

# Model Robust Designs and Analysis with `daewr`

## Abstract

Plackett-Burman and other two-level and three-level designs with complex aliasing are useful for determining the subset of important factors and interactions in screening experiments. Complex aliasing means that main effects and quadratic effects (for three-level designs) are only partially confounded with two-factor interactions. The number of runs required by designs with complex aliasing is similar to the minimal number of runs required by resolution III fractional factorial designs. However, in resolution III fractional factorial designs, main effects are completely confounded with two-factor interactions and cannot be separated without further experiments.

On the other hand, models fit to designs with complex aliasing can contain some main effects and some interactions since they are not completely confounded. Therefore, an appropriate subset of main effects and interactions can be found using forward stepwise regression. Hamada and Wu(1992) and Jones and Nachtsheim(2011) suggested using a forward stepwise regression that enforces model hierarchy. In other words, when an interaction or quadratic term is the next term to enter the model, the parent linear main effect(s) are automatically included.

The R package `daewr`(Lawson(2020)) contains functions for retrieving designs with complex aliasing such as Plackett-Burman designs(1946), Definitive Screening designs(Jones and Nachtsheim(2011)), Model Robust designs(Li and Nachtsheim(2000)), and Alternative Screening designs in 16 runs(Jones and Montgomery(2010)). The package also contains functions for forward stepwise regression that preserves model hierarchy. This vignette illustrates the use of these functions, and the example analyses include data from previously published sources.

## Introduction

Discovered during World War II, Plackett-Burman designs, include only two-levels of each factor and are available in run sizes that are multiples of four (i.e.,  $k=4, 8, 12, \dots$  etc.), and they are very efficient and orthogonal in  $k-1$  factors. Until recently, these designs were considered main-effect plans that were useful for screening main effects but not for entertaining interaction effects. However, Lin and Draper(1992) and Wang and Wu(1995) showed that two-factor interactions are only partially confounded with main effects in Plackett-Burman and similar designs. Therefore, a subset of main effects and some two-factor interactions could be entertained in a regression model using data obtained from an experiment using one of these designs. Designs related to the Plackett-Burman designs such as the 8, 12 and 16 run Model Robust designs(Li and Nachtsheim 2000), and the 16 run Alternative Screening designs(Jones and Montgomery 2010) have similar partial confounding between two-level main effects and two-factor interactions. The partial confounding of main effects with two-factor interactions allows fitting models with a variety of subsets of the main effects and two-factor interactions. For this reason, all of these designs are called model robust since they do not restrict the model that can be fit to the data.

Jones and Nachtsheim(2011) proposed a new class of designs they called Definitive Screening designs. These designs include three-level factors that would be useful in detecting curvilinear relationships, and they require only  $2k+1$  runs for  $k$  factors. The linear main effects are completely orthogonal to the quadratic effects, and linear-by-linear interactions. Linear-by-linear two-factor interactions are partially confounded with quadratic effects and other two-factor interactions. Jones and Nachtsheim(2013) also proposed column-augmented Definitive Screening designs that include both three-level quantitative factors and additional two-level categorical factors. The linear main effects in these designs are completely uncorrelated with linear-by-linear two-factor interactions. Two-factor interactions are partially confounded with other two-factor

interactions. Thus, Definitive Screening designs and column-augmented Definitive Screening designs can also be classified as model robust.

Lin(1999) proposed analyzing the data collected from a Plackett-Burman design by forward stepwise regression with the candidate variables being all main effects and two-factor interactions. This simple method of analysis is also useful for other two-level designs with complex aliasing and for the analysis of data from Definitive Screening designs. However, the obvious objection to a model derived from forward stepwise regression is that it may contain an interaction without either parent main effect involved in the interaction. Usually significant interactions occur between factors where at least one of the main effects is significant. This has been described as *the effect heredity principle* by Hamada and Wu(1992), and it was found to be violated less than 1% of the time in Li et. al.(2006)'s study of 113 published factorial experiments. With this fact in mind, Hamada and Wu(1992) and Jones and Nachtsheim(2011) both proposed forward selection procedures that ensures model hierarchy. Jones and Nachtsheim's implementation is available in the commercial software JMP (JMP®, Version 14 Pro, SAS Institute, Cary, NC 1989-2019) with the COMBINE feature in the forward stepwise regression.

