Chapter 11

Experiments with Mixtures

Mixture designs are linear models parameterized to include the simplex constraint, typically by redefining the (interpretation of the) βs and modeling without an intercept.

11.1 Example 11.1, Table 11.1, p. 564

Read data and fit a first-order model and test $H_0 : \beta_1 = \beta_2 = \beta_3$, which we reject at $\alpha = 0.05$.

```
#### 11.1a
fn.data <- "http://statacumen.com/teach/RSM/data/RSM_EX_11-01.txt"
df.11.1 <- read.table(fn.data, header=TRUE)
df.11.1
```

```
##     x1  x2  x3    y
## 1  1.0 0.0 0.0 11.0
## 2  1.0 0.0 0.0 12.4
## 3  0.5 0.5 0.0 15.0
## 4  0.5 0.5 0.0 14.8
## 5  0.5 0.5 0.0 16.1
## 6  0.0 1.0 0.0  8.8
## 7  0.0 1.0 0.0 10.0
## 8  0.0 0.5 0.5 10.0
## 9  0.0 0.5 0.5  9.7
## 10 0.0 0.5 0.5 11.8
## 11 0.0 0.0 1.0 16.8
## 12 0.0 0.0 1.0 16.0
## 13 0.5 0.0 0.5 17.7
## 14 0.5 0.0 0.5 16.4
## 15 0.5 0.0 0.5 16.6
```

```
# fit intercept-only model (all betas equal)
lm.11.1.y.1 <- lm(y ~ 1, data = df.11.1)
lm.11.1.y.1$studres <- rstudent(lm.11.1.y.1)
summary(lm.11.1.y.1)
```

```
##
## Call:
## lm(formula = y ~ 1, data = df.11.1)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -4.74  -3.04   1.26   2.71   4.16
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   13.540      0.801    16.9    1e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.1 on 14 degrees of freedom
```

```
# fit first-order model
#   the ~ 0 in the formula indicates no intercept will be fit
lm.11.1.y.0FO <- lm(y ~ 0 + x1 + x2 + x3, data = df.11.1)
lm.11.1.y.0FO$studres <- rstudent(lm.11.1.y.0FO)
summary(lm.11.1.y.0FO)
```

```
##
```

```
## Call:
## lm(formula = y ~ 0 + x1 + x2 + x3, data = df.11.1)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.995 -1.813  0.205  1.755  3.687
##
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)
## x1    14.99       1.41   10.63  1.8e-07 ***
## x2     9.83       1.41    6.97  1.5e-05 ***
## x3    15.79       1.41   11.20  1.0e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.54 on 12 degrees of freedom
## Multiple R-squared:  0.973,Adjusted R-squared:  0.967
## F-statistic:  145 on 3 and 12 DF,  p-value: 1.07e-09
```

```
# test H_0: \beta_1 = \beta_2 = \beta_3
anova(lm.11.1.y.1, lm.11.1.y.0FO)
```

```
## Analysis of Variance Table
##
## Model 1: y ~ 1
## Model 2: y ~ 0 + x1 + x2 + x3
##   Res.Df   RSS Df Sum of Sq    F Pr(>F)
## 1     14 134.9
## 2     12  77.2  2      57.6 4.48  0.035 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Fit two-way interaction (quadratic) model and test $H_0 : \beta_{12} = \beta_{13} = \beta_{23} = 0$, which we reject at $\alpha = 0.05$.

```
#### 11.1b

# fit interaction model
lm.11.1.y.0TWI <- lm(y ~ 0 + x1 + x2 + x3 + x1:x2 + x1:x3 + x2:x3, data = df.11.1)
# externally Studentized residuals
lm.11.1.y.0TWI$studres <- rstudent(lm.11.1.y.0TWI)
summary(lm.11.1.y.0TWI)
```

```
##
## Call:
## lm(formula = y ~ 0 + x1 + x2 + x3 + x1:x2 + x1:x3 + x2:x3, data = df.11.1)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -0.80  -0.50  -0.30   0.65   1.30
##
## Coefficients:
##       Estimate Std. Error t value Pr(>|t|)
## x1      11.700      0.604   19.38  1.2e-08 ***
## x2       9.400      0.604   15.57  8.2e-08 ***
```

```
## x3      16.400      0.604   27.17  6.0e-10 ***
## x1:x2   19.000      2.608    7.28  4.6e-05 ***
## x1:x3   11.400      2.608    4.37   0.0018 **
## x2:x3   -9.600      2.608   -3.68   0.0051 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.854 on 9 degrees of freedom
## Multiple R-squared:  0.998,Adjusted R-squared:  0.996
## F-statistic:  658 on 6 and 9 DF,  p-value: 2.27e-11
```

```
# test H_0: \beta_{12} = \beta_{13} = \beta_{23} = 0
anova(lm.11.1.y.0FO, lm.11.1.y.0TWI)
```

```
## Analysis of Variance Table
##
## Model 1: y ~ 0 + x1 + x2 + x3
## Model 2: y ~ 0 + x1 + x2 + x3 + x1:x2 + x1:x3 + x2:x3
##   Res.Df  RSS Df Sum of Sq    F  Pr(>F)
## 1     12 77.2
## 2      9  6.6  3      70.7 32.3 3.8e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

