Experimental Design in Marketing

Accelerate Learning with Agile, Concurrent, Multifactor Experimentation

Robert A Agnew, PhD

Analytics Consultant

July 2023

DESIGN

Design Elements

- Factors - offer attributes that can be varied across prospects in an experiment

- Levels - mutually exclusive discrete variations within factors

- Test Cell - specific combination of factors and levels; prospects are randomized into test cells

- Experimental Design - estimable combination of test cells, meaning that expected response can be estimated for all factor-level combinations; test cells are run concurrently to maximize efficiency and learning

- Response - quantitative experimental outcome, either continuous like sales or binary like conversion

- We will demonstrate these elements for alternative credit card balance transfer offers with the following factors and discrete, bucketed levels within those factors; other elements like card composition are assumed constant; go-to APR is assumed to be risk-score driven and is not part of the experimental design but it would be potentially available for analysis
    - Intro or teaser rate - 0%, 0.9%, 1.9%
    - Intro or teaser duration - 12m, 15m, 18m
    - Balance transfer fee - 2%, 3%

- We will generate estimable designs using the optFederov function in R package AlgDesign; same can be done with SAS public macro %MktEx with no need for a QC or JMP license. Python does not appear to have a comparable general design package.

- In addition, we will analyze experimental results from simulated conversions, again in R but the same thing can be done in SAS using standard statistical procedures. R-code is highlighted in yellow. Key results are highlighted in green.

- Our demonstration problem does not fit within the two-level domain for which Classical 2^k and Plackett-Burman designs are available from various sources, including FrF2 and pb functions in R package FrF2. Informative Plackett-Burman use case is available in Bell, Ledolter & Swersy, Experimental design on the front lines of marketing: Testing new ideas to increase direct mail sales, *International Journal of Research in Marketing* 23 (2006) 309-319.

Full-Factorial Design Grid

```
> CELLS <- expand.grid(RATE=c("0.0%","0.9%","1.9%"),DURATION=c("12m","15m","18m"),FEE=c("2%","3%"))
> CELLS
```

	RATE	DURATION	FEE
1	0.0%	12m	2%
2	0.9%	12m	2%
3	1.9%	12m	2%
4	0.0%	15m	2%
5	0.9%	15m	2%
6	1.9%	15m	2%
7	0.0%	18m	2%
8	0.9%	18m	2%
9	1.9%	18m	2%
10	0.0%	12m	3%
11	0.9%	12m	3%
12	1.9%	12m	3%
13	0.0%	15m	3%
14	0.9%	15m	3%
15	1.9%	15m	3%
16	0.0%	18m	3%
17	0.9%	18m	3%
18	1.9%	18m	3%

```
> write.csv(CELLS,"c:/DOE/DESIGN/FACTORIAL",row.names=FALSE) # Save full-factorial grid
```

This grid displays all 18 potential experimental cells. A full-factorial experiment may be feasible but we want to consider partial-factorial designs with fewer cells. These designs will assume away some or all interactions but they may be more efficient in terms of sample size.

Six-Cell Partial Factorial Design

```
> library(AlgDesign)
> set.seed(2) # Set random number seed
> DESIGN <- optFederov(~.,data=read.csv("c:/DOE/DESIGN/FACTORIAL"),nTrials=6)
> DESIGN$design
```

	RATE	DURATION	FEE
3	1.9%	12m	2%
4	0.0%	15m	2%
8	0.9%	18m	2%
10	0.0%	12m	3%
14	0.9%	15m	3%
18	1.9%	18m	3%

```
> write.csv(DESIGN$design,"c:/DOE/DESIGN/DESIGN6_1",row.names=FALSE) # Save first 6-cell design
```

Six is the minimum number of cells for a feasible design. Formula is: 1 + (Number of Rate Levels - 1) + (Number of Duration Levels - 1) + (Number of Fee Levels - 1). If you put nTrials = 5, there will be an error message. This experimental design will enable estimation of main effects, no interactions. However, a regression model based on these main effects will predict outcomes for all 18 factorial cells.