## R Packages for Design and Analysis of Model Robust Designs

Plackett-Burman designs do not have a generator or defining relation like regular fractional factorial designs. The 12, 20, and 24 run Plackett-Burman designs can be created by cyclically rotating the factor levels in the first run to create the factor levels for the next  $n-2$  runs and including a  $n$ th run with constant levels for all factors. In that way, the factor levels for the first run become the generator for the Plackett-Burman Design. The first run factor levels shown in the book *Design and Analysis of Experiments with R*(Lawson 2015) are used by the `pbDesign()` function in the R package `qualityTools`(Roth 2016), the `pb()` function in the R package `FrF2`(Groemping 2020), and the `PBDes()` function in the R package `daewr`(Lawson 2020) to create Plackett-Burman designs. The `OptPB` function in the R package `daewr` also uses the same set of first run factor levels but selects the columns to be used in the design to minimize the absolute correlation between main effects and two-factor interactions in order to make the design more model robust.

The `PBDes()` and `OptPB()` functions in the R package `daewr` return a data frame with numerical columns that are appropriate for regression analysis of the resulting data. Alternatively the `pbDesign()` function in the R package `qualityTools`, and the `pb()` function in the R package `FrF2` package return a data frames containing factor variables. Factor variables are more appropriate for an ANOVA of main effects.

Although different generators (or sets of factor levels for the first run) can be used to create a Plackett-Burman design, the choice cannot be arbitrary, and the functions in the R packages named above should be used whenever generating Plackett-Burman designs with R.

The `AltScreen()` function in the R package `daewr` package can be used to recall 16 run Alternative Screening designs from Jones and Montgomery(2010). The `ModelRobust()` function in the R package `daewr` can be used to recall the 8, 12 and 16 run Model Robust designs described by Li and Nachtsheim(2000). The `DefScreen()` function in the R package `daewr` can be used to recall both the Definitive Screening designs defined by Jones and Nachtsheim(2011) and the column-augmented Definitive Screening designs defined by Jones and Nachtsheim(2013). All of these functions return a data frame of numerical values that are appropriate for regression analysis of the data.

The `HierAFS()` function in the R package `daewr` package performs a forward stepwise regression enforcing model hierarchy. This function calls the `ihstep()` and `fhstep()` functions in the same package. They can be called individually to see the detailed output of the `lm()` function at each step of the forward selection.

## Illustrations

### Example 1:

Seven factors were studied using a Plackett-Burman design to reduce the variability of the thickness in the copperplating of the drill holes used to connect various layers of a multi-layered printed circuit board(Lawson

and Erjavec 2017). The holes were copperplated by electrolysis, and the factors and their levels are shown in Table 1.

This was an existing plating process that was experiencing a problem. The copperplate thickness was not uniform, since it was thicker at top and bottom of the drilled holes than in the middle. Engineers were hoping to increase the uniformity of the copperplate thickness by experimentation. The response was the average of the natural log of the variance of the thickness of the copperplate measured at three places: top, middle and bottom.

Five of the factors had quantitative levels, and the low and high were chosen to span the range of what was normally used in practice. The last two factors shown in Table 1 had discrete levels. The high levels for these factors represented the normal, and the low represented experimental conditions.

Table 1: Factors and Levels for Copper Plating Experiment

Factor	Levels	
	-	+
A.Copper concentration	16 gm/L	19 gm/L
B.Chloride concentration	65 PPM	85 PPM
C.Acidity (H+) concentration	198 gm/L	225 gm/L
D.Temperature	72	78
E.Total current	180 amp-hrs	192.5 amp-hrs
F.Position in cell	right	left
G.Surface condition	smooth	slightly rough

In the following code the 12-run Plackett-Burman design was created using the `PBDes()` function in the R package `daewr`. The default for the `randomize=FALSE` option was used, and it produced the design in standard order, although the actual experiments were run in random order to prevent biases from unknown factors. The third line gives the response (natural log variance) from the experiments in standard order.

```
library(daewr)
R> Design<-PBDes(nruns=12,nfactors=7)
R> lnVar<-c(-2.18,-2.24,-3.08,-3.63,-3.72,
            -2.04,-3.66,-2.55,-2.54,-2.85,-1.94,-3.21)
```

In the code below, the `HierAFS()` function in the R package `daewr` package was used to fit a model using the forward stepwise regression that enforces model hierarchy. The arguments `m=0` and `c=7` in the call statement indicate the number of three-level and two-level factors respectively. The last argument `step=2` indicates the number of forward steps to perform. This number is usually determined by the number of steps just before the increase in  $R^2$  becomes negligible or just before the last term to enter the model is not significant.

```
R> HierAFS(lnVar,Design,m=0,c=7,step=2)
      formula      R2
1      y~C+F+C:F 0.615
2 y~F+G+F:G+C+C:F 0.776
```

In the output below the code, it can be seen that the the first variable to enter the equation was the C:F (Chloride concentration  $\times$  Position in cell) interaction. The `HierAFS()` function also included the main effect C (Chloride concentration) and main effect F (Position in cell) to insure model hierarchy. In the second step the F:G (Position in cell  $\times$  Surface condition) interaction entered the model along with the main effect for G to insure model hierarchy.

