ANOVA Models with a Block

Neil J. Hatfield

3/25/2021

In this tutorial, we are going to explore using a block in One-way layouts.

Context

A farmer wants to test out four varieties of barley and see if there is any difference in yield (bussels per acre). He has four fields in which he can plant the barley. However, the farmer is aware of differences between each field. For example,

- One field has a higher clay content in the soil than the others
- One field has rockier soil than the others
- Two fields are in wetter climates; two are in drier climates
- One field has loose soil while another field has much more compacted soil.
- Two fields are relatively flat, one has a hill in the middle, and the last has a valley.

Given that the fields will be our measurement units, there is quite a bit of variation between them. This variation could easily become confounded with the impact of barley variety on crop yield. Thus, we're in a perfect situation to make use a block of field.

(Details of the study design are in the slides.)

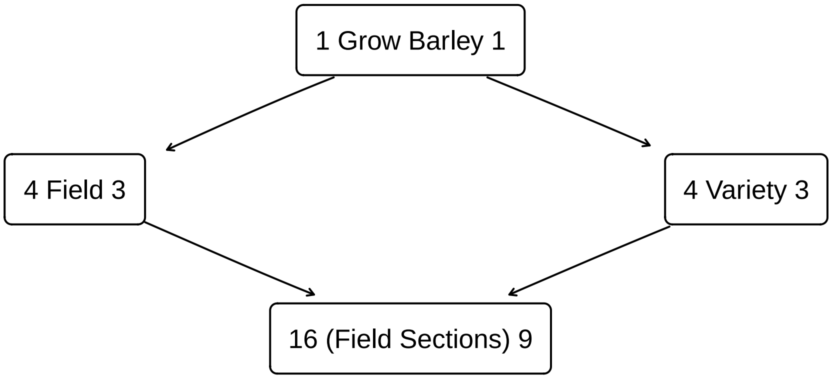

Figure 1: Hasse Diagram for Barley Crop Yield Study

Fit the Model

Since are varietals and fields are fixed effects, we don't need any special packages (i.e., `lme4`) to fit our model. Rather we will go ahead and use the `aov` function.

```r
# Load Data
barley <- read.table(
  file = "https://raw.github.com/neilhatfield/STAT461/master/dataFiles/barley.dat",
  header = TRUE,
  sep = ","
)

# Glancing at the data frame, we need to go ahead and set
# Treatment to a factor and change the name to Varietal
# I'm also going to simplify Planting.Harvesting.Order to Order

barley$Treatment <- as.factor(barley$Treatment)
names(barley)[3] <- "Varietal"
names(barley)[4] <- "Order"

# This will seem counter-intuitive, but we need to tell R
# to view Field also like a factor. This is so we can get
# estimates.

barley$Field <- as.factor(barley$Field)

# Fixed effects model with block
barleyModel <- aov(
  formula = Yield ~ Varietal + Field,
  data = barley
)
```

Check Assumptions

In terms of assumptions, we still have our core three: Gaussian Residuals, Homoscedasticity, and Independence of Observations. With the block, we also want to check an interaction plot to make sure that there isn't anything strange/unexpected going on when you plant a particular varietal in a particular field.

Gaussian Residuals

Use a QQ plot like usual:

```
# QQ plot for residuals
car::qqPlot(
  x = residuals(barleyModel),
  distribution = "norm",
  envelope = 0.90,
  id = FALSE,
  pch = 20,
  ylab = "Residuals (BPA)"
)
```

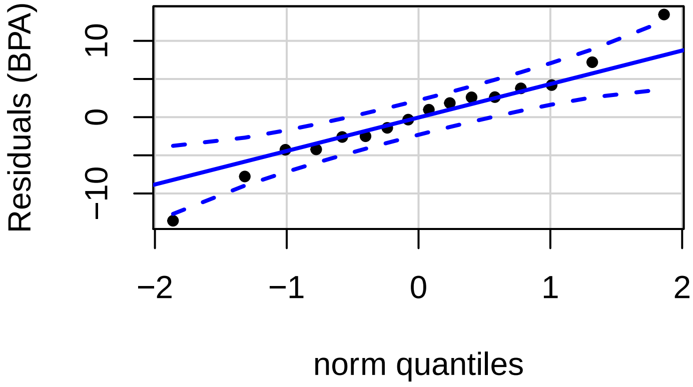

Figure 2: QQ Plot for Residuals

There is very little that is concerning in the QQ plot, even with the two points just outside of the envelope. We will take the Gaussian assumption to be essentially satisfied.