Alternative Six-Cell Partial Factorial Design

```
> library(AlgDesign)
> set.seed(3) # Set random number seed
> DESIGN <- optFederov(~.,data=read.csv("c:/DOE/DESIGN/FACTORIAL"),nTrials=6)
> DESIGN$design
```

	RATE	DURATION	FEE
2	0.9%	12m	2%
6	1.9%	15m	2%
7	0.0%	18m	2%
10	0.0%	12m	3%
14	0.9%	15m	3%
18	1.9%	18m	3%

```
> write.csv(DESIGN$design,"c:/DOE/DESIGN/DESIGN6_2",row.names=FALSE) # Save second 6-cell design
```

An alternative six-cell design is generated by changing the random number seed, although some seeds will simply repeat designs. Marketing can select from alternative feasible designs. Not all combinations of six cells represent feasible designs of course, and this design generator is not infallible. If you set the random number seed at 1, it fails. Before implementing any design in an expensive experiment, it makes sense to run it against simulated data to make sure everything is working right. This will also help with sample size selection. We will illustrate later.

Nine-Cell Partial Factorial Design

```
> library(AlgDesign)
> set.seed(3) # Set random number seed
> DESIGN <- optFederov(~.,data=read.csv("c:/DOE/DESIGN/FACTORIAL"),nTrials=9)
> DESIGN$design
```

	RATE	DURATION	FEE
2	0.9%	12m	2%
3	1.9%	12m	2%
4	0.0%	15m	2%
6	1.9%	15m	2%
7	0.0%	18m	2%
8	0.9%	18m	2%
10	0.0%	12m	3%
14	0.9%	15m	3%
18	1.9%	18m	3%

```
> write.csv(DESIGN$design,"c:/DOE/DESIGN/DESIGN9",row.names=FALSE) # Save 9-cell design
```

This design enables more refined estimation of main effects plus a few interactions as we will illustrate later. Once again, the random number seed can be used to generate alternative designs. This nine-cell design is not automatically better than a six-cell design, however. An experiment will normally be planned within a stipulated overall sample size. Hence, more cells may mean fewer observations per cell.

SIMULATION

Monte Carlo Simulation of Three Experimental Designs

- **Judgmental response probabilities specified for 18 factorial cells**

- **Simulated binary response outcomes for three 90K designs**
  - **6 cells, 15K observations each**
  - **9 cells, 10K observations each**
  - **18 cells, 5K observations each**

- **Simulated experimental datasets allow us to demonstrate regression analysis and to compare estimation results across designs**

Specified Response Probabilities for all Combinations

	RATE	DURATION	FEE	PROB
1	0.0%	12m	2%	0.007
2	0.9%	12m	2%	0.005
3	1.9%	12m	2%	0.003
4	0.0%	15m	2%	0.009
5	0.9%	15m	2%	0.006
6	1.9%	15m	2%	0.004
7	0.0%	18m	2%	0.012
8	0.9%	18m	2%	0.008
9	1.9%	18m	2%	0.005
10	0.0%	12m	3%	0.005
11	0.9%	12m	3%	0.002
12	1.9%	12m	3%	0.001
13	0.0%	15m	3%	0.006
14	0.9%	15m	3%	0.003
15	1.9%	15m	3%	0.002
16	0.0%	18m	3%	0.010
17	0.9%	18m	3%	0.006
18	1.9%	18m	3%	0.004

Judgmentally specified response probabilities for all eighteen combinations. We will estimate these probabilities in three simulated experiments.