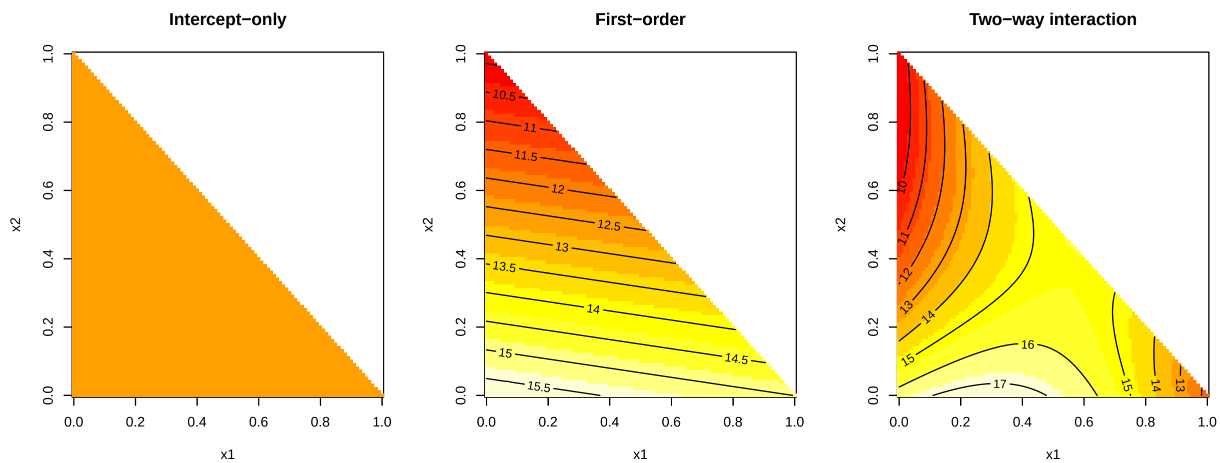
```
# test H_0: \beta_{12} = \beta_{13} = \beta_{23} = 0
#           AND
#           \beta_1 = \beta_2 = \beta_3
anova(lm.11.1.y.1, lm.11.1.y.0TWI)
```

```
## Analysis of Variance Table
##
## Model 1: y ~ 1
## Model 2: y ~ 0 + x1 + x2 + x3 + x1:x2 + x1:x3 + x2:x3
##   Res.Df   RSS Df Sum of Sq    F  Pr(>F)
## 1     14 134.9
## 2      9   6.6  5       128 35.2 1.2e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The two-way interaction (quadratic) model is preferred.
Plot contour plots of each model's RS.

```
# create a grid over x1 and x2
x <- seq(0, 1, by = 0.01)
x1 <- x; x2 <- x; # for plotting labels

grid.x123 <- expand.grid(x1 = x
                       , x2 = x
                       , x3 = NA)
grid.x123$x3 <- 1 - (grid.x123$x1 + grid.x123$x2)
# set any x3 < 0 to NA
grid.x123[(grid.x123$x3 < 0),] <- NA

# predictions for 3 models
predict.11.1.y.1    <- predict(lm.11.1.y.1    , newdata = grid.x123)
```

```
predict.11.1.y.0FO  <- predict(lm.11.1.y.0FO , newdata = grid.x123)
predict.11.1.y.0TWI <- predict(lm.11.1.y.0TWI, newdata = grid.x123)

# manually set predictions outside simplex to NA for intercept model
predict.11.1.y.1[is.na(grid.x123$x3)] <- NA

# plot contour plots for 3 mixture models
par(mfrow = c(1,3))
image  (x = x1, y = x2, z = matrix(predict.11.1.y.1    , nrow = length(x)), main = "Intercept-o
contour(x = x1, y = x2, z = matrix(predict.11.1.y.1    , nrow = length(x)), add = TRUE)
## Warning:  all z values are equal
image  (x = x1, y = x2, z = matrix(predict.11.1.y.0FO , nrow = length(x)), main = "First-order
contour(x = x1, y = x2, z = matrix(predict.11.1.y.0FO , nrow = length(x)), add = TRUE)
image  (x = x1, y = x2, z = matrix(predict.11.1.y.0TWI, nrow = length(x)), main = "Two-way int
contour(x = x1, y = x2, z = matrix(predict.11.1.y.0TWI, nrow = length(x)), add = TRUE)
```



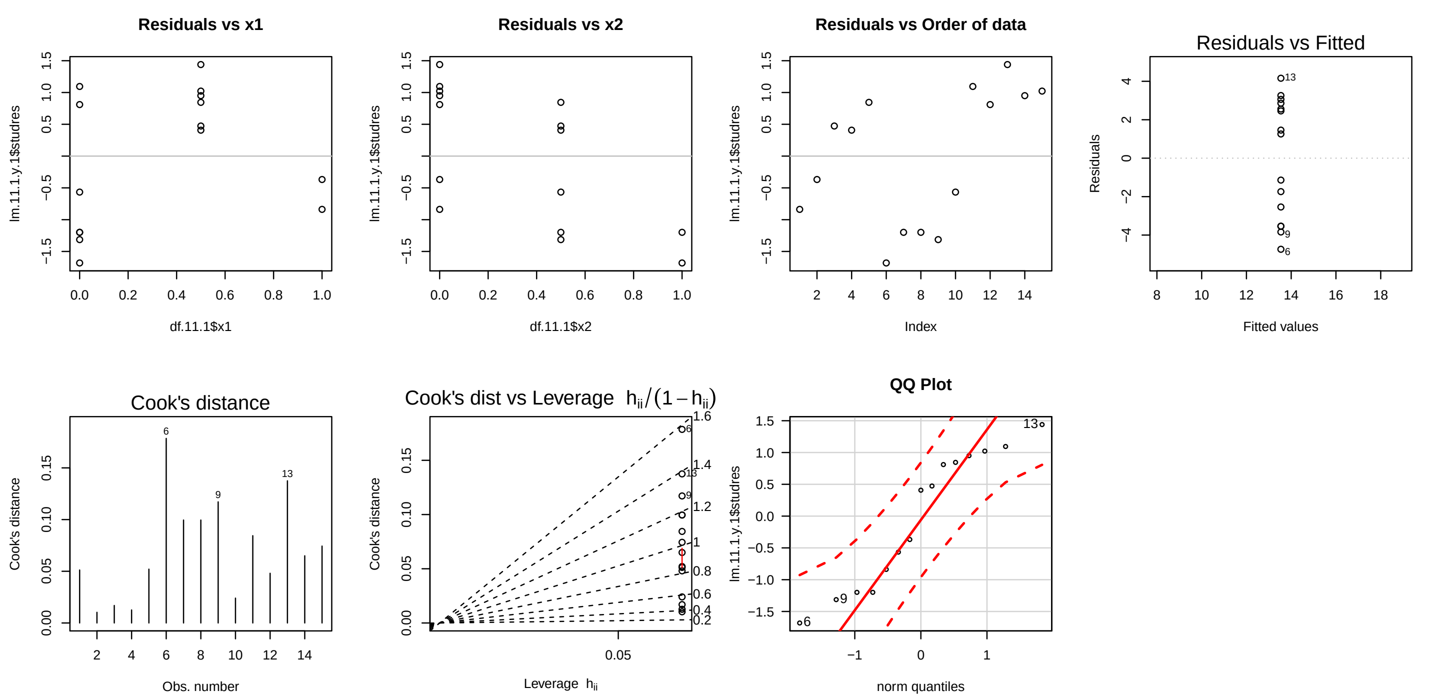
11.1.1 Residuals

Plot the residuals, and comment on the adequacy of each model. Only two-way interaction (quadratic) model does not have an issue with non-dependency on the x values.

Intercept-only model.