Homoscedasticity

When we account for the the block in our model, we won't get the nice strips that we're used to seeing. However, we can still use the basic idea of the strip chart to assess homoscedasticity. We can still look for patterns and we'll add a smoother to our plot, transforming the strip chart to a Tukey-Anscombe Plot.

```
# Strip chart, notice the calls to the random effects model
ggplot(
  data = data.frame(
    residuals = residuals(barleyModel),
    fitted = fitted.values(barleyModel)
  ),
  mapping = aes(x = fitted, y = residuals)
) +
  geom_point(size = 2) +
  geom_hline(
    yintercept = 0,
    linetype = "dashed",
    color = "grey50"
  ) +
  geom_smooth(
    formula = y ~ x,
    method = stats::loess,
    method.args = list(degree = 1),
    se = FALSE,
    size = 0.5
  ) +
  theme_bw() +
  xlab("Fitted values (BPA)") +
  ylab("Residuals (BPA)")
```

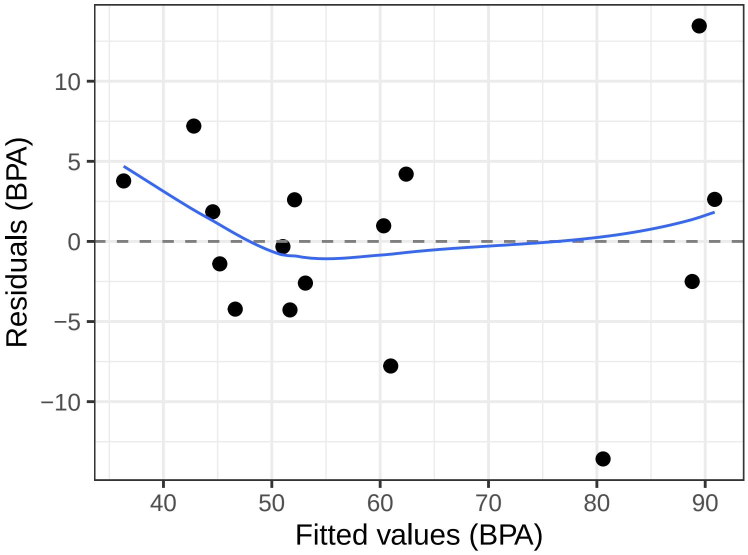

Figure 3: Tukey-Anscombe Plot for Barley Crop Yield Study

We don't see any patterns to the plot, which is a good sign. While there does appear to be more variation on the high end of the fitted values, the smoothed line is fairly flat. The upticks at the ends might be the result of the small sample size. In all, we will take the homoscedasticity assumption to be satisified.

Independence of Observations

```
# Index Plot for Barley Yields
ggplot(
  data = barley,
  mapping = aes(
    x = Order,
    y = Yield,
    color = Field,
    shape = Varietal
  )
) +
  geom_point(size = 3) +
  geom_path(
    mapping = aes(group = Field)
  ) +
  ggplot2::theme_bw() +
  xlab("Planting/Havesting Order") +
  ylab("Yield (BPA)")
```