Specified Response Probabilities for Six-Cell Design

```
> prob6 <- prob[c(3,4,8,10,14,18)]
> x <- read.csv(file="c:/DOE/DESIGN/DESIGN6_1",header=TRUE)
> attach(x,warn.conflicts=FALSE)
> y <- data.frame(cbind(x,prob6))
> names(y) <- c("RATE","DURATION","FEE","PROB")
> y
```

	RATE	DURATION	FEE	PROB
1	1.9%	12m	2%	0.003
2	0.0%	15m	2%	0.009
3	0.9%	18m	2%	0.008
4	0.0%	12m	3%	0.005
5	0.9%	15m	3%	0.003
6	1.9%	18m	3%	0.004

```
> write.csv(y,file="c:/DOE/DESIGN/SIM_DESIGN6",row.names=FALSE) # Save six-cell grid with appended
response probabilities
> detach(x)
```

Specified Response Probabilities for Nine-Cell Design

```
> prob9 <- prob[c(2,3,4,6,7,8,10,14,18)]
> x <- read.csv(file="c:/DOE/DESIGN/DESIGN9",header=TRUE)
> attach(x,warn.conflicts=FALSE)
> y <- data.frame(cbind(x,prob9))
> names(y) <- c("RATE","DURATION","FEE","PROB")
> y
```

	RATE	DURATION	FEE	PROB
1	0.9%	12m	2%	0.005
2	1.9%	12m	2%	0.003
3	0.0%	15m	2%	0.009
4	1.9%	15m	2%	0.004
5	0.0%	18m	2%	0.012
6	0.9%	18m	2%	0.008
7	0.0%	12m	3%	0.005
8	0.9%	15m	3%	0.003
9	1.9%	18m	3%	0.004

```
> write.csv(y,file="c:/DOE/DESIGN/SIM_DESIGN9",row.names=FALSE) # Save nine-cell grid with appended response probabilities
> detach(x)
```

Scripts to Generate Simulated Binary Response Data

```
> # Generate simulated binary response data for 6-cell design, 15,000 observations per cell
> x <- read.csv(file="c:/DOE/DESIGN/SIM_DESIGN6",header=TRUE)
> attach(x,warn.conflicts=FALSE)
> set.seed(2)
> dat <- NULL
> for (i in 1:6){
+ y <- data.frame(cbind(data.frame(rep(RATE[i],15000),rep(DURATION[i],15000),rep(FEE[i],15000)),rbinom(15000,1,PROB[i])))
+ names(y) <- c("RATE","DURATION","FEE","RESPONSE")
+ dat <- rbind(dat,y)}
> detach(x)
> write.csv(dat,file="c:/DOE/DESIGN/SIM_DATA6",row.names=FALSE)
>
> # Generate simulated binary response data for 9-cell design, 10,000 observations per cell
> x <- read.csv(file="c:/DOE/DESIGN/SIM_DESIGN9",header=TRUE)
> attach(x,warn.conflicts=FALSE)
> set.seed(3)
> dat <- NULL
> for (i in 1:9){
+ y <- data.frame(cbind(data.frame(rep(RATE[i],10000),rep(DURATION[i],10000),rep(FEE[i],10000)),rbinom(10000,1,PROB[i])))
+ names(y) <- c("RATE","DURATION","FEE","RESPONSE")
+ dat <- rbind(dat,y)}
> detach(x)
> write.csv(dat,file="c:/DOE/DESIGN/SIM_DATA9",row.names=FALSE)
>
> # Generate simulated binary response data for full-factorial 18-cell design, 5,000 observations per cell
> x <- read.csv(file="c:/DOE/DESIGN/SIM_FACTORIAL",header=TRUE)
> attach(x,warn.conflicts=FALSE)
> set.seed(4)
> dat <- NULL
> for (i in 1:18){
+ y <- data.frame(cbind(data.frame(rep(RATE[i],5000),rep(DURATION[i],5000),rep(FEE[i],5000)),rbinom(5000,1,PROB[i])))
+ names(y) <- c("RATE","DURATION","FEE","RESPONSE")
+ dat <- rbind(dat,y)}
> detach(x)
> write.csv(dat,file="c:/DOE/DESIGN/SIM_DATA18",row.names=FALSE)
```

ESTIMATION

Six-Cell Linear Regression Model

```
> dat <- read.csv(file="c:/DOE/DESIGN/SIM_DATA6",header=TRUE) # Input simulation data for 6-cell case
> attach(dat,warn.conflicts=FALSE)
> mod <- lm(RESPONSE~RATE+DURATION+FEE,data=dat) # Linear main effects regression
> detach(dat)
> summary(mod)
```

