Completely Random Designs

BIOE 498/598 PJ

Spring 2021

Definitions

- **Run** or **Experiment**. A single action with the change of at least one variable followed by an observation.
- **Experimental Unit**. An instance of the item under study that is changed.
- **Replicate**. Two or more runs conducted with the same settings on different experimental units.
  - The respones of replicates vary due to differences inherent in experimental units or the lurking variables.
- **Duplicates**. Multiple measurements on the same experimental unit.
  - Duplicates should always be averaged before analysis.

Definitions (continued)

- **Factor** or **Independent Variable**. The thing under study that can be controlled or changed.
- **Background** or **Lurking Variable**. A thing we are unaware of or cannot control.
- **Response** or **Dependent Variable**. The things we measure. Depends on the settings of the factors and background.
- **Effect**. The change in response due to a change in factor or background.
  - **Calculated effects** come from the model.
  - **Practical effects** come from our knowledge of the system.

Definitions (continued)

- **Experimental Design**. Collection of experiments planned in advance.
- **Confounded Factors**. When a change in factor corresponds with an identical change in another factor.
- **Biased Factors**. When a change in a factor coincides to a change in a lurking variable.
- **Experimental Error**. Difference between observed response and long run average of all experiments with the same settings.
  - There is nothing wrong with error.
  - **Bias error** remains constant or changes consistently.
  - **Random error** changes unpredictably *and averages to zero*.

Overall experiment design

- Use a proper experimental design.
- Randomize as much as you can.

Designs we will study

- Completely Randomized Designs
- Factorial Designs
- Fractional Factorial Designs
- Response Surface Designs
- Screening and Sequential Designs
- Crossover, Mixture, and Split-Plot Designs (briefly)

Three Types of Variables

- **Numerical** (or **continuous**) variables are modeled by real numbers using a single coefficient.
- **Ordinal** variables have discrete but *ordered* levels. If the levels are evenly spaced, we model them using integers.
- **Nominal** (or **categorical**) variables are unordered with no numeric relationship between levels.

One-hot encoding

- In one-hot encoding, a nominal variable with k levels is modeled with k binary dummy variables.
- Only one dummy variable is nonzero ("hot") at a time.
- Example: DNA $\in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$.

$$\beta_{\mathrm{A}} x_{\mathrm{A}} + \beta_{\mathrm{C}} x_{\mathrm{C}} + \beta_{\mathrm{G}} x_{\mathrm{G}} + \beta_{\mathrm{T}} x_{\mathrm{T}}$$

Fitting models with one-hot encoded variables

Consider a model with $x \in \{\text{low, medium, high}\}$:

$$y = \beta_0 + \beta_{\text{low}} x_{\text{low}} + \beta_{\text{med}} x_{\text{med}} + \beta_{\text{high}} x_{\text{high}}$$

which, after fitting is

$$y = 60 + 12 x_{\text{low}} - 20 x_{\text{med}} + 30 x_{\text{high}}$$

where

$$y(x_{\text{low}} = 1) = 72, \quad y(x_{\text{med}} = 1) = 40, \quad y(x_{\text{high}} = 1) = 90$$

Fitting models with one-hot encoded variables

Consider a model with $x \in \{\text{low, medium, high}\}$:

$$y = \beta_0 + \beta_{\text{low}} x_{\text{low}} + \beta_{\text{med}} x_{\text{med}} + \beta_{\text{high}} x_{\text{high}}$$

which, after fitting is

$$y = 60 + 12 x_{\text{low}} - 20 x_{\text{med}} + 30 x_{\text{high}}$$

where

$$y(x_{\text{low}} = 1) = 72, \quad y(x_{\text{med}} = 1) = 40, \quad y(x_{\text{high}} = 1) = 90$$

We could define another model with equivalent predictions:

$$y = 50 + 22 x_{\text{low}} - 10 x_{\text{med}} + 40 x_{\text{high}}$$

Degeneracy

There are infinitely many models with coefficients

$$\beta_0 - \Delta, \quad \beta_{\text{low}} + \Delta, \quad \beta_{\text{med}} + \Delta, \quad \beta_{\text{high}} + \Delta$$

all with the same predictions, residuals, etc.

To avoid the degeneracy, R will not estimate the first (or *base*) level of a factor variable if the model has an intercept. This ensures a unique solution.

Degeneracy in Matrix Form

Consider a design matrix with an intercept, a three-level categorical variable, and two replicates:

$$\mathbf{X} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

This matrix is not full rank since the columns are not linearly independent. ($\mathbf{X}(:,1) = \mathbf{X}(:,2) + \mathbf{X}(:,3) + \mathbf{X}(:,4)$). If we drop any column the matrix will be full rank; R's choice to drop the second column is arbitrary.