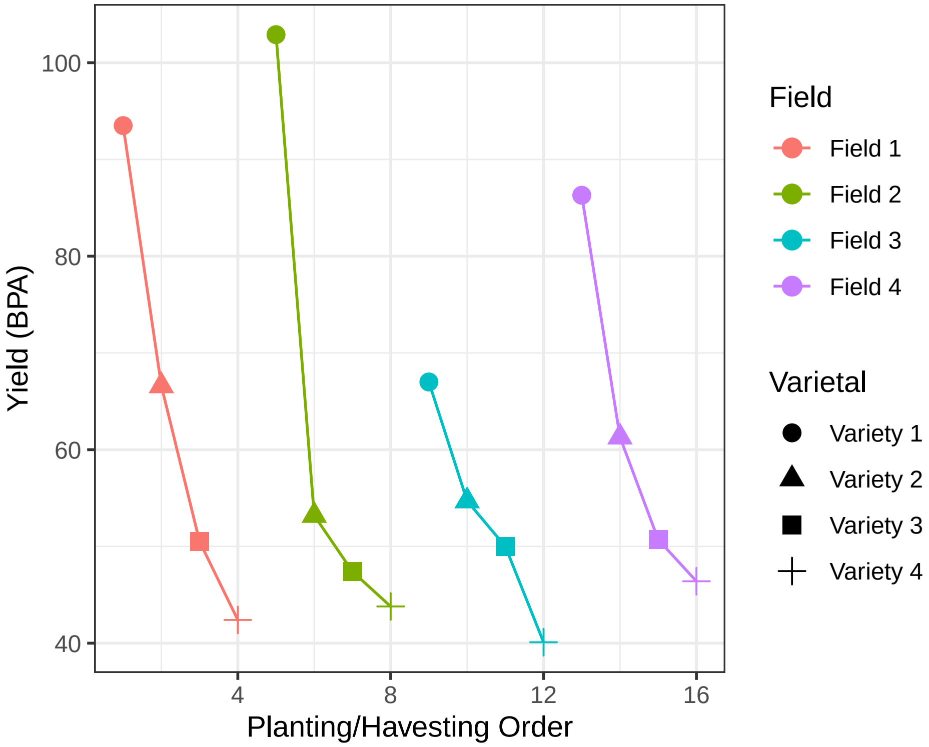

Figure 4: Index Plot for Barley Crop Yield Residuals

While at first glance, there appears to be a pattern to the response, we can see that this is an artifact of the block and treatments.

Interaction of Field and Varietal

The last aspect we need to check is whether there is an interaction between our block and our factor. If there is an interaction, then our model (response = factor + block) is not valid. We will make an interaction plot to assess this issue.

```r
# Interaction Plot for Field and Treatment
ggplot2::ggplot(
  data = barley,
  mapping = aes(
    x = Varietal,
    y = Yield,
    color = Field,
    group = Field
  )
) +
  ggplot2::geom_point(size = 2) +
  ggplot2::geom_line(size = 1) +
  ggplot2::theme_bw() +
  xlab("Variety") +
  ylab("Yield (BPA)") +
  labs(color = "Field") +
  theme(
    legend.position = "right"
  )
```