To see the details of the model fits at each step, the functions `ihstep()` and `fhstep()` can be used. These subroutine functions are called by `HierAFS()` when it is used. In the code below the `ihstep()` function is used to fit the initial model. The arguments `m=0` and `c=7` have the same meaning as they had in the `HierAFS()` function call. This function returns a character vector of the factor names entered at this step and prints the `lm()` summary to the console. It can be seen that the variable entered C:F is significant at the ( $p=0.0106$ ) level. In this example, the variable `trm` captures the variables entered in the first step.

```
R> trm<-ihstep(lnVar,Design,m=0,c=7)

Call:
lm(formula = y ~ (.), data = d1)

Residuals:
    Min       1Q   Median       3Q      Max
-0.70667 -0.19333 -0.08167  0.23500  0.77333

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -2.8033    0.1368 -20.499 3.36e-08 ***
C              0.0750    0.1368   0.548  0.5984
F              0.1683    0.1368   1.231  0.2533
C.F           -0.4533    0.1368  -3.315  0.0106 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4737 on 8 degrees of freedom
Multiple R-squared:  0.6155,    Adjusted R-squared:  0.4713
F-statistic: 4.268 on 3 and 8 DF,  p-value: 0.04473
```

In the code below, the `fhstep()` is used to make the second step. The arguments `m=0` and `c=7` have the same meaning as they had in the `HierAFS()` function call, and the last argument `trm` is the vector of factors that entered the model at the previous step. The resulting output shows that the next term to enter the model was the F:G interaction and its parent main effect G. However, F:G was not significant. Therefore, only one step of the forward stepwise regression enforcing model hierarchy is required for this problem.

```
R> trm<-fhstep(lnVar,Design,m=0,c=7,trm)

Call:
lm(formula = y ~ (.), data = d2)

Residuals:
    Min       1Q   Median       3Q      Max
-0.50333 -0.24500 -0.04792  0.14937  0.54250

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.803333    0.120408 -23.282 4.12e-07 ***
F             0.168333    0.120408   1.398  0.21160
G             0.100625    0.127712   0.788  0.46074
F.G          0.245625    0.127712   1.923  0.10280
C            -0.006875    0.127712  -0.054  0.95882
C.F         -0.486875    0.127712  -3.812  0.00884 **
---

```

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

```
Residual standard error: 0.4171 on 6 degrees of freedom  
Multiple R-squared:  0.7764,    Adjusted R-squared:  0.5901  
F-statistic: 4.167 on 5 and 6 DF,  p-value: 0.05574
```

If more than two steps are required for obtaining the final model, the function `fhstep()` can be called repeatedly to see the details of the model fit at each of the steps.

The final model for this problem is:

$$\ln(s^2) = -2.8033 + 0.0750C + 0.1683F - 0.4533(C \times F), \quad (1)$$

where  $C$ ,  $F$  are the coded factor levels shown in Table 1,  $C \times F$  is the product or interaction of  $C$  and  $F$ , and  $\ln(s^2)$  is the predicted natural log variance. To minimize the natural log of the variance and make the copperplate thickness more uniform from top to bottom, the hole should be placed in the right side of the electrolysis cell (i.e., factor  $F = -$ ) and Acid concentration should be 198gm/L (i.e., factor  $C = -$ ) as shown in the interaction plot in Figure 1.

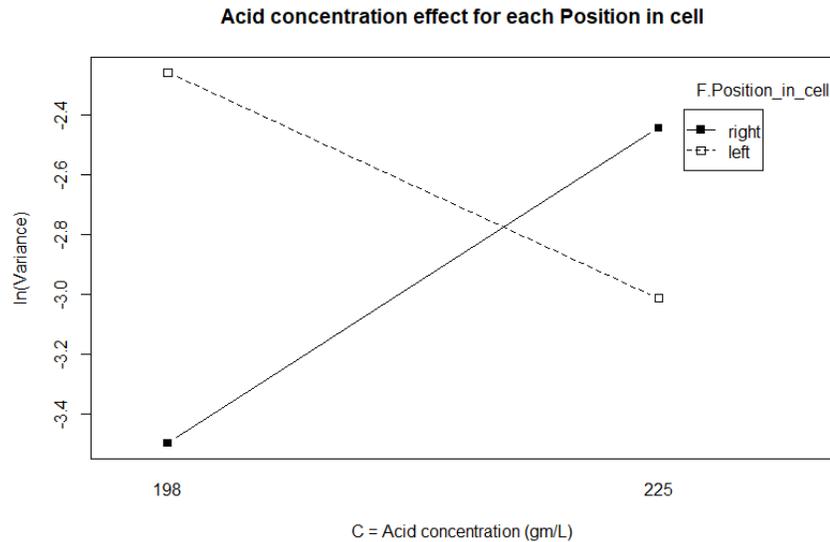


Figure 1: Interaction Plot.