```
# plot diagnostics
par(mfrow=c(2,4))

plot(df.11.1$x1, lm.11.1.y.1$studres, main="Residuals vs x1")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(df.11.1$x2, lm.11.1.y.1$studres, main="Residuals vs x2")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# residuals vs order of data
```

```
plot(lm.11.1.y.1$studres, main="Residuals vs Order of data")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(lm.11.1.y.1, which = c(1,4,6))

# Normality of Residuals
library(car)
qqPlot(lm.11.1.y.1$studres, las = 1, id.n = 3, main="QQ Plot")

##  6 13  9
##  1 15  2

cooks.distance(lm.11.1.y.1)

##       1       2       3       4       5       6       7       8
## 0.05126 0.01033 0.01694 0.01261 0.05207 0.17850 0.09956 0.09956
##       9      10      11      12      13      14      15
## 0.11715 0.02405 0.08444 0.04808 0.13749 0.06499 0.07439
```



First-order model.

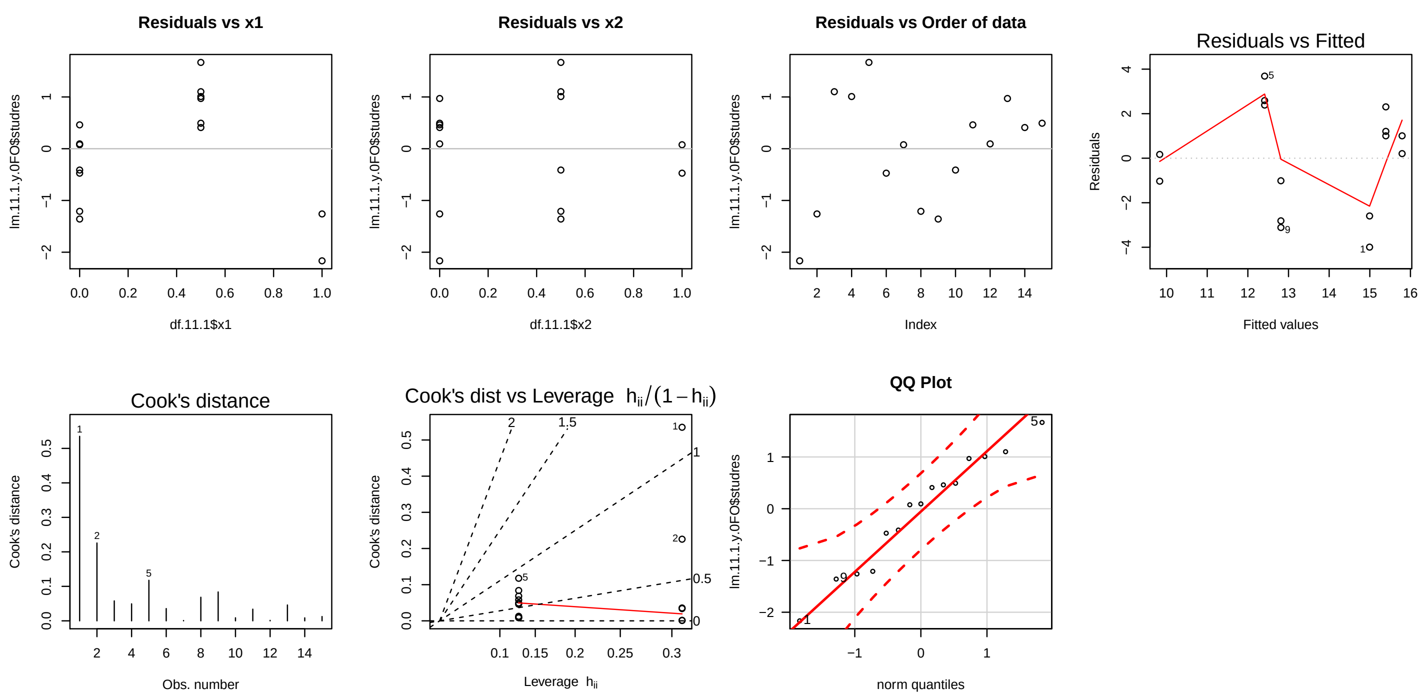
```
# plot diagnostics
par(mfrow=c(2,4))

