Screening Designs II

BIOE 498/598 PJ

Spring 2022

Why do we use screening designs?

- Optimization is expensive—many runs/factor at > 2 levels
- Too many factors waste resources
- Too few factors lead to suboptimal results
- **Solution:** A *screening design* tests a large number of factors
- Only active factors are carried forward for optimization

Types of screening designs

- Resolution III Fractional Factorial Design
  - Pro: Mirror image can clear main effects
  - Con: Run size always a power of 2
- PB Design
  - Pro: Run size in multiples of 4
  - Con: Complex aliasing
- Definitive Screening Designs
  - Hybrid screening/optimization design. We'll discuss later!

Plackett-Burman Designs

- Discovered in 1946 while working in the British Ministry of Supply
- Orthogonal designs, so main effects can be estimated independently
- Run sizes in **multiples of 4**
- Both PB designs and FF designs are *Orthogonal Arrays*
  - PB $=$ FF when $N = 2^k$
- PB designs have *complex aliasing*. Every ME is partially confounded with all TWIs.

Creating a PB design (up to 23 factors)

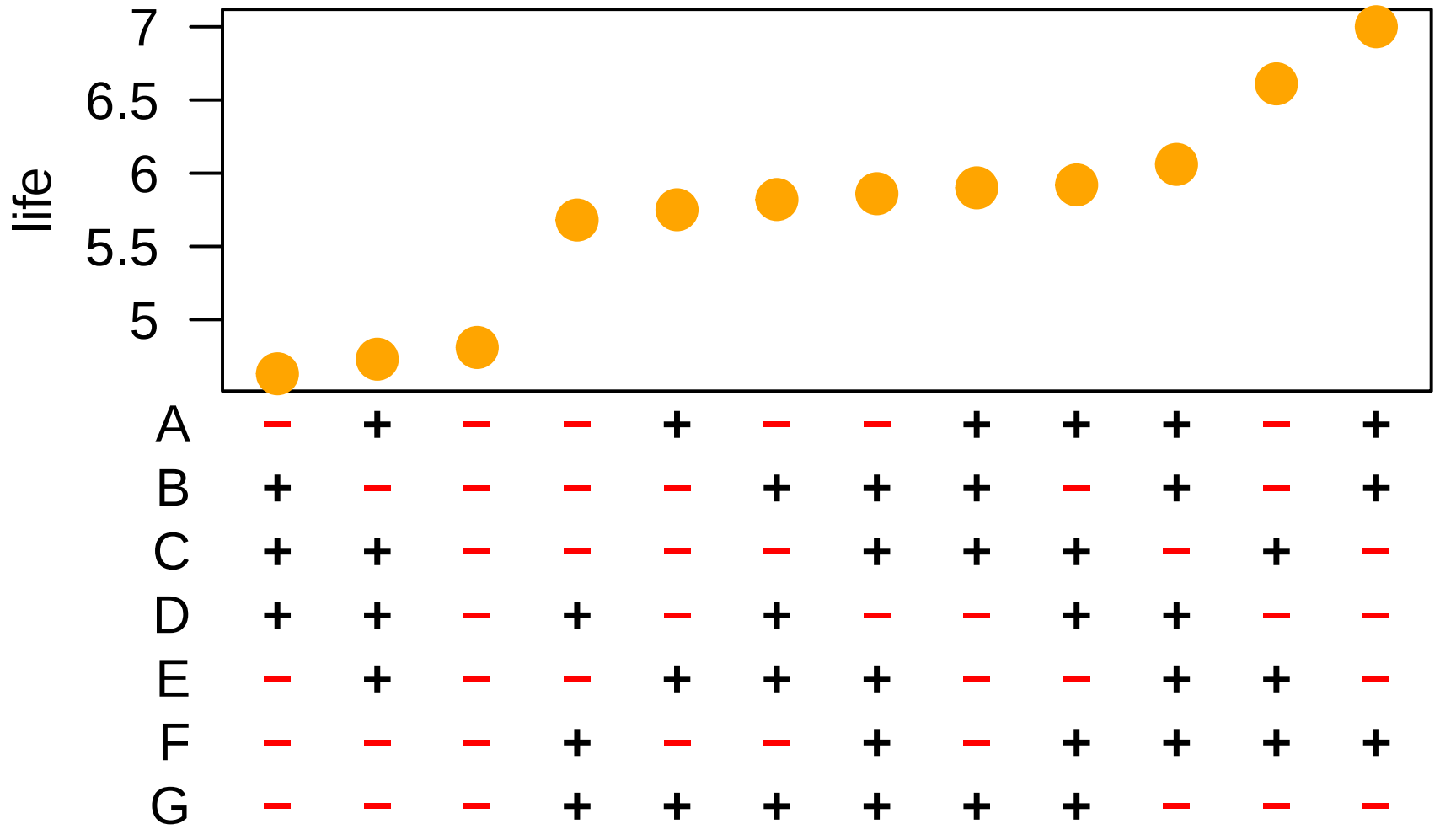
1. Start with the first run from the following table.

