Random Effect Models

NRES 710

Last compiled: 2024-11-06

## Review of recent material

We have been learning about **multi-variable models** using a generic model:

In theory, you could have unlimited number of X-variables… but in practice, you might get 8 or 10 before it gets nonsensically complicated.

We can have **interactions** (e.g., ).

* Note: Sometimes other fields refer to this in other ways (e.g., ‘effect modification’; where one effect modifies another effect). I mostly see these as ‘interactions’ in environmental science fields.

We have talked about the **advantages** of multi-variable models: eliminates swamping, eliminates bias associated with collinearity (but inflates variance), and allows us to look for interactions.

When we build models with interactions, we should generally include the ‘main effects’ as well (i.e., the individual effects of or ).

## Random-effects Models

My main goal today is to introduce the concept of ‘random-effects models’.

Before we do that, let’s take a step back and temporarily remove interactions from our brain. Interactions are confusing! Instead, let’s again consider more simple models without interactions – and focus on our linear model without interactions.

But now let’s make things a little more complicated again, in a different way. Let’s start with an example.

### Example: Fish Density

Let’s say that we have three variables:

**Y – fish density**  
 **– flow**  
 **– year**

We have gone into a particular stream and measured the abundance of fish at ~eight different stretches in the stream. We have measured flow rate, because we are interested in understanding how flow rates influence fish abundance.

But then we repeated the study over a multiple-year period, and we went back measured fish density and flow rate over a 14-year period!

**2010–2024**

**Q:** does anybody have any idea whether fish populations tend to vary from year to year? Yes, they do: fish density can be quite sensitive to rainfall, flow rate, stream disturbance, whatever. We might want to test whether fish density varies from year-to-year.

How would we include ‘year’ in our model…?

– lots of terms here

Alternatively, we could have written the model like this:

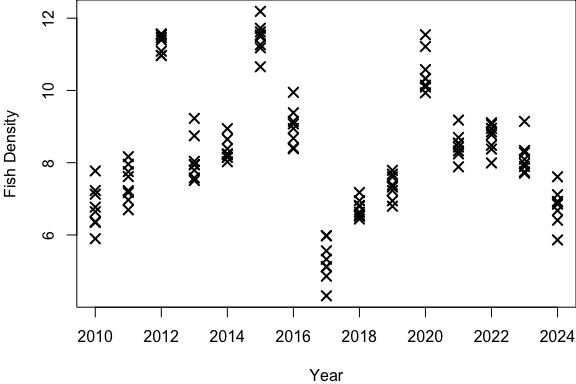
**Q:** What do you think about this second model – is it a good model?

No, because it’s a categorical variable…?

But it’s a number, why can’t we treat this as continuous?

The problem with the second model is we are assuming that there is a linear relationship between year and fish density, where fish density will always be going up or always going down through time. This *might* be an okay assumption… if we have a specific hypothesis suggesting that there might be a linear effect of time on fish abundance.

Graphically, our study design and data might look like:



In this example, the fish abundance bounces all over the place. It’s not going to be a linearly increasing or decreasing relationship. Most of the variation is explained by year in a categorical sense.

**Q:** What do you think of the first model?

It’s LONG and UGLY. It’s not very elegant. And do we really care if… 2013 is different from 2019? Or 2017 is different from 2024?

My point is: **there would be a ton of comparisons we would have to do!** How many pairwise comparisons would a post-hoc test have to do?

A post-hoc test would make 13 + 12 + 11 + 10 + 9 + 8 + 7 + 6 + 5 + 4 + 3 + 2 + 1 interannual comparisons, which equals **91 pairwise comparisons**!

What we really want to do by including **Year** in the model is to *account for variation that occurs from year to year* – and we can do that in a different way than the way described here with the long model.

Here’s how we do that. Instead, we would build this model. (I’m not going to include **Flow** just yet)…

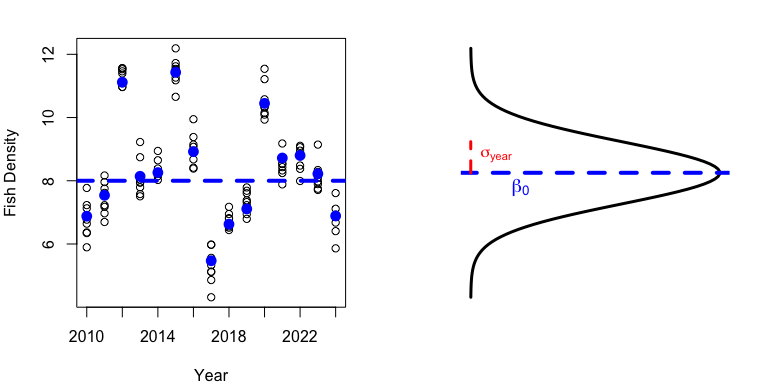
So now we have an equation with **two error terms**: annual yearly error and residual error.