The code to create the interaction plot in Figure 1 is shown below.

```
R> C.Acid_concentration<-211.5+(27/2)*Design$C  
R> F.Position_in_cell<-rep(" ",12)  
R> for (i in 1:12){  
R>   if (Design$F[i]==-1){F.Position_in_cell[i]="right"}  
R>   if (Design$F[i]==1){F.Position_in_cell[i]="left"}  
R> }  
R> C.Feedback_spring<-as.factor(C.Feedback_spring)  
R> hdes<-data.frame(C.Acid_concentration,F.Position_in_cell,lnVar)  
R> with(hdes, (interaction.plot(C.Acid_concentration, F.Position_in_cell,  
+ lnVar, type = "b",pch = c(0,15), leg.bty = "o",
```

```
+ main = "Acid concentration effect for each Position in cell",
+ xlab = "C = Acid concentration (gm/L)",
+ ylab = "ln(Variance)")
```

The analysis of the data from this experiment shown in Lawson and Erjavec(2017) was made by analysis of the main effects only. Factor  $E$  (Total current) was marginally significant, and the 9th column in the full Plackett-Burman design matrix (that included 11 columns) was significant. However, no main effect was assigned to the 9th column, and each column in the full design matrix was shown to be partially confounded with all 21 of the possible two-factor interactions. Therefore, it was concluded that it would be difficult to interpret the results based on this analysis alone. Follow-up experiments would be required to predict what conditions would result in the maximum uniformity of the copperplate thickness.

On the other hand, the analysis based on the forward stepwise regression enforcing model hierarchy has a simple interpretation and the condition predicted to maximize the uniformity was found. The three experiments conducted at the predicted optimal conditions (i.e., low level of factor  $C$  and high level of factor  $F$ ) included two of the three lowest log variance results found. Additional confirmatory experiments could next be run to further test the predicted optimum.

### Example 2:

Montgomery(2005) presented an example of a 16-run  $2_{IV}^{6-2}$  fractional factorial experiment for studying factors that affected the thickness of a photoresist layer. In this example main effects A, B, C, and E along with the confounded string of two-factor interactions AB=CE.were found to be significant. It would have taken follow-up experiments to determine which interaction was causing the significance of the confounded string of interactions and identify the optimal factor settings as was done in the last example. Instead, Johnson and Jones(2010) considered the same situation using an Alternative Screening Design. They simulated data for the photoresist thickness where the generated data had CE as the active two-factor interaction. In the code below, the `AltScreen()` function in the `daewr` package was used to generate the same six-factor Alternative screening design used in Johnson and Jones' article, and the thickness data was the simulated data. The `HierAFS()` function identified the correct model for the simulated data in three steps, and the variable that entered in the fourth step, B:E, was not significant (p-value=0.11327). Had the original experiment been conducted with this design, no follow-up experiments would have been required.

```
R> library(daewr)
R> Design<-AltScreen(nfac=6,randomize=FALSE)
R> Thickness<-c(4494,4592,4357,4489,4513,4483,4288,4448,4691,4671,4219,4271,
+ 4530,4632,4337,4391)
R> HierAFS(Thickness,Design,m=0,c=6,step=3)
      formula      R2
1      y~A 0.660
2      y~B+A 0.791
3 y~C+E+C:E+B+A 0.953
```

The details of the fit at the third step is shown following the code below. This code shows the second call of `fhstep()`. All terms in the model were significant at this step.

```
R> trm<-fhstep(Thickness,Design,m=0,c=6,trm)

Call:
lm(formula = y ~ (.), data = d2\)
Residuals:
      Min       1Q   Median       3Q      Max
-56.625 -19.625   2.625  23.125  53.875
```

```

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) 4462.875      9.494 470.073 < 2e-16 ***
C            -34.250      9.494  -3.608 0.00479 **
E             21.500      9.494   2.265 0.04700 *
C.E          54.750     13.427   4.078 0.00222 **
B           -77.750     11.628  -6.687 5.45e-05 ***
A            85.500     11.628   7.353 2.44e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

Residual standard error: 37.98 on 10 degrees of freedom
Multiple R-squared:  0.9533,    Adjusted R-squared:  0.93
F-statistic: 40.85 on 5 and 10 DF,  p-value: 2.447e-06

