth RV (2009) “Response-Surface Methods in R,
Using rsm”, Journal of Statistical Software, 32(7), 1–17.

Updated to version 2.10.2, 3 September 2020

- 作成

片瀬雅彦

- 作成時期

2021年7月21日