Let’s think about what this means…

#### Annual error

The variation in fish density from year to year can be described by some normal probability distribution with some standard deviation, . We can think about this as: the **between-year variation**.

Graphically, this might be:



The average fish density among all years is (**horizontal blue line**).

*Brian draw long horizontal line to indicate the average among all years*

Within each year, there is also an average fish density (**solid blue dots**), and this varies from year to year.

*Brian draw small horizontal lines to indicate the average density within each year*

The **between-year variation** is normally distributed, which we can visualize with a bell curve that is centered on , has a mean of 0 (so error is both positive and negative on average), and has a standard deviation of .

*Brian draw a sideways bell curve indicating the distribution of error in interannual variation*

#### Residual error

The residual variation is the **within-year variation**.

*Brian illustrate this by drawing little bell curves on the data within each year*.

It’s the amount of variation we have from stretch-to-stretch within any given year. We assume that it’s the same among years.

#### Total model

So now we have a model for how average fish density varies both among years () and within years (). We have an overall average fish density that can vary each year and also vary within years.

What are the **advantages** of the model with between-year variation compared to the model with many annual effects?

* The model with **year as X-variables** has a **ton of variables** that have to be estimated by the model: **14 betas and the residual error**.
* The model with **between-year variation only has 3 parameters**! Average fish density, the standard deviation in variation due to year (i.e., between-year variation), and the standard deviation due to error (i.e., residual standard deviation). Much more elegant.

### Nomenclature

These two models have names!

The model with many X-variables and one residual error term is called a ‘**fixed-effects models**’. Up to this point in class, we have been dealing exclusively with fixed-effects models. This is generally an implicit assumption of any model that someone is talking about (regression, t-test, ANOVA, ANCOVA, whatever) – when they say they did one of those analyses, they typically mean they built a fixed-effects version of that. The model with the many effects of years is a fixed-effect model; year is treated as a fixed-effect.

The model with the two error terms is called a ‘**random-effects model**’. Year is an error term that is treated as a random effect. Since the second model does not have any fixed effects, it is called a random-effects model. Random effects models have no X-variables – they just have random error terms.

If we added in a fixed effect (e.g., flow), this would become a ’**mixed-effects model** – it has both fixed and random effects – both X-variables and random error terms.

### Fixed or random effects?

How do you determine whether variables should be considered as fixed or random effects? **No hard-and-fast rules about this.** Up to you!

But, depending on your variable, there are advantages to treating them as random or fixed. Here are some guidelines:

|  | Fixed Effects | Random Effects |
| --- | --- | --- |
| One | Mean of Y | Variation in Y |
| Two | Understand effects | Account for variation |
| Three | Groups or values are chosen | Values are a sample of what’s possible |
| Four | Limited groups | Many, many groups |

Fixed effects influence the ***mean* of the Y-value** within groups, whereas random effects influence the ***variation* in Y**. This has important implications in your decision.

* If you are interested in understanding what an effect is, you should use a fixed-effects, to understand the effects of the X-variable.
* If you just want to account for variation, you are better off treating a variable as a random effect.

If you chose the treatment levels or treatment groups (i.e., ran an experiment), then it’s better to treat the variable as a fixed-effect. However, if the values are just a sample of what’s possible, then maybe treat the variable as a random-effect.

If you have a limited number of groups (e.g., treatment vs. control), then probably treat it as a fixed effect. If you have many, many groups (e.g., years in a long-term study), then better to treat as a random-variable.

* It’s hard to estimate a standard deviation among a limited number of groups (e.g., two groups), whereas it’s easier to estimate a standard deviation among many group (e.g., 14 years of a study).

But the most important consideration is: **are you interested in how the *X-variable influences the mean*, or *how the variable influences variation*?**

## Analysis in R

We can briefly look at ‘Truth’ to see how the data were analyzed.