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.0066667	0.0005999	11.113	< 2e-16
RATE 0.9%	-0.0016667	0.0006927	-2.406	0.0161
RATE 1.9%	-0.0032667	0.0006927	-4.716	2.41e-06
DURATION 15m	0.0016000	0.0006927	2.310	0.0209
DURATION 18m	0.0041333	0.0006927	5.967	2.43e-09
FEE 3%	-0.0030000	0.0004898	-6.124	9.13e-10

Residual standard error: 0.07348 on 89994 degrees of freedom
Multiple R-squared: 0.001022, Adjusted R-squared: 0.0009668
F-statistic: 18.42 on 5 and 89994 DF, p-value: < 2.2e-16

```
> # Precise p-value of model fit on experimental cells
> F <- as.vector(summary(mod)$fstatistic)
> pf(F[1],F[2],F[3],lower.tail=FALSE)
[1] 2.497426e-18
```

Main effects only, no interactions. RATE 0.0%, DURATION 12m, and FEE 2% embedded in intercept. All coefficients are statistically significant with correct signs.

Six-Cell Linear Regression Model Estimates

	RATE	DURATION	FEE	PROB	EST
1	0.0%	12m	2%	0.007	0.00667
2	0.9%	12m	2%	0.005	0.00500
3	1.9%	12m	2%	0.003	0.00340
4	0.0%	15m	2%	0.009	0.00827
5	0.9%	15m	2%	0.006	0.00660
6	1.9%	15m	2%	0.004	0.00500
7	0.0%	18m	2%	0.012	0.01080
8	0.9%	18m	2%	0.008	0.00913
9	1.9%	18m	2%	0.005	0.00753
10	0.0%	12m	3%	0.005	0.00367
11	0.9%	12m	3%	0.002	0.00200
12	1.9%	12m	3%	0.001	0.00040
13	0.0%	15m	3%	0.006	0.00527
14	0.9%	15m	3%	0.003	0.00360
15	1.9%	15m	3%	0.002	0.00200
16	0.0%	18m	3%	0.010	0.00780
17	0.9%	18m	3%	0.006	0.00613
18	1.9%	18m	3%	0.004	0.00453

- Extrapolation to all 18 factorial cells
- 6 experimental cells highlighted in blue
- Estimates are from regression formula via simple arithmetic on dummy variables, e.g., for Cell 17, EST = .0066667 - .0016667 + .0041333 - .003 = .0061333
- Error measures are intuitive rather than statistical

Six-Cell Logistic Regression Model

```
> dat <- read.csv(file="c:/DOE/DESIGN/SIM_DATA6",header=TRUE) # Input simulation data for 6-cell case
> attach(dat,warn.conflicts=FALSE)
> mod <- glm(RESPONSE~RATE+DURATION+FEE,family=binomial,data=dat) # Logistic main effects regression
> detach(dat)
> summary(mod)
```

Coefficients:

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-5.11636	0.12540	-40.801	< 2e-16
RATE 0.9%	-0.34755	0.13619	-2.552	0.010712
RATE 1.9%	-0.56422	0.14927	-3.780	0.000157
DURATION 15m	0.32914	0.13578	2.424	0.015347
DURATION 18m	0.77726	0.14534	5.348	8.90e-08
FEE 3%	-0.48844	0.09819	-4.974	6.55e-07

Main effects only, no interactions. RATE 0.0%, DURATION 12m, and FEE 2% embedded in intercept. All coefficients are statistically significant with correct signs. Regression formula predicts logits which convert to probabilities via the logistic function.