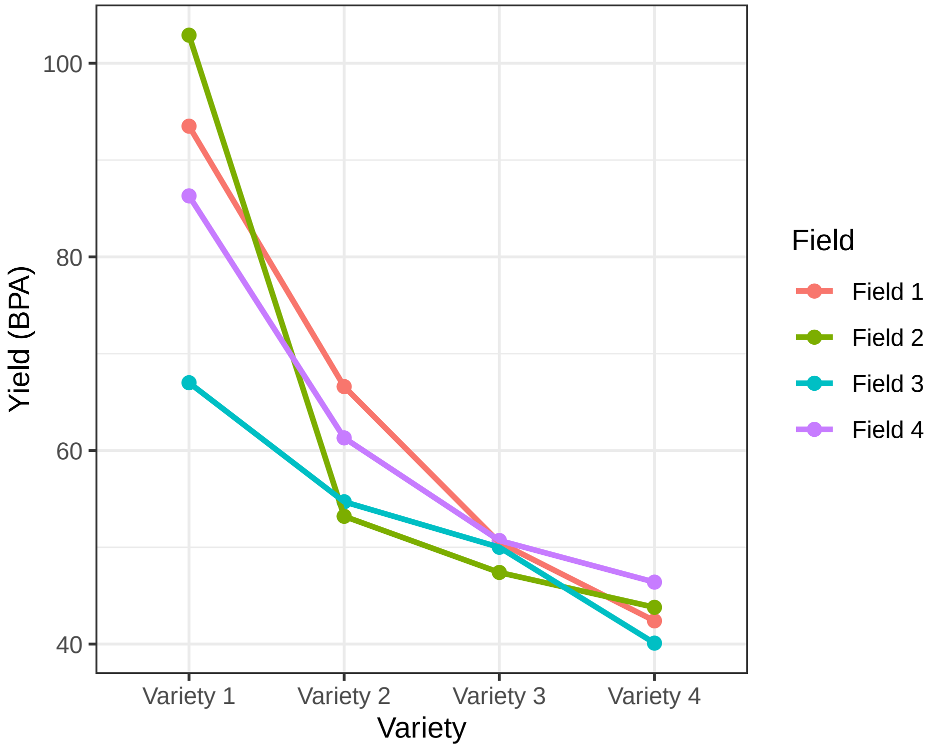

Figure 5: Interaction Plot for Barley Varietal and Field

In the interaction plot, we see essentially the same behavior of barley variety in each field. This indicates that there is not any type of interaction between field and barley varietal.

Results

Remember, there are essentially two parts to results: the omnibus test and the post hoc analysis.

Omnibus Results

```
# Omnibus Test/Modern ANOVA Table
parameters::model_parameters(
    model = barleyModel,
    omega_squared = "partial",
    eta_squared = "partial",
    epsilon_squared = "partial"
) %>%
  knitr::kable(
    digits = 4,
  col.names = c("Source", "SS", "df", "MS", "F", "p-value",
                "Partial Omega Sq.", "Partial Eta Sq.", "Partial Epsilon Sq."),
  caption = "ANOVA Table for Barley Crop Yield Study",
  align = c('l',rep('c',8)),
  booktab = TRUE
) %>%
  kableExtra::kable_styling(
    bootstrap_options = c("striped", "condensed"),
    font_size = 10,
    latex_options = c("scale_down", "HOLD_position")
  )
```

Table 1: ANOVA Table for Barley Crop Yield Study

Source	SS	df	MS	F	p-value	Partial Omega Sq.	Partial Eta Sq.	Partial Epsilon Sq.
Varietal	4573.105	3	1524.3683	23.7116	0.0001	0.8098	0.8877	0.8503
Field	259.265	3	86.4217	1.3443	0.3203	0.0606	0.3094	0.0793
Residuals	578.590	9	64.2878					

Remember, we need to ignore the Field row's *p*-value (and effect sizes) as R automatically assumes that the block effect is of interest.

In addition, we can look at the relative efficiency of our block:

```
# Relative Efficiency of Block
block.RelEff(
  aov.obj = barleyModel,
  blockName = "Field",
  trtName = "Varietal"
)
```

```
## [1] "The relative efficiency of the block, Field, is 1.028."
```

Note: you can run this code inline to not have the result appear as raw output: The relative efficiency of the block, Field, is 1.028.

Post Hoc Analysis

Point Estimates If we want to get point estimates for our Grand Mean, Treatment Effects, and Block Effects, we can.

```
# Point Estimates for Insect Sprays
pEst <- dummy.coef(barleyModel)
pEst <- unlist(pEst)
names(pEst) <- c(
  "Grand Mean",
  levels(barley$Varietal),
  levels(barley$Field)
  )

data.frame("Estimate" = pEst) %>%
  knitr::kable(
  digits = 3,
  caption = "Point Estimates from the Barley Crop Yield Study",
  booktabs = TRUE,
  align = "c"
  ) %>%
  kableExtra::kable_styling(
    font_size = 12,
    latex_options = c("HOLD_position")
  )
```

Table 2: Point Estimates from the Barley Crop Yield Study

	Estimate
Grand Mean	59.800
Variety 1	27.625
Variety 2	-0.850
Variety 3	-10.150
Variety 4	-16.625
Field 1	3.450
Field 2	2.025
Field 3	-6.850
Field 4	1.375