### Truth

################### 'Truth' ####################   
### Lecture 15: code to simulate data for class  
  
# Set the seed for reproducibility  
set.seed(123)  
  
# Simulate data  
Year <- rep(1:10, each = 10)  
  
# Between-year error  
MeanYear <- rep(rnorm(10, 0, 10), each = 10)  
  
# Within-year error  
Error <- rnorm(length(Year), 0, 1)  
  
# Response variable  
Abundance <- 50 + MeanYear + Error  
  
# Save the data  
datum <- data.frame(Year = Year, Error = Error, MeanYear = MeanYear, Abundance = Abundance)  
  
# Save the data  
write.csv(datum, "lecture\_15\_dataset1.csv", row.names = FALSE)

If we look at truth (the response variable), the average predicted abundance of fish for any given year is 50!

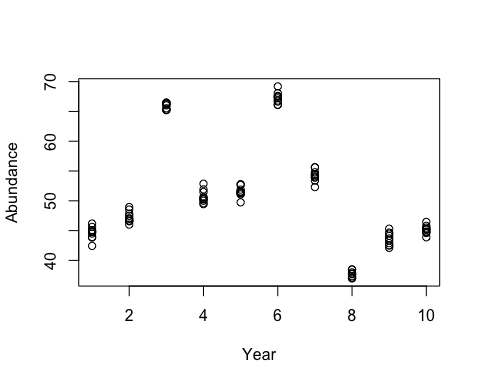
But within any given year, the MeanYear effect has a standard deviation of 10. Since 95% of all data fall within 2 standard deviations, then any given year could be 50 minus 20 or 50 plus 20. So we might expect annual averages to be between 30 and 70.

The term ‘Error’ is random variation for each observation within each year, which is just usual random error with a standard deviation of 1, so that within each year we might see some random error that is 2 units in either direction.

### Code

Let’s look at the data and plot it.

# Read in the data  
datum <- read.csv("lecture\_15\_dataset1.csv")  
  
# Examine it  
head(datum, 15)  
  
# Plot it!  
plot(Abundance ~ Year, data = datum)



When we look at the scatterplot, we can see that the overall average is about 50. For any year, the average varies: it can be increased or decreased by up to 20 fish. And then within the year, the abudnance for any sample can vary by 2-3 in either direction.

We obviously don’t want to run a linear model on this. Year does not have a linear effect! Year does have an effect on abundance, but it’s not a linear effect.

Let’s run a ‘lm()’ but with Year as a fixed, categorical effect.

# Year as a categorical, fixed effect  
results <- lm(Abundance ~ as.factor(Year), data = datum)  
summary(results)

##   
## Call:  
## lm(formula = Abundance ~ as.factor(Year), data = datum)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.1752 -0.6054 0.0049 0.5107 2.1777   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 44.604 0.282 158.34 < 2e-16 \*\*\*  
## as.factor(Year)2 2.670 0.398 6.70 1.7e-09 \*\*\*  
## as.factor(Year)3 21.305 0.398 53.48 < 2e-16 \*\*\*  
## as.factor(Year)4 6.093 0.398 15.29 < 2e-16 \*\*\*  
## as.factor(Year)5 6.911 0.398 17.35 < 2e-16 \*\*\*  
## as.factor(Year)6 22.670 0.398 56.91 < 2e-16 \*\*\*  
## as.factor(Year)7 9.642 0.398 24.20 < 2e-16 \*\*\*  
## as.factor(Year)8 -6.941 0.398 -17.42 < 2e-16 \*\*\*  
## as.factor(Year)9 -1.035 0.398 -2.60 0.011 \*   
## as.factor(Year)10 0.544 0.398 1.36 0.176   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.891 on 90 degrees of freedom  
## Multiple R-squared: 0.991, Adjusted R-squared: 0.99   
## F-statistic: 1.15e+03 on 9 and 90 DF, p-value: <2e-16

This code forces Year to be a factor in a fixed-effect model. This is ugly! All it really tells us is whether each year is significantly different than year 1. Not super useful for telling us about relationships between different years…

What if we were to use a post-hoc test? Which will adjust our p-values to ‘correct’ them…