Contrasts in the Rothamsted Experiment

The sugar beet experiment is modeled as four treatment variables:

- (A) no fertilizer
- (B) plowed fertilizer in January
- (C) broadcast fertilizer in January
- (D) broadcast fertilizer in April

By default the first treatment (A) will be absorbed into the intercept. The remaining effect sizes are relative to the no fertilizer treatment:

$$\text{yield} = \beta_A + \beta_B x_B + \beta_C x_C + \beta_D x_D$$

Contrasts in the Rothamsted Experiment

- (A) no fertilizer
- (B) plowed fertilizer in January
- (C) broadcast fertilizer in January
- (D) broadcast fertilizer in April

What if we wanted to make other comparisons?

- Effect of broadcast vs. plowed: (C & D) = (B)
- Effect of early vs. late application: (B & C) = (D)
- Effect of any fertilizer: (A) = (B, C, & D)

Contrasts in the Rothamsted Experiment

- (A) no fertilizer
- (B) plowed fertilizer in January
- (C) broadcast fertilizer in January
- (D) broadcast fertilizer in April

What if we wanted to make other comparisons?

- Effect of broadcast vs. plowed: (C & D) = (B)
- Effect of early vs. late application: (B & C) = (D)
- Effect of any fertilizer: (A) = (B, C, & D)

There are all *contrasts*, or comparisons between variables. The null hypotheses for each contrast can be written as a linear combination of the model's variables:

$$\frac{1}{2}\beta_{\mathrm{C}} + \frac{1}{2}\beta_{\mathrm{D}} - \beta_{\mathrm{B}} = 0$$

When specifying contrasts, we require that the coefficient sum to zero (hence the 1/2 factors above).

Rothamsted Sugar Beet Data

```
##    treat  yield
## 1      A 36.900
## 2      A 40.100
## 3      A 37.707
## 4      A 40.093
## 5      B 43.700
## 6      B 46.200
## 7      B 44.000
## 8      B 46.100
## 9      C 48.200
## 10     C 47.700
## 11     C 50.600
## 12     C 48.600
## 13     C 48.900
## 14     D 47.300
## 15     D 49.200
## 16     D 51.300
## 17     D 47.200
## 18     D 48.500
```

How do we test contrasts?

Fit a linear model with a categorical variable with four levels (treatments).

```r
model <- lm(yield ~ treat, data=sugarbeet)
summary(model)
```

```
##
## Call:
## lm(formula = yield ~ treat, data = sugarbeet)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -1.800 -1.075 -0.200  1.175  2.600
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  38.7000     0.7269  53.242  < 2e-16 ***
## treatB        6.3000     1.0279   6.129 2.61e-05 ***
## treatC       10.1000     0.9752  10.357 6.04e-08 ***
## treatD       10.0000     0.9752  10.254 6.84e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
```

How do we test contrasts?

Fit a linear model with a categorical variable with four levels (treatments).

```
model <- lm(yield ~ treat, data=sugarbeet)
```

Let's test the effect of broadcast vs. plowed fertilizer (C & D) = (B). First we define the *contrast coefficients* for the null hypothesis.

```
contrast <- c(0, -1, 0.5, 0.5)
```

How do we test contrasts?

Fit a linear model with a categorical variable with four levels (treatments).

```
model <- lm(yield ~ treat, data=sugarbeet)
```

Let's test the effect of broadcast vs. plowed fertilizer (C & D) = (B). First we define the *contrast coefficients* for the null hypothesis.

```
contrast <- c(0, -1, 0.5, 0.5)
```

Then we use the fit.contrast function from the gmodels package to test the contrast.

```
gmodels::fit.contrast(model, "treat", contrast)
```

```
##                              Estimate Std. Error  t value     Pr(
## treat c=( 0 -1 0.5 0.5 )     3.75  0.8600363 4.360281 0.00065
## attr(,"class")
## [1] "fit_contrast"
```

How do we test contrasts?

Fit a linear model with a categorical variable with four levels (treatments).

```
model <- lm(yield ~ treat, data=sugarbeet)
```

Let's test the effect of broadcast vs. plowed fertilizer (C & D) = (B). First we define the *contrast coefficients* for the null hypothesis.

```
contrast <- c(0, -1, 0.5, 0.5)
```

Then we use the fit.contrast function from the gmodels package to test the contrast.

```
gmodels::fit.contrast(model, "treat", contrast)
```

```
##                              Estimate Std. Error  t value     Pr(
## treat c=( 0 -1 0.5 0.5 )     3.75  0.8600363 4.360281 0.00065
## attr(,"class")
## [1] "fit_contrast"
```

We can also test multiple contrasts at the same time using a contrast matrix as shown in the textbook.