```

### Example 3:

A 17-run Definitive Screening Design was used in studying the effect of eight factors on the properties of  $\text{TiO}_2$  as a support material for metal and metal oxide catalysts in oxidative synthesis and pollution control reactions Olsen et. al.(2016). The eight factors were: B (Speed of  $\text{H}_2\text{O}$  addition), C (Amount of  $\text{H}_2\text{O}$ ), E (Drying Time), F (Drying Temp.), G (Calcination Ramp Rate), H (Calcination Temp.), I (Calcination Time), and J (Mole % Al). The process outputs or properties of the  $\text{TiO}_2$  of interest were the Crystallite diameter, Surface Area, Pore volume, and Pore diameter. The purpose of the experiments was to identify combinations of process input factors that would be appropriate for various catalysts applications.

The following code illustrates calling the `DefScreen()` function to create a Definitive Screening design with 8 three-level factors in standard order. The vector of data `pd` is the pore diameter data from Table 5 in Olsen et. al.(2016). The default column names in the design created by the `DefScreen()` were renamed in the fourth line of code to match the factor names in the article. The `FitDefSc()` function was called in the next line of code to fit a model to the data, and the output is shown below the code.

The `FitDefSc()` attempts to find an appropriate model by taking into account the special structure of the Definitive Screening Design. That special structure is that the linear main effects are completely unconfounded with quadratic and two-factor interaction terms, while quadratic and two-factor interaction terms are partially confounded. Therefore, the `FitDefSc()` function fits a model to a Definitive Screening Design by first restricting main effects to the smallest main effects and those significant at at least the .20 level in a main effects model. Next forward stepwise selection is used to enter two-factor interactions and quadratic effects involving the main effects in the model. Usually this procedure results in a reasonable model, but sometimes it includes insignificant terms in the model, and additional steps can be taken to eliminate them.

```

library(daewr)
Design<-DefScreen(m=8,c=0)
pd<-c(5.4,4.35,12.91,3.79,4.15,14.05,11.40,4.29,3.56,11.4,
      10.09,5.9,9.54,4.53,3.919,8.1,5.35)
colnames(Design)<-c('B','C','E','F','G','H','J','K')
FitDefSc(pd,Design,alpha=.10)
Call:
lm(formula = y ~ (.), data = ndesign)

Residuals:
Min 1Q Median 3Q Max
-0.8140 -0.4303 0.1504 0.2182 0.8289

Coefficients:

```

```

      Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.6710    0.4749   11.940 6.57e-06 ***
B            0.7664    0.2066    3.710 0.00756 **
G            0.7356    0.2066    3.561 0.00921 **
B:G         -0.5987    0.2268   -2.640 0.03344 *
C            0.4442    0.2066    2.150 0.06858 .
G:C         -0.9565    0.2438   -3.923 0.00573 **
I(B^2)      1.8801    0.5299    3.548 0.00937 **
E           -0.8686    0.2066   -4.205 0.00401 **
H            3.1579    0.2066   15.287 1.23e-06 ***
K           -0.5743    0.2066   -2.780 0.02730 *
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7729 on 7 degrees of freedom
Multiple R-squared: 0.9797, Adjusted R-squared: 0.9537
F-statistic: 37.62 on 9 and 7 DF, p-value: 4.168e-05

```

The model shown above contains a term C that is not significant at the  $\alpha = 0.05$  level, therefore a backwards elimination was performed using the `ols_step_backward_p()` function in the R package `olsrr`. When this was done, all terms were eliminated except H and the quadratic term for B as shown in the section of output below the code.

```

B<-Design$B
C<-Design$C
E<-Design$E
G<-Design$G
H<-Design$H
K<-Design$K
BG<-B*G
B2<-B*B
CG<-C*G
desmod<-data.frame(B,C,E,G,H,K,BG,B2,CG)
library(olsrr)
modr<-lm(pd~.,data=desmod)
back<-ols_step_backward_p(modr,prem=.05,progress=T)

```

Parameter Estimates

---

model	Beta	Std. Error	Std. Beta	t	Sig	lower	upper
(Intercept)	5.033	1.085		4.640	0.000	2.707	7.360
H	3.158	0.502	0.822	6.288	0.000	2.081	4.235
B2	2.654	1.195	0.290	2.220	0.043	0.090	5.218