# Year as a categorical, fixed effect  
results2 <- aov(results)  
TukeyHSD(results2)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = results)  
##   
## $`as.factor(Year)`  
## diff lwr upr p adj  
## 2-1 2.6698 1.37728 3.9623 0.0000  
## 3-1 21.3053 20.01275 22.5978 0.0000  
## 4-1 6.0925 4.79999 7.3850 0.0000  
## 5-1 6.9107 5.61818 8.2032 0.0000  
## 6-1 22.6699 21.37735 23.9624 0.0000  
## 7-1 9.6424 8.34986 10.9349 0.0000  
## 8-1 -6.9414 -8.23390 -5.6489 0.0000  
## 9-1 -1.0353 -2.32782 0.2572 0.2336  
## 10-1 0.5437 -0.74885 1.8362 0.9346  
## 3-2 18.6355 17.34295 19.9280 0.0000  
## 4-2 3.4227 2.13019 4.7152 0.0000  
## 5-2 4.2409 2.94838 5.5334 0.0000  
## 6-2 20.0001 18.70755 21.2926 0.0000  
## 7-2 6.9726 5.68006 8.2651 0.0000  
## 8-2 -9.6112 -10.90370 -8.3187 0.0000  
## 9-2 -3.7051 -4.99762 -2.4126 0.0000  
## 10-2 -2.1261 -3.41865 -0.8336 0.0000  
## 4-3 -15.2128 -16.50528 -13.9202 0.0000  
## 5-3 -14.3946 -15.68708 -13.1020 0.0000  
## 6-3 1.3646 0.07209 2.6571 0.0299  
## 7-3 -11.6629 -12.95540 -10.3704 0.0000  
## 8-3 -28.2466 -29.53916 -26.9541 0.0000  
## 9-3 -22.3406 -23.63308 -21.0480 0.0000  
## 10-3 -20.7616 -22.05411 -19.4691 0.0000  
## 5-4 0.8182 -0.47432 2.1107 0.5642  
## 6-4 16.5774 15.28485 17.8699 0.0000  
## 7-4 3.5499 2.25736 4.8424 0.0000  
## 8-4 -13.0339 -14.32640 -11.7414 0.0000  
## 9-4 -7.1278 -8.42032 -5.8353 0.0000  
## 10-4 -5.5488 -6.84135 -4.2563 0.0000  
## 6-5 15.7592 14.46665 17.0517 0.0000  
## 7-5 2.7317 1.43917 4.0242 0.0000  
## 8-5 -13.8521 -15.14460 -12.5596 0.0000  
## 9-5 -7.9460 -9.23851 -6.6535 0.0000  
## 10-5 -6.3670 -7.65955 -5.0745 0.0000  
## 7-6 -13.0275 -14.32001 -11.7350 0.0000  
## 8-6 -29.6113 -30.90377 -28.3187 0.0000  
## 9-6 -23.7052 -24.99768 -22.4127 0.0000  
## 10-6 -22.1262 -23.41872 -20.8337 0.0000  
## 8-7 -16.5838 -17.87628 -15.2912 0.0000  
## 9-7 -10.6777 -11.97019 -9.3852 0.0000  
## 10-7 -9.0987 -10.39123 -7.8062 0.0000  
## 9-8 5.9061 4.61357 7.1986 0.0000  
## 10-8 7.4850 6.19253 8.7776 0.0000  
## 10-9 1.5790 0.28645 2.8715 0.0055

These p-values are inflated! Dislike. Buuut… we don’t care about this – because we are actually interested in how year influences *variation*…

I’ve got some bad news… We can’t use ‘lm()’ anymore!! It’s gotten us all the way through two months of class, but we have hit it’s limits. It cannot handle random effects!

We have to move onto a new model called an ‘lme()’ which is in the ‘nlme’ package.

# We don't have to install this package - it now comes defaul in R!  
library(nlme)  
  
# Examine the help file

‘nlme’ – non-linear mixed effects

# Examine the help file  
help(lme)

“A generic function that fits a linear fixed effects model…”

Requires a fixed model – which we are used to using.

Data = datum

Random variables! We have to specify what the random variables are now.

Rest of the stuff you can mostly ignore.

# Fit the model  
results <- lme(Abundance ~ 1, data = datum, random = ~1|Year)  
summary(results)

## Linear mixed-effects model fit by REML  
## Data: datum   
##   
## Random effects:  
## Formula: ~1 | Year  
## (Intercept) Residual  
## StdDev: 9.539 0.8908  
##   
## Fixed effects: Abundance ~ 1   
##   
## Standardized Within-Group Residuals:  
## Min Q1 Med Q3 Max   
## -2.447929 -0.678133 0.001199 0.585246 2.444519   
##   
## Number of Observations: 100  
## Number of Groups: 10

We don’t have any fixed effects for our model! No X-variables with fixed effects so far. So we can just specify 1 to tell the model we want to know what the average fish abundance is – and how it varies in response to the random effect.