plot(df.11.1$x1, lm.11.1.y.OFO$studres, main="Residuals vs x1")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(df.11.1$x2, lm.11.1.y.OFO$studres, main="Residuals vs x2")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# residuals vs order of data
plot(lm.11.1.y.OFO$studres, main="Residuals vs Order of data")
```

```
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(lm.11.1.y.0FO, which = c(1,4,6))

# Normality of Residuals
library(car)
qqPlot(lm.11.1.y.0FO$studres, las = 1, id.n = 3, main="QQ Plot")
```

```
##  1  5  9
##  1 15  2
```

```
cooks.distance(lm.11.1.y.0FO)
```

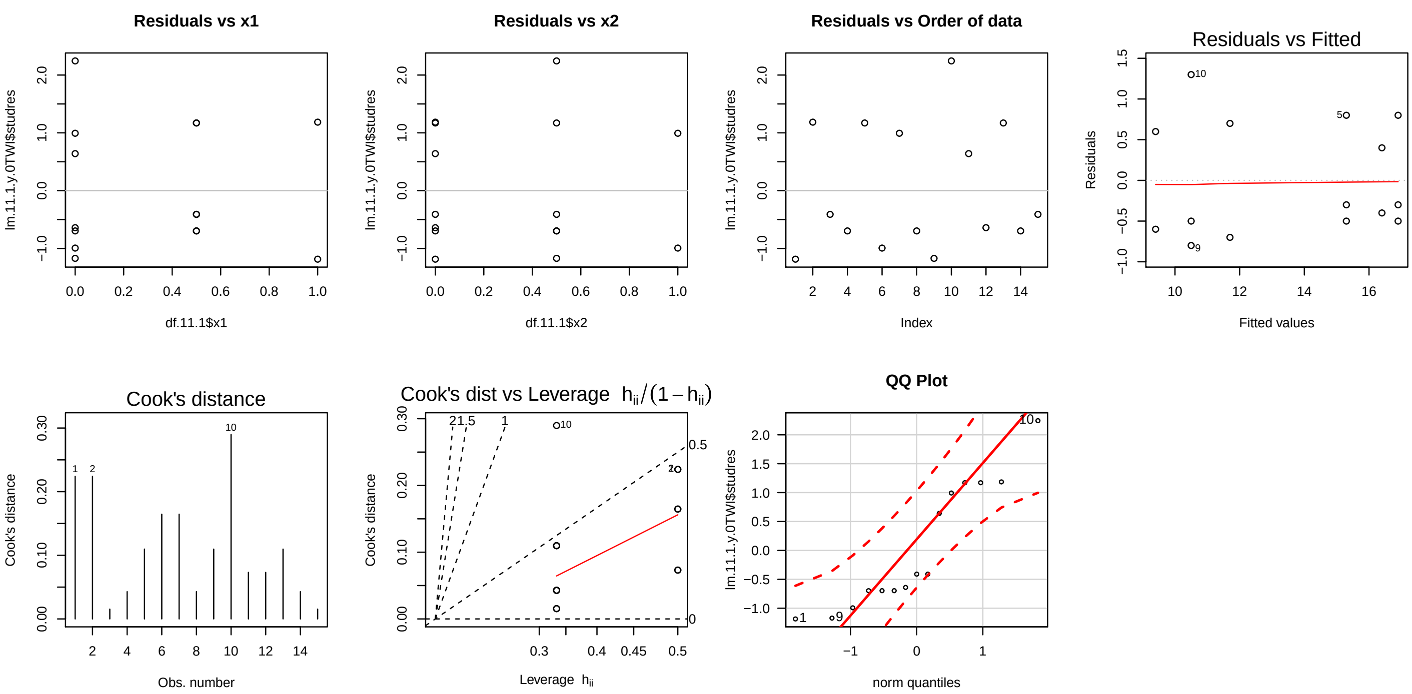
```
##         1         2         3         4         5         6         7
## 0.5351438 0.2257659 0.0579367 0.0493257 0.1176739 0.0356432 0.0009589
##         8         9        10        11        12        13        14
## 0.0684739 0.0838594 0.0088768 0.0339048 0.0014157 0.0460026 0.0087497
##        15
## 0.0125768
```



Two-way interaction model.