---

Since this model only has two terms, a comparison was made by using the `HierAFS()` function to fit a model, as shown in the code below. It can be seen from the output of the first step that the linear main effect for factor H entered the model first. In the second step, the quadratic term for factor B entered the model and the linear effect of B was also included to preserve model hierarchy. A third step (not shown) entered the main effect E, but it was not significant. The final model found after two steps was fit by the `lm()` function and is shown below the code. Therefore after two steps model found is the nearly same as the model found after performing backwards elimination on the model found by the `FitDefSc()` function.

```

> HierAFS(pd,Design,m=8,c=0,step=2)
      formula      R2
1      y~H 0.676
2 y~I(B^2)+B+H 0.800

moddf<-lm(pd~B+I(B^2)+H,data=Design)
summary(moddf)
Call:
lm(formula = pd ~ B + I(B^2) + H, data = Design)

Residuals:
      Min       1Q   Median       3Q      Max
-2.7913 -1.0214 -0.2121  0.5265  2.8306

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   5.0333      1.0279   4.897 0.000292 ***
B              0.7664      0.4758   1.611 0.131286
I(B^2)        2.6545      1.1327   2.343 0.035662 *
H              3.1579      0.4758   6.636 1.62e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.78 on 13 degrees of freedom
Multiple R-squared:  0.8004,    Adjusted R-squared:  0.7543
F-statistic: 17.38 on 3 and 13 DF,  p-value: 7.809e-05

```

In order to identify the process inputs that would produce different pore diameters, the contour plots like the ones shown in Figure 2 were used. The plot on the left is the two term model found using the `FitDefSc()` and backwards elimination, and the plot on the right was the three term model found by the `HierAFS()` function. Since the linear coefficient on B(Speed of H<sub>2</sub>O addition) is less than one, the two plots are very similar.

The code below was used to create the contour plots shown in Figure 2.

```

par (mfrow=c(1,2), cex.sub=.8)
##### contour plot of two term equation
Speed_H2O_Addition<-seq(from=1.0,to=3.0,by=(2)/(24))
Calc_Temp<-seq(from=400,to=700,by=(300/24))
B<-(Speed_H2O_Addition-2)
H<-(Calc_Temp-550)/150
Pore_Diameter<-matrix(rep(0,625),nrow=25)
for (i in 1:25) {
  for (j in 1:25) {
    Pore_Diameter[i,j]<-5.033+3.158*H[j]+2.654*B[i]*B[i]
  }
}
contour(Speed_H2O_Addition,Calc_Temp,Pore_Diameter,
  xlab="B=Speed of H2O Addition (1=Slow, 2=Med., 3=Fast)",
  ylab="H=Calcination Temp. Deg. C", main="Pore Diameter (nm) Two term model")

# contour plot of 3 term equation
Speed_H2O_Addition<-seq(from=1.0,to=3.0,by=(2)/(24))

```

```

Calc_Temp<-seq(from=400,to=700,by=(300/24))
B<-(Speed_H2O_Addition-2)
H<-(Calc_Temp-550)/150
Pore_Diameter<-matrix(rep(0,625),nrow=25)
for (i in 1:25) {
  for (j in 1:25) {
    Pore_Diameter[i,j]<-5.033+3.158*H[j]+0.7664*B[i]+2.654*B[i]*B[i]
  }
}
contour(Speed_H2O_Addition,Calc_Temp,Pore_Diameter,
  xlab="B=Speed of H2O Addition (1=Slow, 2=Med., 3=Fast)",
  ylab="H=Calcination Temp. Deg. C", main="Pore Diameter (nm) Three term model" )
par (mfrow=c(1,1))