What the tilde means is that… we want a random change in the *intercept* due to year.

Let’s discuss the output.

**Linear mixed-effects model fit by REML.** These models are no longer fit by the sum of squared error (SSE). We have now moved into the realm of Maximum Likelihood (‘restricted maximum likelihood’; the default). This is ~distinct from SSE models and gives us distinct metrics (AIC, BIC, etc.).

The **coefficients table** that we are used to is the ~same. This model is super simple, so it just gives us the intercept (in this case, fish abundance), including the estimate, SE, t-value, and p-value.

The **random effects** information tells us about our two error terms – the two sigma terms.

* The intercept (poorly termed) is the **random year effect** (). Truth was 10; it estimated 9.53.
* The residual error term is our **residual error** (). Truth was 1; it estimated 0.89.

I wish it labeled the random intercept as “year”, but instead it just says (Intercept). Not super useful…

Questions?

## Another Example

### Field experiment

Let’s assume we have 15 fields where we are doing a good, old-fashioned, classic agricultural experiment! Within each plot, we have a treatment and a control.

*Brian draw this on the board: field 1, field 2, field 3, … , field 15: each field with a C and T plot within it that are randomly top or bottom of field*

**Q:** How many samples do we have? **30**: a treatment and a control within each of 15 plots.

We could run this model:

**Q:** What’s do normal people call this? **T-test!**

Is there anything we should be worried about for this? What about swamping?

Swamping – when the effect of one X-variable is masked out by another effect.

Should we be worried about swamping here?

What if things that are different in each field (drainage, sunlight, deer abundance, etc.) influence biomass and thus prevent us from learning about our effect of interest: treatment vs. control. If so, we then we should deal with swamping.

**Q:** How do we normally deal with swamping? We include field in the model! But that requires us to add 14 field effects!! That’s impractical and not elegant. Plus we don’t care about how fields influence biomass, we mostly want to control for that variation while examining the effect of treatment.

The answer to this problem is to run a **mixed-effects model**:

Here, is the *between-field variation*, while is the *within-field variation*.

We can partition the treatment effect from random, noise effects among plots.

The first model has a fancy name: t-test. This second, more complicated model also has a fancy name: **paired t-test**! By including the field effect, we have tied each pair of treatment and control samples within each field together. Without that field effect, we lump all of our samples into giant pools of treatment and control and then compare them.

This second model is **much more powerful**! We have mathematically eliminated a lot of residual, eliminated swamping, and accounted for residual variation due to field by only adding one parameter: sigma ().

This experimental design also has a fancy term: **randomized block design**. The fields are blocks; we have ‘blocked’ by field. And within each field, we have randomized treatment assignment. Randomized block designs are very, very powerful!! By eliminating swamping.

We could included field-effect as an X-variable, but it’s much better to do it this way!!

Let’s look at another dataset.

### Truth

Here is how the data were simulated:

### Dataset 2  
# Set the seed for reproducibility  
set.seed(123)  
  
# Simulate data  
Field <- rep(1:15, each = 2)  
Treatment <- rep(0:1, 15)  
  
# Field error  
FieldError <- rep(rnorm(15, 0, 5), each = 2)  
  
# Residual error  
Error <- rnorm(length(Field), 0, 1)  
  
# Response variable  
Biomass <- 25 + 10\*Treatment + FieldError + Error  
  
# Save the data  
datum <- data.frame(Field = Field, Treatment = Treatment, FieldError = FieldError, Error = Error, Biomass = Biomass)  
  
# Save the data  
write.csv(datum, "lecture\_15\_dataset2.csv", row.names = FALSE)

There are 2 samples from 15 fields, Treatment is dummy-coded, Field Error has a standard deviation of 5, residual error has a standard deviation of 1, and Biomass is 25 + 10\*treatment + the two error terms.

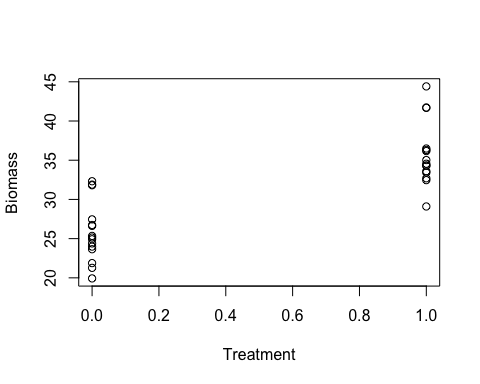
Truth:

* – average biomass in control
* – difference in treatment and control
* – error associated with fields
* – residual error

### Analysis

Load it into R:

# Load the data  
datum <- read.csv("lecture\_15\_dataset2.csv")  
  
# Examine  
head(datum)  
  
# Plot  
plot(Biomass ~ Treatment, data = datum)



Now remember, our observations within each field are *paired*. Some researchers will connect the dots in each field to indicate this – or change the pixels to be paired by plot.