```
# plot diagnostics
par(mfrow=c(2,4))

plot(df.11.1$x1, lm.11.1.y.0TWI$studres, main="Residuals vs x1")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(df.11.1$x2, lm.11.1.y.0TWI$studres, main="Residuals vs x2")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# residuals vs order of data
```

```
plot(lm.11.1.y.OTWI$studres, main="Residuals vs Order of data")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(lm.11.1.y.OTWI, which = c(1,4,6))

# Normality of Residuals
library(car)
qqPlot(lm.11.1.y.OTWI$studres, las = 1, id.n = 3, main="QQ Plot")

## 10  1  9
## 15  1  2

cooks.distance(lm.11.1.y.OTWI)

##       1       2       3       4       5       6       7       8
## 0.22409 0.22409 0.01543 0.04287 0.10976 0.16463 0.16463 0.04287
##       9      10      11      12      13      14      15
## 0.10976 0.28982 0.07317 0.07317 0.10976 0.04287 0.01543
```



11.2 Example 11.2, Table 11.4, p. 571

Read data and assign special variables.

```
#### 11.2a
fn.data <- "http://statacumen.com/teach/RSM/data/RSM_EX_11-02.txt"
df.11.2 <- read.table(fn.data, header=TRUE)
# the data are coded such that any nonzero value is mean to be 1/value
for (i.col in 1:3) {
  ind <- (df.11.2[,i.col] > 0)
  df.11.2[ind, i.col] <- 1 / df.11.2[ind, i.col]
}
df.11.2

##        x1     x2     x3   y
## 1  1.0000 0.0000 0.0000 540
## 2  1.0000 0.0000 0.0000 560
## 3  0.0000 1.0000 0.0000 330
## 4  0.0000 1.0000 0.0000 350
## 5  0.0000 0.0000 1.0000 295
## 6  0.0000 0.0000 1.0000 260
## 7  0.5000 0.5000 0.0000 610
## 8  0.0000 0.5000 0.5000 330
## 9  0.5000 0.0000 0.5000 425
## 10 0.6667 0.1667 0.1667 710
## 11 0.1667 0.6667 0.1667 640
## 12 0.1667 0.1667 0.6667 460
## 13 0.3333 0.3333 0.3333 800
## 14 0.3333 0.3333 0.3333 850

# define some special variables
df.11.2$x12   <- df.11.2$x1  * df.11.2$x2
df.11.2$x13   <- df.11.2$x1  * df.11.2$x3
df.11.2$x23   <- df.11.2$x2  * df.11.2$x3
df.11.2$x1212 <- df.11.2$x12 * (df.11.2$x1 - df.11.2$x2)
df.11.2$x1313 <- df.11.2$x13 * (df.11.2$x1 - df.11.2$x3)
df.11.2$x2323 <- df.11.2$x23 * (df.11.2$x2 - df.11.2$x3)
df.11.2$x123  <- df.11.2$x12 * df.11.2$x3
df.11.2$group <- factor(100 * df.11.2$x1 + 10 * df.11.2$x2 + df.11.2$x3)
```

Next, we successively fit linear, quadratic, special cubic, and full cubic models. Root MSE values from models 1–4 show up in bottom part of Table 11.5, p. 573. Note that the R^2 values do not match up (excluding the intercept messes these up). The sequential Model Sum of Squares (top part of the table) is obtained by the hypothesis test output from `anova()`; the RSS is from model 4. In the Lack-of-Fit Tests (middle part of the table) the LOF SS and MS come from each individual Model 1–4 and the Pure Error comes from Model 5 (an ANOVA treating each factor level as a category). This is illustrated by the results in Table 11.6, p. 574, which come from Model 3.

```
## Model 0
# fit intercept-only model (all betas equal)
lm.11.2.y.mod0 <- lm(y ~ 1, data = df.11.2)
lm.11.2.y.mod0$studres <- rstudent(lm.11.2.y.mod0)
summary(lm.11.2.y.mod0)

##
## Call:
## lm(formula = y ~ 1, data = df.11.2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -251.4 -176.4  -11.4  121.1  338.6
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    511.4       51.3    9.97  1.9e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 192 on 13 degrees of freedom

## Model 1
# fit first-order model
#   the ~ 0 in the formula indicates no intercept will be fit
lm.11.2.y.mod1 <- lm(y ~ 0 + x1 + x2 + x3, data = df.11.2)
lm.11.2.y.mod1$studres <- rstudent(lm.11.2.y.mod1)
summary(lm.11.2.y.mod1)

##
## Call:
## lm(formula = y ~ 0 + x1 + x2 + x3, data = df.11.2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -155.4 -120.4  -80.7   89.3  338.6
##
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)
## x1      686        103    6.67  3.5e-05 ***
## x2      485        103    4.71  0.00064 ***
## x3      362        103    3.52  0.00480 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 177 on 11 degrees of freedom
## Multiple R-squared:  0.917,Adjusted R-squared:  0.894
## F-statistic: 40.3 on 3 and 11 DF,  p-value: 3.16e-06

# test H_0: \beta_1 = \beta_2 = \beta_3
anova(lm.11.2.y.mod0, lm.11.2.y.mod1)

## Analysis of Variance Table
##
## Model 1: y ~ 1
```

```
## Model 2: y ~ 0 + x1 + x2 + x3
##   Res.Df    RSS Df Sum of Sq    F Pr(>F)
## 1     13 478821
## 2     11 345066  2    133755 2.13   0.17
```

```
## Model 2
# fit two-way interaction (quadratic) model
#   the ~ 0 in the formula indicates no intercept will be fit
lm.11.2.y.mod2 <- lm(y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23, data = df.11.2)
lm.11.2.y.mod2$studres <- rstudent(lm.11.2.y.mod2)
summary(lm.11.2.y.mod2)
```