```

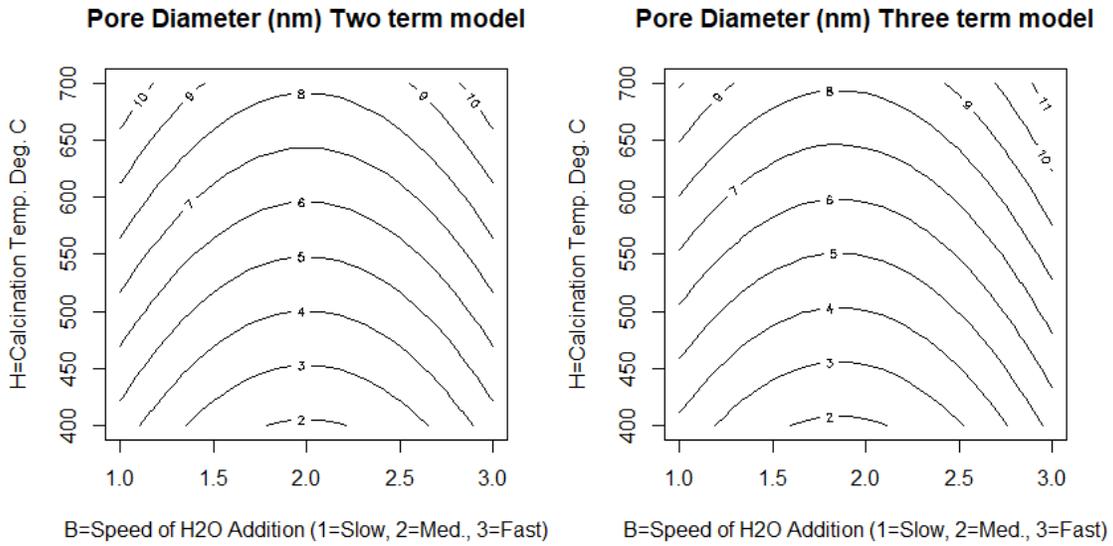


Figure 2: Contour plot of TiO<sub>2</sub> Pore Diameter.

## Summary and discussion

The general quadratic model for  $k$  three-level quantitative factors is:

$$y = \beta_0 + \sum_{i=1}^k \beta_i x_i + \sum_{i=1}^k \beta_{ii} x_i^2 + \sum_{i < j}^k \sum_{i < j}^k \beta_{ij} x_i x_j + \epsilon \quad (2)$$

This model is a good approximation to any unknown functional relationship between the response  $y$  and the independent variables  $x_i$  for  $i = 1$  to  $k$  since it is in the form of a second-order Taylor series approximation. When the independent variables  $x_i$  only have two levels, or a linear approximation is sufficient, this model simplifies to:

$$y = \beta_0 + \sum_{i=1}^k \beta_i x_i + \sum_{i < j}^k \sum_{i < j}^k \beta_{ij} x_i x_j + \epsilon \quad (3)$$

If there are only a few independent variables, data to fit model (2) or (3) can usually be obtained in one set of experiments. However, if there are several independent variables the number of experiments required, in one design, to estimate all the coefficients in either model (2) or model (3) could become excessive. In that situation, a sequential approach is usually recommended as illustrated in the right column of Figure 3.

	<b>Few independent variables</b>	<b>Many independent variables</b>
	One-shot Experiment	Sequential Set of Experiments
Model (3)	Full factorial or Resolution V Fractional Factorial	Resolution III or IV Fractional Factorial ↓ Follow-up Experiments to obtain unconfounded estimates of significant main effects and two factor interactions
Model (2)	Response Surface Design	↓ Addition of centerpoints and axial points on quantitative variables to obtain estimates of quadratic effects

Figure 3: One-shot versus Sequential Experimentation.

When there are only two-level factors, the Plackett-Burman designs, Alternative Screening designs, or Model-Robust designs offer an alternative to obtaining enough data for fitting a model with the important subset of terms in model (3) requiring less experimentation than the sequential approach. When there are three-level quantitative factors or a mixture of two-level and three-level factors, the Definitive Screening designs or column-augmented Definitive Screening designs also offer a one-shot alternative to obtaining the data to fit model (2) or a mixture of model (2) and (3) with less experimentation than required by the sequential experimentation approach.

There is a difference in the approach to model fitting when there are few independent variables than when there are many independent variables. With few independent variables, the designs listed in column 1 of Figure 3, provide sufficient data for estimating the coefficients in model (2) or (3), and these models can be fit by regression analysis. The fitted model can then be refined by dropping insignificant terms. On the other hand, when there are many independent variables, the one-shot alternatives to sequential experimentation do not provide enough data to fit model (2) or (3) with all independent variables. The assumption is that not all the variables will be important and a subset selection procedure will be used to obtain and refine the model.

Subset selection procedures like forward stepwise regression, full stepwise regression, or all possible subset regression (up to a given size) are different ways of obtaining a subset model. However, for finding a subset of model (2) or (3) in a subspace of the independent variables, the forward selection procedure that enforces model hierarchy seems to be particularly valuable. This selection procedure automatically results in one model unlike the all possible subsets regression that may result in several near equivalent models for explaining the data. Secondly, unlike the simple forward or full stepwise regression, it ensures the model hierarchy normally found in the real world. When a Definitive Screening design is used with all quantitative three-level factors, the `FitDefSc()` function can sometimes find an appropriate model containing a subset of the terms in model (2) in one step.