Runs	Factor Levels
12	$++-+++---+-$
20	$++--++++-+-+-----++-$
24	$+++++-+-++--++--+-+----$

2. Cycle the factor levels by one to get run #2. Repeat for 11, 19, or 23 runs.
3. Set the final run to all low ($-$).
4. If the number of factors k is less than the number of runs, select the first k columns.

Analyzing PB designs

```
data <- read.csv("PBLife.csv")
farplot(data, factors=c("A","B","C","D","E","F","G"), response="life")
```



Analyzing effects with a linear model

```
model_me <- lm(life ~ A+B+C+D+E+F+G, data=data)
model_int <- lm(life ~ A+B+C+D+E+F+G+c8+c9+c10+c11, data=data)
```

```
show_effects(model_me,
             ordered="abs")
```

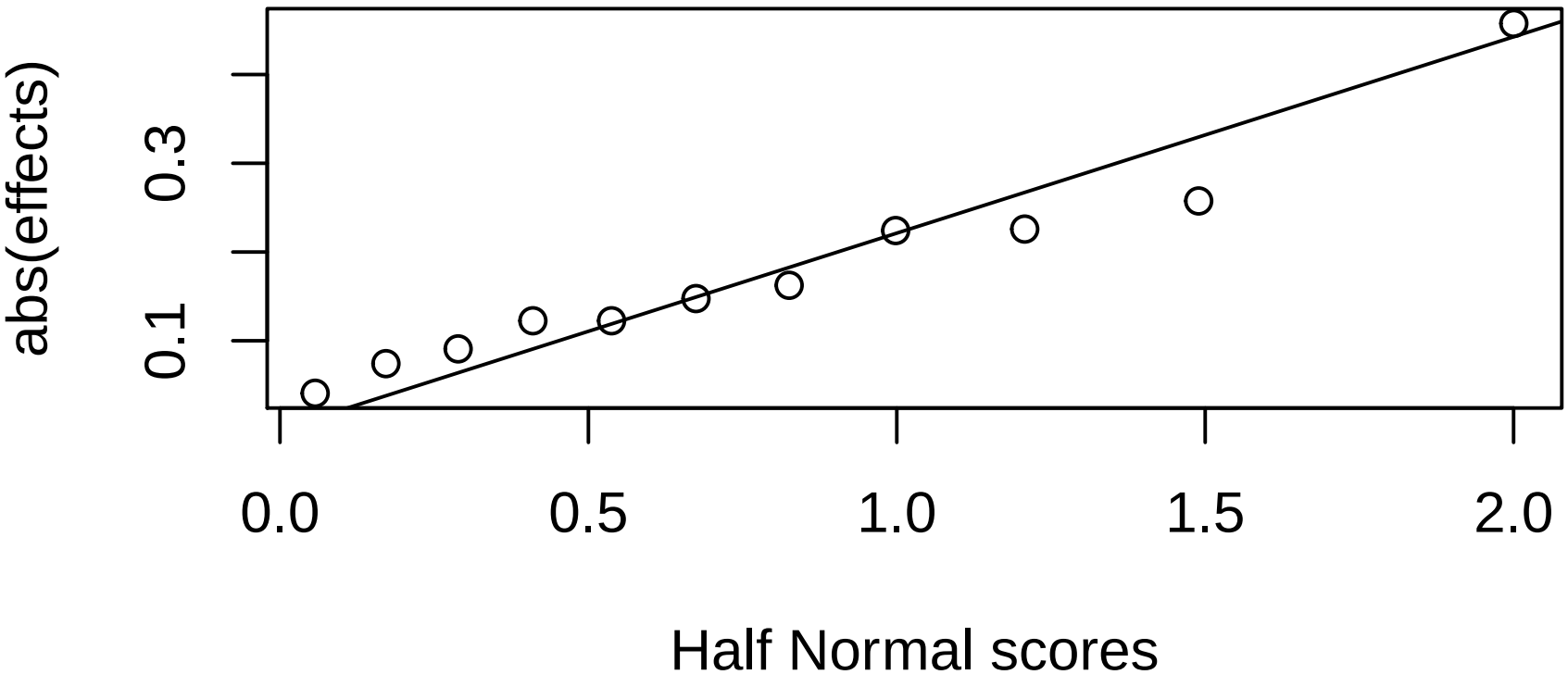
```
## (Intercept)    5.73083
##           F     .4575
##           D    -.2575
##           A     .1625
##           B     .1475
##           C    -.1225
##           G     .09083
##           E     .07417
```

```
show_effects(model_int,
             ordered="abs")
```

```
## (Intercept)    5.73083
##           F     .4575
##           D    -.2575
##          c9     .22583
##          c8     .22417
##           A     .1625
##           B     .1475
##         c11    -.1225
##           C    -.1225
##           G     .09083
##           E     .07417
##         c10     .04083
```