```
##
## Call:
## lm(formula = y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23, data = df.11.2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -157.7  -30.1   11.6   38.0  177.9
##
## Coefficients:
##     Estimate Std. Error t value Pr(>|t|)
## x1     534.6       83.3    6.41  0.00021 ***
## x2     329.2       83.3    3.95  0.00424 **
## x3     252.7       83.3    3.03  0.01625 *
## x12   1343.1      469.6    2.86  0.02114 *
## x13    644.5      469.6    1.37  0.20713
## x23    711.7      469.6    1.52  0.16810
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 120 on 8 degrees of freedom
## Multiple R-squared:  0.972,Adjusted R-squared:  0.951
## F-statistic: 46.4 on 6 and 8 DF,  p-value: 8.74e-06
```

```
# test H_0: \beta_{12} = \beta_{13} = \beta_{23} = 0
anova(lm.11.2.y.mod1, lm.11.2.y.mod2)
```

```
## Analysis of Variance Table
##
## Model 1: y ~ 0 + x1 + x2 + x3
## Model 2: y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23
##   Res.Df    RSS Df Sum of Sq    F Pr(>F)
## 1     11 345066
## 2      8 115702  3    229365 5.29  0.027 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# test H_0: \beta_{12} = \beta_{13} = \beta_{23} = 0
#            AND
#            \beta_1 = \beta_2 = \beta_3
anova(lm.11.2.y.mod0, lm.11.2.y.mod2)
```

```
## Analysis of Variance Table
##
## Model 1: y ~ 1
```

```
## Model 2: y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23
##   Res.Df    RSS Df Sum of Sq    F Pr(>F)
## 1     13 478821
## 2      8 115702  5    363120 5.02  0.022 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Model 3
# fit special cubic model
#   the ~ 0 in the formula indicates no intercept will be fit
lm.11.2.y.mod3 <- lm(y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23 + x123, data = df.11.2)
lm.11.2.y.mod3$studres <- rstudent(lm.11.2.y.mod3)
summary(lm.11.2.y.mod3)
```

```
##
## Call:
## lm(formula = y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23 + x123, data = df.11.2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -63.80 -10.11  -1.05  15.96  37.85
##
## Coefficients:
##      Estimate Std. Error t value Pr(>|t|)
## x1     550.20      23.22   23.69  6.1e-08 ***
## x2     344.72      23.22   14.84  1.5e-06 ***
## x3     268.29      23.22   11.55  8.2e-06 ***
## x12    689.54     146.51    4.71   0.0022 **
## x13     -9.03     146.51   -0.06   0.9526
## x23     58.11     146.51    0.40   0.7035
## x123  9243.33     940.85    9.82  2.4e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 33.4 on 7 degrees of freedom
## Multiple R-squared:  0.998,Adjusted R-squared:  0.996
## F-statistic:  528 on 7 and 7 DF,  p-value: 5.44e-09
```

```
# test H_0: \beta_{123} = 0
anova(lm.11.2.y.mod2, lm.11.2.y.mod3)
```

```
## Analysis of Variance Table
##
## Model 1: y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23
## Model 2: y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23 + x123
##   Res.Df    RSS Df Sum of Sq    F  Pr(>F)
## 1      8 115702
## 2      7   7824  1    107878 96.5 2.4e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# test H_0: \beta_{12} = \beta_{13} = \beta_{23} = \beta_{123} = 0
#           AND
#           \beta_1 = \beta_2 = \beta_3
anova(lm.11.2.y.mod0, lm.11.2.y.mod3)
```

```
## Analysis of Variance Table
##
## Model 1: y ~ 1
## Model 2: y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23 + x123
##   Res.Df    RSS Df Sum of Sq    F  Pr(>F)
## 1     13 478821
## 2      7   7824  6    470998 70.2 6.7e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Model 4
# fit full cubic model
#   the ~ 0 in the formula indicates no intercept will be fit
lm.11.2.y.mod4 <- lm(y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23 + x123 + x1212 + x1313, data = df.
lm.11.2.y.mod4$studres <- rstudent(lm.11.2.y.mod4)
summary(lm.11.2.y.mod4)
```

```
##
## Call:
## lm(formula = y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23 + x123 +
##     x1212 + x1313, data = df.11.2)
##
## Residuals:
##      1      2      3      4      5      6      7      8      9     10
##  -8.57  11.43  -8.57  11.43  18.93 -16.07   5.71   5.71   5.71 -17.13
##     11     12     13     14
## -17.13 -17.13 -12.15  37.85
##
## Coefficients:
##       Estimate Std. Error t value Pr(>|t|)
## x1       548.6       18.9   29.03  9.1e-07 ***
## x2       338.6       18.9   17.91  1.0e-05 ***
## x3       276.1       18.9   14.61  2.7e-05 ***
## x12      642.9      119.0    5.40   0.0029 **
## x13       27.9      119.0    0.23   0.8242
## x23       67.9      119.0    0.57   0.5933
## x123    9243.3      753.3   12.27  6.4e-05 ***
## x1212   -775.0      424.9   -1.82   0.1278
## x1313    980.0      424.9    2.31   0.0692 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26.8 on 5 degrees of freedom
## Multiple R-squared:  0.999,Adjusted R-squared:  0.998
## F-statistic:  641 on 9 and 5 DF,  p-value: 4.09e-07
```

```
# test H_0: \beta_{1212} = \beta_{1313} = 0
anova(lm.11.2.y.mod3, lm.11.2.y.mod4)
```

```
## Analysis of Variance Table
##
## Model 1: y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23 + x123
## Model 2: y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23 + x123 + x1212 + x1313
##   Res.Df  RSS Df Sum of Sq    F Pr(>F)
```