Can we test any contrasts?

No. A contrast must be *estimable* for it to be tested. A contrast is estimable if

- its coefficients sum to zero
- it can be expressed as a linear combination of the rows of the design matrix.

Estimable Example: Measuring only main effects

$$\mathbf{X} = \begin{array}{c} \begin{matrix} \beta_0 & \beta_2 & \beta_3 & \beta_{12} & \beta_{13} & \beta_{23} \end{matrix} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

Any contrast about the interaction terms is not estimable. To test for an β_{12} effect ($H_0 : \beta_{12} - \beta_0 = 0$)

$$c = \begin{pmatrix} 1 & 0 & 0 & -1 & 0 & 0 \end{pmatrix}$$

which is not a combination of the rows in $\mathbf{X}$. In fact, we cannot fit this model since the interaction terms are confounded!

Testing all possible contrasts

As our models grow, the number of possible contrasts increases rapidly. It is likely that at least one random contrast passes our p-value threshold **even if there is not a true difference**.

When testing all contrasts in a model it is wise to adjust your p-value threshold accordingly. A good method is Tukey's HSD. See Section 2.8.2 for an example.

Blocking Factors

Sometimes we are unable to perform all runs on the same day, or with the same reagent batch, or in the same field, etc.

How do we account for differences between each *block* of runs?

Blocking Factors

Sometimes we are unable to perform all runs on the same day, or with the same reagent batch, or in the same field, etc.

How do we account for differences between each *block* of runs?

Solution: Add a **blocking factor** with a level for each block. The blocking factor is included as a main effect in our model.

Blocking Factors: Example

```
data
```

```
## # A tibble: 8 x 4
##         y    x1    x2 block
##     <dbl> <dbl> <dbl> <chr>
## 1  0.780      1     0 day1
## 2  1.35       1     1 day1
## 3  0.438      0     0 day1
## 4  1.30       0     1 day1
## 5 -1.17      -1     0 day2
## 6 -2.18      -1    -1 day2
## 7 -1.31       0    -1 day2
## 8 -0.679      0     0 day2
```

Blocking Factors: Example (continued)

```
##
## Call:
## lm(formula = y ~ block + x1 * x2, data = data)
##
## Residuals:
##         1          2          3          4          5          6
## -0.098989  0.011092 -0.004824  0.092721 -0.076805 -0.011092
##         8
## -0.027008
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.44298    0.08337   5.313  0.01302 *
## blockday2   -1.09504    0.13874  -7.893  0.00424 **
## x1           0.43635    0.08010   5.448  0.01215 *
## x2           0.76785    0.08010   9.586  0.00241 **
## x1:x2       -0.30980    0.09249  -3.349  0.04408 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
##
## Residual standard error: 0.1133 on 3 degrees of freedom
```

What if we forgot to block?

```
##
## Call:
## lm(formula = y ~ x1 * x2, data = data)
##
## Residuals:
##       1       2       3       4       5       6
##  0.0835 -0.1714  0.5427  0.2752 -0.2593  0.1714 -0.0676
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.1045     0.1868  -0.559   0.6057
## x1            0.8014     0.2642   3.033   0.0387 *
## x2            1.1329     0.2642   4.287   0.0128 *
## x1:x2        -0.3098     0.3737  -0.829   0.4537
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
##
## Residual standard error: 0.4577 on 4 degrees of freedom
## Multiple R-squared:  0.932,  Adjusted R-squared:  0.8809
## F-statistic: 18.26 on 3 and 4 DF,  p-value: 0.00848
```

Limitations of Blocking Factors

Blocking factors allow the means of each block to be different. The blocking factor adjusts the intercept for each group of runs

Couldn't the main effects and interactions also be different in each block?

Limitations of Blocking Factors

Blocking factors allow the means of each block to be different. The blocking factor adjusts the intercept for each group of runs

Couldn't the main effects and interactions also be different in each block?

Yes, but there are two issues when correcting this:

1. Adding multiple blocking factors reduces our degrees of freedom, requiring more runs.
2. Some of these effects are confounded with background variables in the block.

Limitations of Blocking Factors

Blocking factors allow the means of each block to be different. The blocking factor adjusts the intercept for each group of runs

Couldn't the main effects and interactions also be different in each block?

Yes, but there are two issues when correcting this:

1. Adding multiple blocking factors reduces our degrees of freedom, requiring more runs.
2. Some of these effects are confounded with background variables in the block.

Our recommendation: Always include a blocking factor when needed, but assume the blocking effects are limited to the intercept.