Finding active effects with a half-normal plot

```
halfnorm(get_effects(model_int))
```




```
## zscore= 0.05699967 0.1717471 0.2888094 0.4099833 0.5375191 0.6744898
```

All-subsets regression

- Effect sparsity predicts that few effects will be active.
- If we knew the subset of active effects, we could build a model using all the PB runs.

All-subsets regression

- Effect sparsity predicts that few effects will be active.
- If we knew the subset of active effects, we could build a model using all the PB runs.
- Since we don't know active effects *a prior*, we can build models for all possible subsets.
- We expect that models with the active effects will have the best fit (R^2, AIC, BIC, ...).

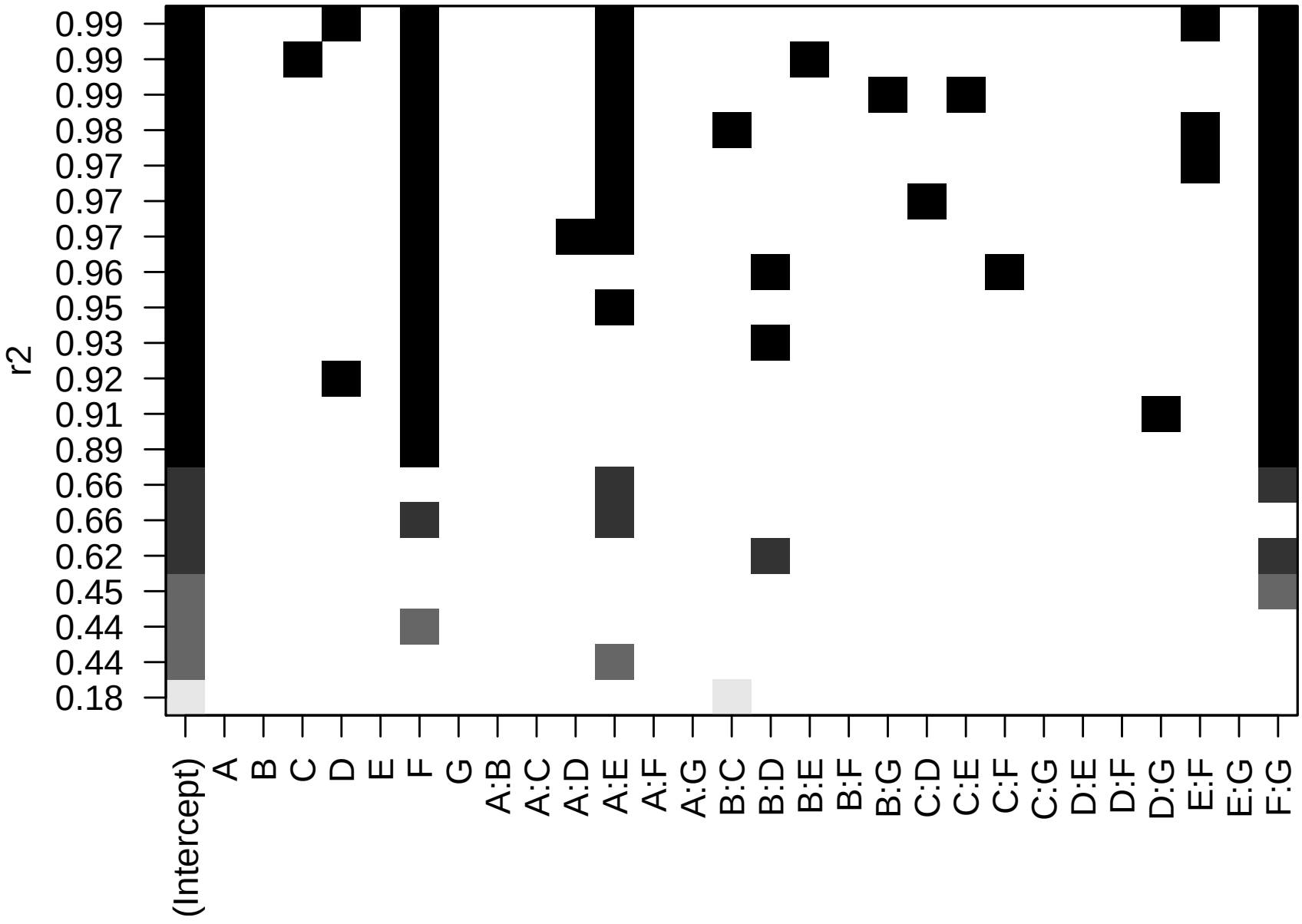
All-subsets regression

- Effect sparsity predicts that few effects will be active.
- If we knew the subset of active effects, we could build a model using all the PB runs.
- Since we don't know active effects *a prior*, we can build models for all possible subsets.
- We expect that models with the active effects will have the best fit (R^2, AIC, BIC, ...).

```
library(leaps)
data$y <- data$life
regsubs <- regsubsets(
  y ~ (.)^2,
  data=data[ ,c("A","B","C","D","E","F","G", "y")],
  method="exhaustive", nvmax=5, nbest=4)
```

Results of all-subsets regression

```
plot(regsubs, scale="r2")
```



A Bayesian approach

We can also calculate the probability that an effect is active given the data using Bayes' rule:

$$p(\beta|D) \propto p(D|\beta)p(\beta).$$

A Bayesian approach

We can also calculate the probability that an effect is active given the data using Bayes' rule:

$$p(\beta|D) \propto p(D|\beta)p(\beta).$$

- $p(\beta|D)$ is the probability that effect β is active given our data $D = (X, y)$.