A Google Scholar search reveals several recent articles that illustrate the use of Plackett-Burman designs (Plackett and Burman 1946) and Definitive Screening designs (Jones and Nachtsheim 2011) in applications. The articles discussing Definitive Screening designs illustrate obtaining a subset of the full quadratic model (2) for fitting the data. However, most all applications of Plackett-Burman designs illustrate the analysis by main effects only. The Model Robust designs (Li and Nachtsheim 2000), and Alternative Screening designs in 16 runs (Jones and Montgomery 2010) appear to be underutilized. When there are only two-level factors, and a forward stepwise regression enforcing model hierarchy is employed, the latter two designs are very useful for screening main effects and two-factor interactions.

As illustrated in this vignette, the R package `daewr` has functions for obtaining Definitive Screening designs, Model Robust designs, Alternative Screening designs, and Plackett-Burman designs. It also contains functions for model fitting by forward selection enforcing model hierarchy or using the special structure of a Definitive Screening design. The data used in the examples in this article were taken from other published sources, and these examples illustrate economy of the designs available in `daewr` and the efficacy of the model fitting functions. It is hoped that the functions available in `daewr` will make these designs and method of analysis more accessible to practitioners in various fields of application.

## Acknowledgments

The function `step.forward()` was originally written by Gerhard, Krennrich, email="krennri@uni-heidelberg.de" to call `ihstep()` and `fhstep()` repeatedly. `step.forward()` was modified and renamed `HierAFS()` (so as not to override other functions with the same name in other packages) and then included in the `daewr` package.

## References

- Groemping, U. (2020) "FrF2:: Fractional Factorial Designs with 2-Level Factors", R package version 1.55, <https://CRAN.R-project.org/package=FrF2>.
- Hamada, M. and Wu, C. F. J. (1992) "Analysis of Experiments with Complex Aliasing", *JQT*, 24, pp 130-137.
- Johnson, M. E. and Jones, B. (2010) "Classical Design Structure of Orthogonal Designs with Six to Eight Factors and Sixteen Runs.", *Quality and Reliability Engineering International* 27, pp. 61–70
- Jones, B. and Montgomery, D. C. (2010) "Alternatives to Resolution IV Screening Designs in 16 Runs", *Int. J. of Experimental Design and Process Optimisation*,1,pp 285-295.
- Jones, B. and Nachtsheim (2011) "A Class of Three-level Designs for Definitive Screening in the Presence of Second-order Effects", *JQT*, 43, pp 1-15.
- Jones, B. and Nachtsheim (2013) "Definitive Screening Designs with added two-level categorical factors", *JQT*, 45, pp 121-129.
- Lawson, J. (2015) *Design and Analysis of Experiments with R*, CRC Press, Boca Raton.
- Lawson, J. (2020) "`daewr`: Design and Analysis of Experiments with R", R package version 1.2-5, <http://www.r-qualitytools.org>.
- Lawson, J. and Erjavec, J. (2017) *Basic Experimental Strategies and Data Analysis for Science and Engineering*, CRC Press, Boca Raton
- Li, X., Sudarsanam, N. and Frey, D.(2006) "Regularities in Data from factorial Experiments", *Complexity*, 11, pp. 32-45.
- Li, W. and Nachtsheim, C. (2000) "Model Robust Factorial Designs", *Technometrics*, 42, pp 345-352.
- Lin, D.K.J. (1999) "Spotlight Interaction Effects in Main Effect Plans: A Supersaturated Design Approach", *Quality Engineering*, 11, pp. 133-139.
- Lin, D.K.J. and Draper, N.R. (1992) "Projection Properties of Plackett-Burman Designs", *Technometrics*, 34, pp 423-428.
- Montgomery, D. C.(2005) *Design and Analysis of Experiments* 6th ed., John Wiley & Sons, Hoboken, N.J.
- Olsen, R., Lawson, J., Rhobok, N. and Woodfield, B.(2016) "Practical comparison of traditional and definitive screening designs in chemical process development", *Int. J. of Experimental Design and Process Optimisation*, 5, pp. 1-22.
- Plackett, R. L. and Burman, J. P.(1946) "The Design of Optimum Multifactor Experiments", *Biometrika*, 33, pp. 305-325.

Roth, T. (2016) “`qualityTools`:: Statistics in Quality Science.”, R package version 2.1-1, <https://CRAN.R-project.org/package=qualityTools>.

Wang, J.C. and Wu, C.F.J.(1995) “A hidden projection property of Plackett-Burman and related designs”, *Statistica Sinica*, 5, 235-250.