Six-Cell Logistic Regression Model Estimates

```
> prob18 <- read.csv(file="c:/DOE/DESIGN/SIM_FACTORIAL",header=TRUE) # Prob file for all 18 cells
> attach(prob18,warn.conflicts=FALSE)
> preds <- predict(mod,prob18[,-1],type="link") # Produces logit predictions
> pred18 <- cbind(prob18,mod$family$linkinv(preds)) # Converts to probability predictions
> detach(prob18)
> names(pred18) <- c("RATE","DURATION","FEE","PROB","EST")
> attach(pred18,warn.conflicts=FALSE)
> print(pred18,digits=3) # Print actual response probs vs estimates for all 18 cells
```

	RATE	DURATION	FEE	PROB	EST
1	0.0%	12m	2%	0.007	0.00596
2	0.9%	12m	2%	0.005	0.00422
3	1.9%	12m	2%	0.003	0.00340
4	0.0%	15m	2%	0.009	0.00827
5	0.9%	15m	2%	0.006	0.00585
6	1.9%	15m	2%	0.004	0.00472
7	0.0%	18m	2%	0.012	0.01288
8	0.9%	18m	2%	0.008	0.00913
9	1.9%	18m	2%	0.005	0.00737
10	0.0%	12m	3%	0.005	0.00367
11	0.9%	12m	3%	0.002	0.00259
12	1.9%	12m	3%	0.001	0.00209
13	0.0%	15m	3%	0.006	0.00509
14	0.9%	15m	3%	0.003	0.00360
15	1.9%	15m	3%	0.002	0.00290
16	0.0%	18m	3%	0.010	0.00794
17	0.9%	18m	3%	0.006	0.00562
18	1.9%	18m	3%	0.004	0.00453

```
>
> I <- c(3,4,8,10,14,18) # Indexes of experimental cells
> # Mean absolute errors for experimental cells, nonexperimental cells, and all cells
> c(mean(abs(PROB[I]-EST[I])),mean(abs(PROB[-I]-EST[-I])),mean(abs(PROB-EST)))
[1] 0.0007888889      0.0009882502      0.0009217964
> # Mean absolute percentage errors for experimental cells, nonexperimental cells, and all cells
> c(mean(abs(PROB[I]-EST[I])/PROB[I]),mean(abs(PROB[-I]-EST[-I])/PROB[-I]),mean(abs(PROB-EST)/PROB))
[1] 0.1594136      0.2759565      0.2371089
```

- Extrapolation to all 18 factorial cells
- 6 experimental cells highlighted in blue
- Estimates are from regression formula via logistic function
- Estimates for experimental cells are the same as linear regression model
- Logistic regression is much worse than linear regression in terms of extrapolation to nonexperimental cells - nonlinear link function evidently backfires. We focus on linear models.
- Error measures are intuitive rather than statistical

Nine-Cell Linear Regression Model

```
> dat <- read.csv(file="c:/DOE/DESIGN/SIM_DATA9",header=TRUE) # Input simulation data for 9-cell case
> attach(dat,warn.conflicts=FALSE)
> mod <- lm(RESPONSE~RATE+DURATION+FEE,data=dat) # Linear main effects regression
> detach(dat)
> summary(mod)
```

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.0085167	0.0005999	14.197	< 2e-16
RATE 0.9%	-0.0040333	0.0006266	-6.437	1.22e-10
RATE 1.9%	-0.0049667	0.0006266	-7.927	2.28e-15
DURATION 15m	0.0007333	0.0006266	1.170	0.242
DURATION 18m	0.0036667	0.0006266	5.852	4.88e-09
FEE 3%	-0.0031500	0.0005426	-5.805	6.46e-09

Residual standard error: 0.07674 on 89994 degrees of freedom
Multiple R-squared: 0.001587, Adjusted R-squared: 0.001531
F-statistic: 28.61 on 5 and 89994 DF, p-value: < 2.2e-16

```
> # Precise p-value of model fit on experimental cells
> F <- as.vector(summary(mod)$fstatistic)
> pf(F[1],F[2],F[3],lower.tail=FALSE)
[1] 4.266449e-29
```

Main effects only, no interactions. RATE 0.0%, DURATION 12m, and FEE 2% embedded in intercept. All coefficients except DURATION 15m are statistically significant with correct signs. Overall better fit on experimental cells than six-cell model.