Let’s run a t-test:

# T-test  
results <- lm(Biomass ~ Treatment, data = datum)  
summary(results)

##   
## Call:  
## lm(formula = Biomass ~ Treatment, data = datum)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -6.649 -2.169 -0.817 0.886 8.663   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 25.82 1.01 25.55 < 2e-16 \*\*\*  
## Treatment 9.93 1.43 6.94 1.5e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 3.91 on 28 degrees of freedom  
## Multiple R-squared: 0.633, Adjusted R-squared: 0.62   
## F-statistic: 48.2 on 1 and 28 DF, p-value: 1.5e-07

We are done! Or we could say that. This is sufficient. It is statistically significant. But we are leaving a lot of variation unexplained that we can easily explain with a better model.

Estimate is 9.9, and confidence intervals are about ~3, which overlaps with Truth (10). But that’s a pretty big confidence interval. Residual standard error: 6.87. All of the error is being chalked up as residual error.

There is nothing wrong with this, but our power is low due to the large confidence intervals.

Let’s try a LME.

# LME; i.e., paired T-test  
results2 <- lme(Biomass ~ Treatment, data = datum, random = ~1|Field)  
summary(results2)

## Linear mixed-effects model fit by REML  
## Data: datum   
##   
## Random effects:  
## Formula: ~1 | Field  
## (Intercept) Residual  
## StdDev: 3.813 0.888  
##   
## Fixed effects: Biomass ~ Treatment   
## Correlation:   
## (Intr)  
## Treatment -0.16   
##   
## Standardized Within-Group Residuals:  
## Min Q1 Med Q3 Max   
## -1.61959 -0.52825 0.00999 0.42882 1.68140   
##   
## Number of Observations: 30  
## Number of Groups: 15

Examine the field effect between the two analyses. Mostly unchanged!! Zero effect on our effect estimates. But the SE and 95% CI are much reduced now. Confidence intervals went from 3 to 0.6! Big reduction. And our p-values are drastically smaller.

We have now split the error into the two error terms: error due to field, and error due to randomness in observations. Residual error reduced from ~4 to ~1 (Truth), by assigning that error to the field (intercept: 3.8).

No p-value associated with the effect of field! We did not test for this as a fixed-effect, we did not estimate a beta for field, so it doesn’t get a statistical test with a p-value. We are mostly interested in effect of treatment, not an effect of field.

If we wanted to test for an effect of field, we could run an F-drop test! We’ll do that during the next class.

## Truth

Again, here is Truth, which we always have at the end of each lecture.

################### 'Truth' ####################   
### Lecture 15: code to simulate data for class  
  
# Set the seed for reproducibility  
set.seed(123)  
  
# Simulate data  
Year <- rep(1:10, each = 10)  
  
# Between-year error  
MeanYear <- rep(rnorm(10, 0, 10), each = 10)  
  
# Within-year error  
Error <- rnorm(length(Year), 0, 1)  
  
# Response variable  
Abundance <- 50 + MeanYear + Error  
  
# Save the data  
datum <- data.frame(Year = Year, MeanYear = MeanYear, Error = Error, Abundance = Abundance)  
  
# Save the data  
write.csv(datum, "lecture\_15\_dataset1.csv", row.names = FALSE)  
  
  
### Dataset 2  
# Set the seed for reproducibility  
set.seed(123)  
  
# Simulate data  
Field <- rep(1:15, each = 2)  
Treatment <- rep(0:1, 15)  
  
# Field error  
FieldError <- rep(rnorm(15, 0, 5), each = 2)  
  
# Residual error  
Error <- rnorm(length(Field), 0, 1)  
  
# Response variable  
Biomass <- 25 + 10\*Treatment + FieldError + Error  
  
# Save the data  
datum <- data.frame(Field = Field, Treatment = Treatment, FieldError = FieldError, Error = Error, Biomass = Biomass)  
  
# Save the data  
write.csv(datum, "lecture\_15\_dataset2.csv", row.names = FALSE)

[–go to next lecture–](lecture_16.html)