Data Analysis in R

Analysis of Variance (ANOVA)

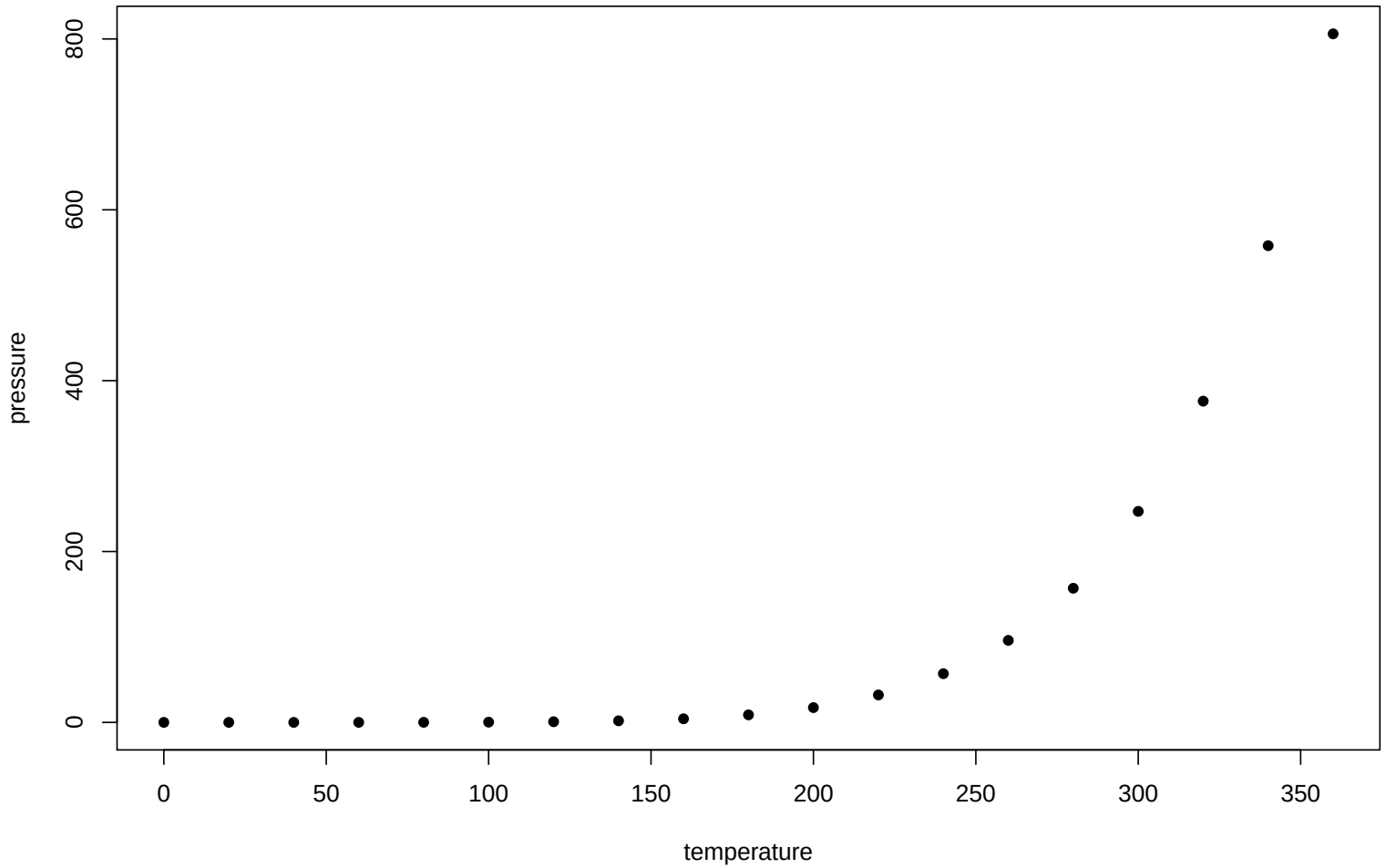
Maxwel Coura Oliveira, PhD

University of Wisconsin-Madison

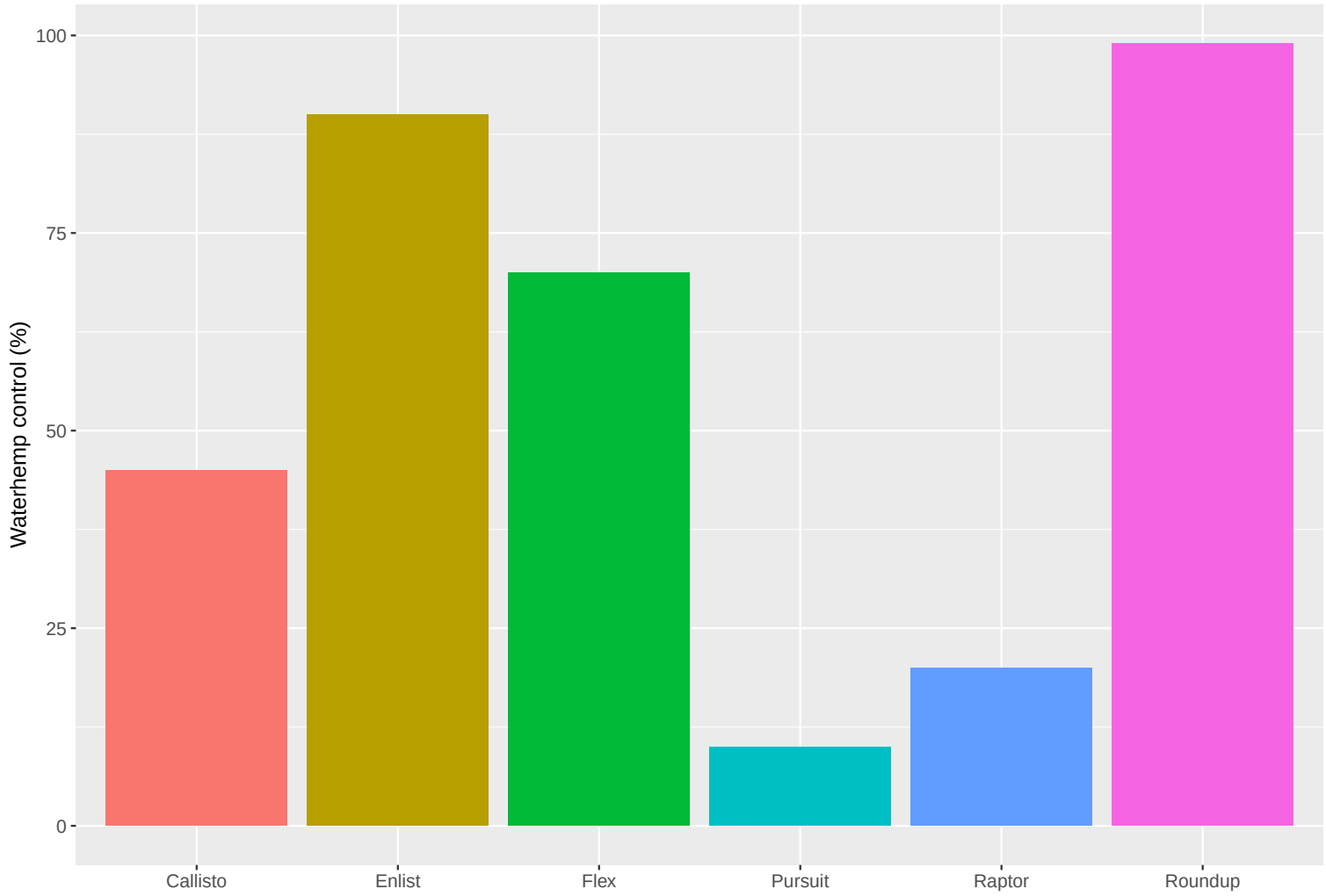
Check your data

- **Quantitative** data is information about quantities; that is, information that can be measured and written down with numbers. *Example*: x-axis = 1.5, 5.6, 12.8, 26.7. 39.4, 45.1
  - Proceed with regression
  - Numeric or integer
- **Qualitative** data is information about qualities; information that can't actually be measured. *Example*: x-axis = Pursuit, Roundup, Callisto
  - Proceed with ANOVA
  - Factor

Quantitative data



Qualitative data



The ANOVA Test

- A way to find if experiment results are significant. It helps you to figure out if you need to reject the null hypothesis or accept the alternate hypothesis
- Test groups to see if there's a difference between them
  - A group of weed scientists are trying different herbicides for Palmer amaranth control. You want to see if one herbicide is better than others
  - A company has 20 industrial hemp varieties from outside United States. They want to know which ones performed better in Wisconsin
  - Students from different colleges take the same exam. You want to see if one college outperforms the other

Assumptions of ANOVA

- Independence
- Normality
- Homogeneity of variances (aka, Homoscedasticity)

Parametric and Non-Parametric Tests