Nine-Cell Linear Regression Model Estimates

```
> prob18 <- read.csv(file="c:/DOE/DESIGN/SIM_FACTORIAL",header=TRUE) # Prob file for all 18 cells
> attach(prob18,warn.conflicts=FALSE)
> pred18 <- cbind(prob18,predict(mod,prob18[,-1]))
> detach(prob18)
> names(pred18) <- c("RATE","DURATION","FEE","PROB","EST")
> attach(pred18,warn.conflicts=FALSE)
> print(pred18,digits=3) # Print actual response probs vs estimates for all 18 cells
```

	RATE	DURATION	FEE	PROB	EST
1	0.0%	12m	2%	0.007	0.00852
2	0.9%	12m	2%	0.005	0.00448
3	1.9%	12m	2%	0.003	0.00355
4	0.0%	15m	2%	0.009	0.00925
5	0.9%	15m	2%	0.006	0.00522
6	1.9%	15m	2%	0.004	0.00428
7	0.0%	18m	2%	0.012	0.01218
8	0.9%	18m	2%	0.008	0.00815
9	1.9%	18m	2%	0.005	0.00722
10	0.0%	12m	3%	0.005	0.00537
11	0.9%	12m	3%	0.002	0.00133
12	1.9%	12m	3%	0.001	0.00040
13	0.0%	15m	3%	0.006	0.00610
14	0.9%	15m	3%	0.003	0.00207
15	1.9%	15m	3%	0.002	0.00113
16	0.0%	18m	3%	0.010	0.00903
17	0.9%	18m	3%	0.006	0.00500
18	1.9%	18m	3%	0.004	0.00407

```
>
> I <- c(2,3,4,6,7,8,10,14,18) # Indexes of experimental cells
> # Mean absolute errors for experimental cells, nonexperimental cells, and all cells
> c(mean(abs(PROB[I]-EST[I])),mean(abs(PROB[-I]-EST[-I])),mean(abs(PROB-EST)))
[1] 0.0003666667    0.0009685185    0.0006675926
> # Mean absolute percentage errors for experimental cells, nonexperimental cells, and all cells
> c(mean(abs(PROB[I]-EST[I]))/PROB[I]),mean(abs(PROB[-I]-EST[-I]))/PROB[-I]),mean(abs(PROB-EST)/PROB))
[1] 0.09115741    0.27080247    0.18097994
```

- Extrapolation to all 18 factorial cells
- 9 experimental cells highlighted in blue
- Estimates from regression formula via simple arithmetic on dummy variables
- Error measures are intuitive rather than statistical
- Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) are lower on experimental cells than the 6-cell case. Both are higher for nonexperimental cells.
- Overall, MAE is lower for the 9-cell case and MAPE is higher for the 9-cell case.

Nine-Cell Linear Regression Model with Interactions

Coefficients: (5 not defined because of singularities)

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.0087500	0.0007674	11.402	< 2e-16
RATE 0.9%	-0.0045500	0.0010853	-4.192	2.76e-05
RATE 1.9%	-0.0046500	0.0007674	-6.059	1.37e-09
DURATION 15m	0.0006500	0.0010853	0.599	0.549
DURATION 18m	0.0035500	0.0007674	4.626	3.73e-06
FEE 3%	-0.0036500	0.0007674	-4.756	1.98e-06
RATE 0.9% : DURATION 15m	0.0012000	0.0018798	0.638	0.523
RATE 1.9% : DURATION 15m	-0.0009500	0.0013292	-0.715	0.475
RATE 0.9% : DURATION 18m	0.0003500	0.0013292	0.263	0.792
RATE 1.9% : DURATION 18m	NA	NA	NA	NA
RATE 0.9% : FEE 3%	NA	NA	NA	NA
RATE 1.9% : FEE 3%	NA	NA	NA	NA
DURATION 15m : FEE 3%	NA	NA	NA	NA
DURATION 18m : FEE 3%	NA	NA	NA	NA