Pairwise Comparisons From the slides, we noted that we were going to control the MEER at 0.1 by using Tukey's HSD. Thus, we will do so for our pairwise comparisons.

```r
# Post Hoc Analysis--Pairwise comparisons
barleyPH <- TukeyHSD(
  x = barleyModel,
  which = "Varietal", #We need to specify which model term we want
  conf.level = 0.9
)

knitr::kable(
  barleyPH$Varietal,
  digits = 4,
  caption = "Post Hoc Tukey HSD Comparisons",
  col.names = c("Difference", "Lower Bound",
                "Upper Bound", "Adj. p-Value"),
  align = 'cccc',
  booktabs = TRUE
) %>%
  kableExtra::kable_styling(
    bootstrap_options = c("condensed", "boardered"),
    font_size = 12,
    latex_options = "HOLD_position"
  )
```

Table 3: Post Hoc Tukey HSD Comparisons

	Difference	Lower Bound	Upper Bound	Adj. p-Value
Variety 2-Variety 1	-28.475	-43.5533	-13.3967	0.0033
Variety 3-Variety 1	-37.775	-52.8533	-22.6967	0.0004
Variety 4-Variety 1	-44.250	-59.3283	-29.1717	0.0001
Variety 3-Variety 2	-9.300	-24.3783	5.7783	0.4053
Variety 4-Variety 2	-15.775	-30.8533	-0.6967	0.0833
Variety 4-Variety 3	-6.475	-21.5533	8.6033	0.6747

Effect Sizes

To get the effect sizes for our desired pairwise comparisons, we will turn to the `anova.PostHoc` function. However, we need to take care with the arguments of this function:

- The `aov.obj` is the primary argument and is where we pass the `aov` (or `lm`) output to.
- The `response` argument takes a character string as the name of the response attribute.
- The `mainEffect` argument takes a character string that names the factor you are wanting to do pairwise comparisons on.

```
anova.PostHoc(
  aov.obj = barleyModel, # Our aov output
  response = "Yield", # Our response variable
  mainEffect = "Varietal" # Our factor variable
) %>%
  knitr::kable(
    digits = 3,
    caption = "Post Hoc Comparison Effect Sizes",
    col.names = c("Pairwise Comparison","Cohen's d", "Hedge's g",
                  "Prob. Superiority"),
    align = 'lccc',
    booktabs = TRUE
  ) %>%
  kableExtra::kable_styling(
    bootstrap_options = c("condensed", "boardered"),
    font_size = 12,
    latex_options = "HOLD_position"
  )
```

Table 4: Post Hoc Comparison Effect Sizes

Pairwise Comparison	Cohen's d	Hedge's g	Prob. Superiority
Variety.1 vs. Variety.2	2.451	2.131	0.958
Variety.1 vs. Variety.3	3.493	3.037	0.993
Variety.1 vs. Variety.4	4.052	3.523	0.998
Variety.2 vs. Variety.3	2.061	1.792	0.927
Variety.2 vs. Variety.4	3.313	2.881	0.990
Variety.3 vs. Variety.4	3.005	2.613	0.983

We need to make an important note here: suppose that our overall Unusualness Threshold was 0.1. From the Tukey HSD results, Variety 3 vs 2, 4 vs 2, and 4 vs 3 would NOT be statistically significant. From the effect size table, we would say that there are rather large effects. Just as effect sizes temper statistical significance, statistical significance moderates effect sizes. In these three cases, while there appears to be a large effect, there is enough variation in those groups that the effect is not statistically large enough.

Contrasts

Even with a block, we can still use make use of the idea of contrasts. For example, let's say that barley varieties 1 and 2 are from one company while 3 and 4 are from a second company. We can test the contrast of companies, even in this blocking design.

```
# Define the contrast
company <- c(1/2, 1/2, -1/2, -1/2)
```

```r
# Bind the contrast to our factor
contrasts(barley$Varietal) <- company

# Refit our model so that our contrast gets tested
barleyContrast <- aov(
  formula = Yield ~ Varietal + Field,
  data = barley
)

# Get the updated ANOVA Table
## Remember, you could also use the DescTools package for Scheffé here
conOut <- summary.aov(
  object = barleyContrast,
  split = list(
    Varietal = list(
      "Company A vs. Company B" = 1
    )
  )
)

# Make a nice table
knitr::kable(
  x = conOut[[1]],
  digits = 4,
  col.names = c(
    "DF", "SS", "MS", "F", "p-value"),
  caption = "ANOVA Table for Barley Crop Yield Contrasts",
  booktabs = TRUE,
  align = rep("c", 5)
) %>%
  kableExtra::kable_styling(
    font_size = 12,
    latex_options = c("HOLD_position")
  )
```