  - It is called the *posterior probability* since it is calculated *after* we observe D.

A Bayesian approach

We can also calculate the probability that an effect is active given the data using Bayes' rule:

$$p(\beta|D) \propto p(D|\beta)p(\beta).$$

- $p(\beta|D)$ is the probability that effect β is active given our data $D = (X, y)$.
  - It is called the *posterior probability* since it is calculated *after* we observe D.
- $p(D|\beta)$ is the likelihood of observing D given that effect β is active.
  - Likelihood is calculated based on the size of the residuals of a model that includes an active β.

A Bayesian approach

We can also calculate the probability that an effect is active given the data using Bayes' rule:

$$p(\beta|D) \propto p(D|\beta)p(\beta).$$

- $p(\beta|D)$ is the probability that effect β is active given our data $D = (X, y)$.
  - It is called the *posterior probability* since it is calculated *after* we observe D.
- $p(D|\beta)$ is the likelihood of observing D given that effect β is active.
  - Likelihood is calculated based on the size of the residuals of a model that includes an active β.
- $p(\beta)$ is our prior belief about the probability that β is active.
  - Priors need to be specified by the modeler.
  - Common priors are based on effect sparsity, effect hierarchy, and effect heredity. This is a strength of the Bayesian approach!

Bayesian model selection in R

```
X = as.matrix(data[ ,c("A","B","C","D","E","F","G")])
y = data$life
prob <- BsMD::BsProb(X, y, blk=0, mFac=5, mInt=3)
```

```
##   Factor Code  Prob
## 1   none none 0.013
## 2      A   x1 0.008
## 3      B   x2 0.005
## 4      C   x3 0.007
## 5      D   x4 0.050
## 6      E   x5 0.013
## 7      F   x6 0.976
## 8      G   x7 0.964
```