Residual standard error: 0.07674 on 89991 degrees of freedom
Multiple R-squared: 0.001603, Adjusted R-squared: 0.001514
F-statistic: 18.06 on 8 and 89991 DF, p-value: < 2.2e-16

Main effects plus pairwise interactions. RATE 0.0%, DURATION 12m, and FEE 2% embedded in intercept. Three RATE : DURATION interaction terms are estimated but they are all insignificant. Overall fit is worse than main effects alone. We do not pursue this model further. In real data, of course, some interaction terms could turn out to be significant, but in our simulated data they are not. This approach provides a good way to assess the significance of interactions.

Eighteen-Cell Full-Factorial Linear Regression Model

```
> dat <- read.csv(file="c:/DOE/DESIGN/SIM_DATA18",header=TRUE) # Input simulation data for 18-cell full-factorial case
> attach(dat,warn.conflicts=FALSE)
> mod <- lm(RESPONSE~RATE+DURATION+FEE,data=dat) # Linear main effects regression
> detach(dat)
> summary(mod)
```

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.0073000	0.0005954	12.260	< 2e-16
RATE 0.9%	-0.0034000	0.0005954	-5.710	1.13e-08
RATE 1.9%	-0.0044333	0.0005954	-7.445	9.75e-14
DURATION 15m	0.0008333	0.0005954	1.400	0.162
DURATION 18m	0.0046333	0.0005954	7.781	7.25e-15
FEE 3%	-0.0023111	0.0004862	-4.754	2.00e-06

Residual standard error: 0.07293 on 89994 degrees of freedom
Multiple R-squared: 0.001688, Adjusted R-squared: 0.001632
F-statistic: 30.42 on 5 and 89994 DF, p-value: < 2.2e-16

```
> # Precise p-value of model fit on experimental cells
> F <- as.vector(summary(mod)$fstatistic)
> pf(F[1],F[2],F[3],lower.tail=FALSE)
[1] 5.018023e-31
```

Main effects only, no interactions. RATE 0.0%, DURATION 12m, and FEE 2% embedded in intercept. All coefficients except DURATION 15m are statistically significant with correct signs. Overall better fit on experimental cells than six-cell and nine-cell models.

Eighteen-Cell Linear Regression Model Estimates

```
> prob18 <- read.csv(file="c:/DOE/DESIGN/SIM_FACTORIAL",header=TRUE) # Prob file for all 18 full-factorial cells
> attach(prob18,warn.conflicts=FALSE)
> pred18 <- cbind(prob18,predict(mod,prob18[,-1]))
> detach(prob18)
> names(pred18) <- c("RATE","DURATION","FEE","PROB","EST")
> attach(pred18,warn.conflicts=FALSE)
> print(pred18,digits=3) # Print actual response probs vs estimates for all 18 cells
```

	RATE	DURATION	FEE	PROB	EST
1	0.0%	12m	2%	0.007	0.007300
2	0.9%	12m	2%	0.005	0.003900
3	1.9%	12m	2%	0.003	0.002867
4	0.0%	15m	2%	0.009	0.008133
5	0.9%	15m	2%	0.006	0.004733
6	1.9%	15m	2%	0.004	0.003700
7	0.0%	18m	2%	0.012	0.011933
8	0.9%	18m	2%	0.008	0.008533
9	1.9%	18m	2%	0.005	0.007500
10	0.0%	12m	3%	0.005	0.004989
11	0.9%	12m	3%	0.002	0.001589
12	1.9%	12m	3%	0.001	0.000556
13	0.0%	15m	3%	0.006	0.005822
14	0.9%	15m	3%	0.003	0.002422
15	1.9%	15m	3%	0.002	0.001389
16	0.0%	18m	3%	0.010	0.009622
17	0.9%	18m	3%	0.006	0.006222
18	1.9%	18m	3%	0.004	0.005189

```
>
> I <- 1:18 # Indexes of experimental cells
> # Mean absolute errors for experimental cells, nonexperimental cells, and all cells
> c(mean(abs(PROB[I]-EST[I])),mean(abs(PROB[-I]-EST[-I])),mean(abs(PROB-EST)))
[1] 0.0006160494        NaN        0.0006160494
> # Mean absolute percentage errors for experimental cells, nonexperimental cells, and all cells
> c(mean(abs(PROB[I]-EST[I])/PROB[I]),mean(abs(PROB[-I]-EST[-I])/PROB[-I]),mean(abs(PROB-EST)/PROB))
[1] 0.1563316        NaN        0.1563316
```