Table 5: ANOVA Table for Barley Crop Yield Contrasts

	DF	SS	MS	F	p-value
Varietal	3	4573.105	1524.3683	23.7116	0.0001
Varietal: Company A vs. Company B	1	2867.602	2867.6025	44.6057	0.0001
Field	3	259.265	86.4217	1.3443	0.3203
Residuals	9	578.590	64.2878		

Code Appendix

```
# Setting Document Options
knitr::opts_chunk$set(
  echo = FALSE,
  warning = FALSE,
  message = FALSE,
  fig.align = "center"
)

packages <- c("tidyverse", "knitr", "kableExtra",
              "parameters", "hasseDiagram", "DescTools")
lapply(packages, library, character.only = TRUE)

options(knitr.kable.NA = "")
options(contrasts = c("contr.sum", "contr.poly"))

source("https://raw.github.com/neilhatfield/STAT461/master/rScripts/ANOVATools.R")
modelLabels <- c("1 Grow Barley 1", "4 Field 3", "4 Variety 3", "16 (Field Sections) 9")
modelMatrix <- matrix(
  data = c(FALSE, FALSE, FALSE, FALSE, TRUE, FALSE, FALSE, FALSE, TRUE, FALSE, FALSE,
           FALSE, TRUE, TRUE, TRUE, FALSE),
  nrow = 4,
  ncol = 4,
  byrow = FALSE
)
hasseDiagram::hasse(
 data = modelMatrix,
 labels = modelLabels
)

# Load Data
barley <- read.table(
  file = "https://raw.github.com/neilhatfield/STAT461/master/dataFiles/barley.dat",
  header = TRUE,
  sep = ","
)

# Glancing at the data frame, we need to go ahead and set
# Treatment to a factor and change the name to Varietal
# I'm also going to simplify Planting.Harvesting.Order to Order

barley$Treatment <- as.factor(barley$Treatment)
names(barley)[3] <- "Varietal"
names(barley)[4] <- "Order"

# This will seem counter-intuitive, but we need to tell R
# to view Field also like a factor. This is so we can get
# estimates.

barley$Field <- as.factor(barley$Field)

# Fixed effects model with block
barleyModel <- aov(
  formula = Yield ~ Varietal + Field,
  data = barley
)

# QQ plot for residuals
car::qqPlot(
  x = residuals(barleyModel),
  distribution = "norm",
  envelope = 0.90,
  id = FALSE,
  pch = 20,
  ylab = "Residuals (BPA)"
)

# Strip chart, notice the calls to the random effects model
ggplot(
  data = data.frame(
    residuals = residuals(barleyModel),
    fitted = fitted.values(barleyModel)
  ),
  mapping = aes(x = fitted, y = residuals)
) +
  geom_point(size = 2) +
  geom_hline(
    yintercept = 0,
    linetype = "dashed",
    color = "grey50"
  ) +
  geom_smooth(
    formula = y ~ x,
    method = stats::loess,
    method.args = list(degree = 1),
    se = FALSE,
    size = 0.5
  ) +
  theme_bw() +
  xlab("Fitted values (BPA)") +
  ylab("Residuals (BPA)")

# Index Plot for Barley Yields
ggplot(
  data = barley,
  mapping = aes(
    x = Order,
    y = Yield,
    color = Field,
    shape = Varietal
  )
) +
  geom_point(size = 3) +
  geom_path(
    mapping = aes(group = Field)
  ) +
  ggplot2::theme_bw() +
  xlab("Planting/Havesting Order") +
  ylab("Yield (BPA)")
# Interaction Plot for Field and Treatment
ggplot2::ggplot(
  data = barley,
  mapping = aes(
    x = Varietal,
    y = Yield,
    color = Field,
    group = Field
  )
) +
  ggplot2::geom_point(size = 2) +
  ggplot2::geom_line(size = 1) +
  ggplot2::theme_bw() +
  xlab("Variety") +
  ylab("Yield (BPA)") +
  labs(color = "Field") +