```
## 1      7 7824
## 2      5 3583  2      4241 2.96   0.14
# test H_0: \beta_{12} = \beta_{13} = \beta_{23} = \beta_{123} = \beta_{1212} = \beta_{1313}
#            AND
#            \beta_1 = \beta_2 = \beta_3
anova(lm.11.2.y.mod0, lm.11.2.y.mod4)
## Analysis of Variance Table
##
## Model 1: y ~ 1
## Model 2: y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23 + x123 + x1212 + x1313
##   Res.Df    RSS Df Sum of Sq    F  Pr(>F)
## 1     13 478821
## 2      5   3583  8    475239 82.9 6.9e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Model 5
# ANOVA model
#   the ~ 0 in the formula indicates no intercept will be fit
lm.11.2.y.mod5 <- lm(y ~ 0 + group, data = df.11.2)
lm.11.2.y.mod5$studres <- rstudent(lm.11.2.y.mod5)
summary(lm.11.2.y.mod5)
##
## Call:
## lm(formula = y ~ 0 + group, data = df.11.2)
##
## Residuals:
##     1     2     3     4     5     6     7     8     9    10    11
## -10.0  10.0 -10.0  10.0  17.5 -17.5   0.0   0.0   0.0   0.0   0.0
##    12    13    14
##   0.0 -25.0  25.0
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## group1       277.5       16.8    16.5  7.9e-05 ***
## group5.5     330.0       23.8    13.9  0.00016 ***
## group10      340.0       16.8    20.2  3.5e-05 ***
## group19      460.0       23.8    19.3  4.2e-05 ***
## group23.5    640.0       23.8    26.9  1.1e-05 ***
## group37      825.0       16.8    49.1  1.0e-06 ***
## group50.5    425.0       23.8    17.9  5.8e-05 ***
## group55      610.0       23.8    25.6  1.4e-05 ***
## group68.5    710.0       23.8    29.9  7.5e-06 ***
## group100     550.0       16.8    32.7  5.2e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 23.8 on 4 degrees of freedom
## Multiple R-squared:  0.999,Adjusted R-squared:  0.998
## F-statistic:  732 on 10 and 4 DF,  p-value: 4.47e-06
# test H_0: ANOVA vs full cubic
```

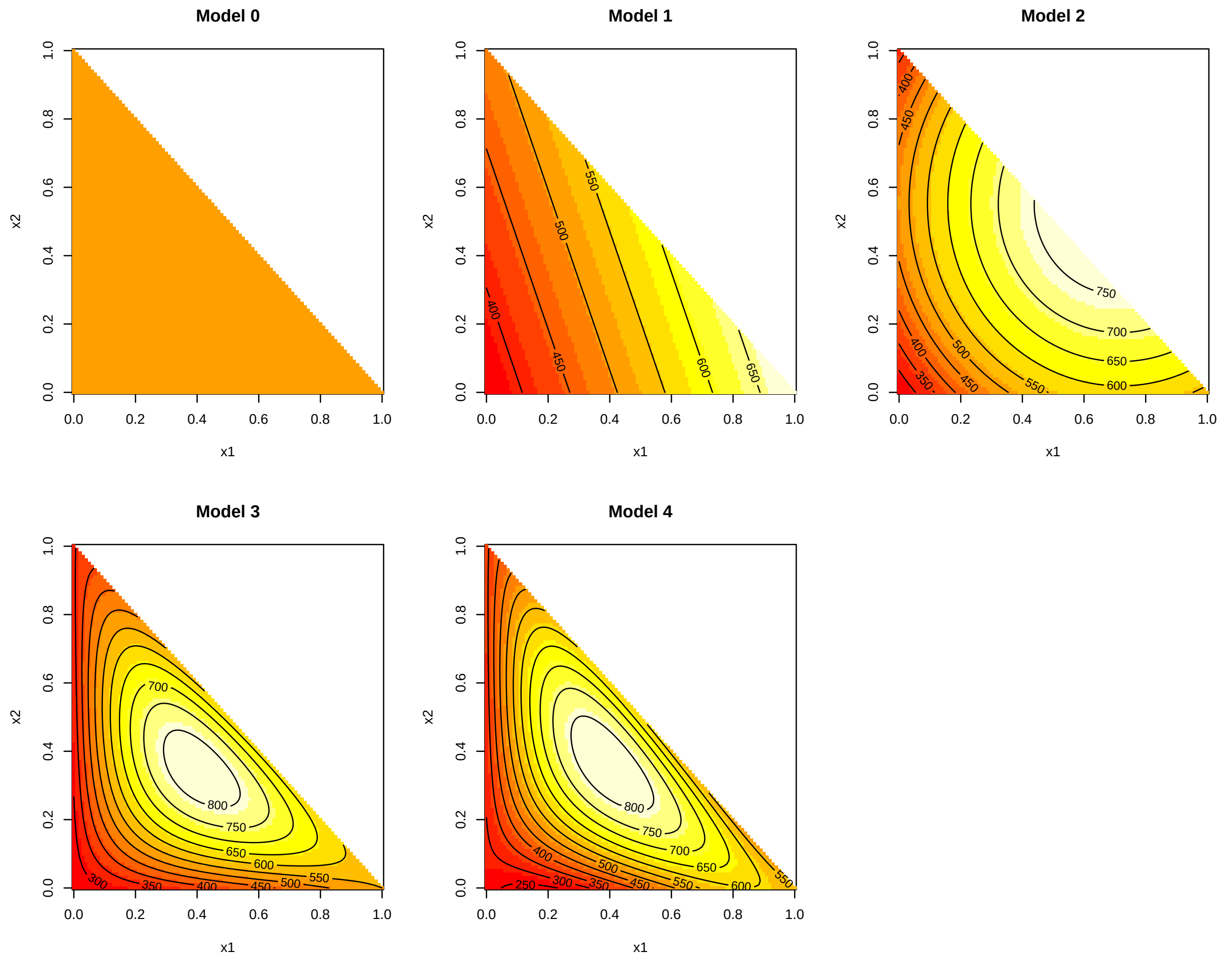
```
anova(lm.11.2.y.mod4, lm.11.2.y.mod5)
```