- Model covers all 18 factorial cells
- All 18 cells are experimental
- Estimates from regression formula via simple arithmetic on dummy variables
- Error measures are intuitive rather than statistical
- Overall, MAE and MAPE are lower than 6-cell and 9-cell cases but full-factorial experiment may be operationally infeasible

Eighteen-Cell Linear Regression Model with Interactions

```
> dat <- read.csv(file="c:/DOE/DESIGN/SIM_DATA18",header=TRUE) # Input simulation data for 18-cell full factorial case
> attach(dat,warn.conflicts=FALSE)
> mod <- lm(RESPONSE~(RATE+DURATION+FEE)^2,data=dat) # Linear main effects regression with pairwise interactions
> detach(dat)
> summary(mod)
```

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.0067111	0.0009095	7.379	1.61e-13
RATE 0.9%	-0.0012333	0.0011909	-1.036	0.300366
RATE 1.9%	-0.0043000	0.0011909	-3.611	0.000305
DURATION 15m	0.0012000	0.0011909	1.008	0.313618
DURATION 18m	0.0050667	0.0011909	4.255	2.10e-05
FEE 3%	-0.0020222	0.0010871	-1.860	0.062864
RATE 0.9% : DURATION 15m	-0.0017000	0.0014585	-1.166	0.243792
RATE 1.9% : DURATION 15m	0.0009000	0.0014585	0.617	0.537193
RATE 0.9% : DURATION 18m	-0.0022000	0.0014585	-1.508	0.131460
RAT E1.9% : DURATION 18m	-0.0010000	0.0014585	-0.686	0.492949
RATE 0.9% : FEE 3%	-0.0017333	0.0011909	-1.456	0.145530
RATE 1.9% : FEE 3%	-0.0002000	0.0011909	-0.168	0.866628
DURATION 15m : FEE 3%	-0.0002000	0.0011909	-0.168	0.866628
DURATION 18m : FEE 3%	0.0012667	0.0011909	1.064	0.287492

Residual standard error: 0.07293 on 89986 degrees of freedom
Multiple R-squared: 0.00179,   Adjusted R-squared: 0.001646
F-statistic: 12.41 on 13 and 89986 DF, p-value: < 2.2e-16

```
> # Precise p-value of model fit on experimental cells
> F <- as.vector(summary(mod)$fstatistic)
> pf(F[1],F[2],F[3],lower.tail=FALSE)
[1] 1.10961e-27
```

Main effects plus pairwise interactions. RATE 0.0%, DURATION 12m, and FEE 2% embedded in intercept. All pairwise interactions are estimated but none is significant. Overall fit is worse than main effects alone. We do not pursue this model further. In real data, of course, some interaction terms could turn out to be significant, but in our simulated data they are not. This approach provides a good way to assess the significance of interactions. Additional interactions can be examined with (RATE+DURATION+FEE)^3 on the right but again the overall fit is worse.

Key Takeaways

- Multifactor experimentation enables agile, concurrent learning as compared to traditional A/B testing

- Partial factorial experimental designs are easily generated in R, or alternatively in SAS

- Alternative designs can be compared on the front end in terms of coverage and sample size, allowing selection by Marketing

- Analysis of experimental results can be accomplished with linear regression, even for binary responses. Logistic regression yields the same estimates for experimental cells and worse estimates for nonexperimental extrapolations.

- Linear regression allows estimation and evaluation of main effects and interactions

- Linear regression can also be employed in simulation mode on the front end to elucidate design tradeoffs