  theme(
    legend.position = "right"
  )

# Omnibus Test/Modern ANOVA Table
parameters::model_parameters(
    model = barleyModel,
    omega_squared = "partial",
    eta_squared = "partial",
    epsilon_squared = "partial"
) %>%
  knitr::kable(
    digits = 4,
  col.names = c("Source", "SS", "df", "MS", "F", "p-value",
        "Partial Omega Sq.", "Partial Eta Sq.", "Partial Epsilon Sq."),
  caption = "ANOVA Table for Barley Crop Yield Study",
  align = c('l',rep('c',8)),
  booktab = TRUE
) %>%
  kableExtra::kable_styling(
    bootstrap_options = c("striped", "condensed"),
    font_size = 10,
    latex_options = c("scale_down", "HOLD_position")
  )

# Relative Efficiency of Block
block.RelEff(
  aov.obj = barleyModel,
  blockName = "Field",
  trtName = "Varietal"
)

# Point Estimates for Insect Sprays
pEst <- dummy.coef(barleyModel)
```

```r
pEst <- unlist(pEst)
names(pEst) <- c(
  "Grand Mean",
  levels(barley$Varietal),
  levels(barley$Field)
  )

data.frame("Estimate" = pEst) %>%
  knitr::kable(
  digits = 3,
  caption = "Point Estimates from the Barley Crop Yield Study",
  booktabs = TRUE,
  align = "c"
  ) %>%
  kableExtra::kable_styling(
    font_size = 12,
    latex_options = c("HOLD_position")
  )

# Post Hoc Analysis--Pairwise comparisons
barleyPH <- TukeyHSD(
  x = barleyModel,
  which = "Varietal", #We need to specify which model term we want
  conf.level = 0.9
)

knitr::kable(
  barleyPH$Varietal,
  digits = 4,
  caption = "Post Hoc Tukey HSD Comparisons",
  col.names = c("Difference", "Lower Bound",
                "Upper Bound", "Adj. p-Value"),
  align = 'cccc',
  booktabs = TRUE
) %>%
  kableExtra::kable_styling(
    bootstrap_options = c("condensed", "boardered"),
    font_size = 12,
    latex_options = "HOLD_position"
  )

anova.PostHoc(
  aov.obj = barleyModel, # Our aov output
  response = "Yield", # Our response variable
  mainEffect = "Varietal" # Our factor variable
) %>%
  knitr::kable(
    digits = 3,
    caption = "Post Hoc Comparison Effect Sizes",
    col.names = c("Pairwise Comparison","Cohen's d", "Hedge's g",
                  "Prob. Superiority"),
    align = 'lccc',
    booktabs = TRUE
  ) %>%
  kableExtra::kable_styling(
    bootstrap_options = c("condensed", "boardered"),
    font_size = 12,
    latex_options = "HOLD_position"
  )

# Define the contrast
company <- c(1/2, 1/2, -1/2, -1/2)

# Bind the contrast to our factor
contrasts(barley$Varietal) <- company

# Refit our model so that our contrast gets tested
barleyContrast <- aov(
  formula = Yield ~ Varietal + Field,
  data = barley
)

# Get the updated ANOVA Table
## Remember, you could also use the DescTools package for Scheffé here
conOut <- summary.aov(
  object = barleyContrast,
  split = list(
    Varietal = list(
      "Company A vs. Company B" = 1
    )
  )
)

# Make a nice table
knitr::kable(
  x = conOut[[1]],
  digits = 4,
  col.names = c(
    "DF", "SS", "MS", "F", "p-value"),
  caption = "ANOVA Table for Barley Crop Yield Contrasts",
  booktabs = TRUE,
  align = rep("c", 5)
) %>%
  kableExtra::kable_styling(
    font_size = 12,
    latex_options = c("HOLD_position")
  )
```