```
## Analysis of Variance Table
##
## Model 1: y ~ 0 + x1 + x2 + x3 + x12 + x13 + x23 + x123 + x1212 + x1313
## Model 2: y ~ 0 + group
##   Res.Df  RSS Df Sum of Sq    F Pr(>F)
## 1      5 3583
## 2      4 2263  1      1320 2.33    0.2
```

The results show that the special cubic is the model of choice.

Plot contour plots of each model's RS.

```
# add special variables to grid
grid.x123$x12   <- grid.x123$x1  * grid.x123$x2
grid.x123$x13   <- grid.x123$x1  * grid.x123$x3
grid.x123$x23   <- grid.x123$x2  * grid.x123$x3
grid.x123$x1212 <- grid.x123$x12 * (grid.x123$x1 - grid.x123$x2)
grid.x123$x1313 <- grid.x123$x13 * (grid.x123$x1 - grid.x123$x3)
grid.x123$x2323 <- grid.x123$x23 * (grid.x123$x2 - grid.x123$x3)
grid.x123$x123  <- grid.x123$x12 * grid.x123$x3
grid.x123$group <- factor(100 * grid.x123$x1 + 10 * grid.x123$x2 + grid.x123$x3)



# predictions for 6 models
predict.11.2.y.mod0 <- predict(lm.11.2.y.mod0, newdata = grid.x123)
predict.11.2.y.mod1 <- predict(lm.11.2.y.mod1, newdata = grid.x123)
predict.11.2.y.mod2 <- predict(lm.11.2.y.mod2, newdata = grid.x123)
predict.11.2.y.mod3 <- predict(lm.11.2.y.mod3, newdata = grid.x123)
predict.11.2.y.mod4 <- predict(lm.11.2.y.mod4, newdata = grid.x123)
#predict.11.2.y.mod5 <- predict(lm.11.2.y.mod5, newdata = grid.x123)

# manually set predictions outside simplex to NA for intercept model
predict.11.2.y.mod0[is.na(grid.x123$x3)] <- NA

# plot contour plots for 6 mixture models
par(mfrow = c(2,3))
image  (x = x1, y = x2, z = matrix(predict.11.2.y.mod0, nrow = length(x)), main = "Model 0")
contour(x = x1, y = x2, z = matrix(predict.11.2.y.mod0, nrow = length(x)), add = TRUE)
## Warning:  all z values are equal
image  (x = x1, y = x2, z = matrix(predict.11.2.y.mod1, nrow = length(x)), main = "Model 1")
contour(x = x1, y = x2, z = matrix(predict.11.2.y.mod1, nrow = length(x)), add = TRUE)
image  (x = x1, y = x2, z = matrix(predict.11.2.y.mod2, nrow = length(x)), main = "Model 2")
contour(x = x1, y = x2, z = matrix(predict.11.2.y.mod2, nrow = length(x)), add = TRUE)
image  (x = x1, y = x2, z = matrix(predict.11.2.y.mod3, nrow = length(x)), main = "Model 3")
contour(x = x1, y = x2, z = matrix(predict.11.2.y.mod3, nrow = length(x)), add = TRUE)
image  (x = x1, y = x2, z = matrix(predict.11.2.y.mod4, nrow = length(x)), main = "Model 4")
contour(x = x1, y = x2, z = matrix(predict.11.2.y.mod4, nrow = length(x)), add = TRUE)
#image  (x = x1, y = x2, z = matrix(predict.11.2.y.mod5, nrow = length(x)), main = "Model 5'
#contour(x = x1, y = x2, z = matrix(predict.11.2.y.mod5, nrow = length(x)), add = TRUE)
```



11.2.1 Residuals

Residual plots for the preferred special cubic model indicates observation 12 may be influential.

```
# plot diagnistics
par(mfrow=c(2,4))

plot(df.11.2$x1, lm.11.2.y.mod3$studres, main="Residuals vs x1")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(df.11.2$x2, lm.11.2.y.mod3$studres, main="Residuals vs x2")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# residuals vs order of data
plot(lm.11.2.y.mod3$studres, main="Residuals vs Order of data")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(lm.11.2.y.mod3, which = c(1,4,6))

# Normality of Residuals
```

```
library(car)
qqPlot(lm.11.2.y.mod3$studres, las = 1, id.n = 3, main="QQ Plot")

## 12  9 14
##  1 14 13

cooks.distance(lm.11.2.y.mod3)

##        1        2        3        4        5        6        7
## 0.023968 0.022129 0.049944 0.006415 0.164312 0.015852 1.452808
##        8        9       10       11       12       13       14
## 1.204219 4.861999 0.002273 0.016374 0.170419 0.018122 0.175761
```