- **Parametric Tests**: Relies on theoretical distributions of the test statistic under the null hypothesis and assumptions about the distribution of the sample data (i.e., normality)
- **Non-Parametric Tests**: Referred to as "Distribution Free" as they do not assume that data are drawn from any particular distribution

Case of study

- Using the data **barley** from package *lattice*

```
library(lattice)
sample_n(barley, 5)
```

```
##      yield   variety year            site
## 1 41.33333    Velvet 1931       Crookston
## 2 25.23333  Peatland 1932       Crookston
## 3 36.56666     Trebi 1931 University Farm
## 4 26.16667   Glabron 1932       Crookston
## 5 48.86667 Manchuria 1931          Waseca
```

- A data frame with 120 observations on the following 4 variables
  - Yield (averaged across three blocks) in bushels/acre.
  - Variety (factor) with 10 levels “Svansota”, “No. 462”, “Manchuria”, “No. 475”, “Velvet”, “Peatland”, “Glabron”, “No. 457”, “Wisconsin No. 38”, “Trebi”.
  - Year (factor) with 2 levels 1931 and 1932
  - Site (factor) with 6 levels: “Grand Rapids”, “Duluth”, “University Farm”, “Morris”, “Crookston”, “Waseca”

Homogeneity of variances

- Bartlett test

```
bartlett.test(yield ~ variety, data = barley)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  yield by variety
## Bartlett's K-squared = 7.9816, df = 9, p-value = 0.536
```

- Levene test

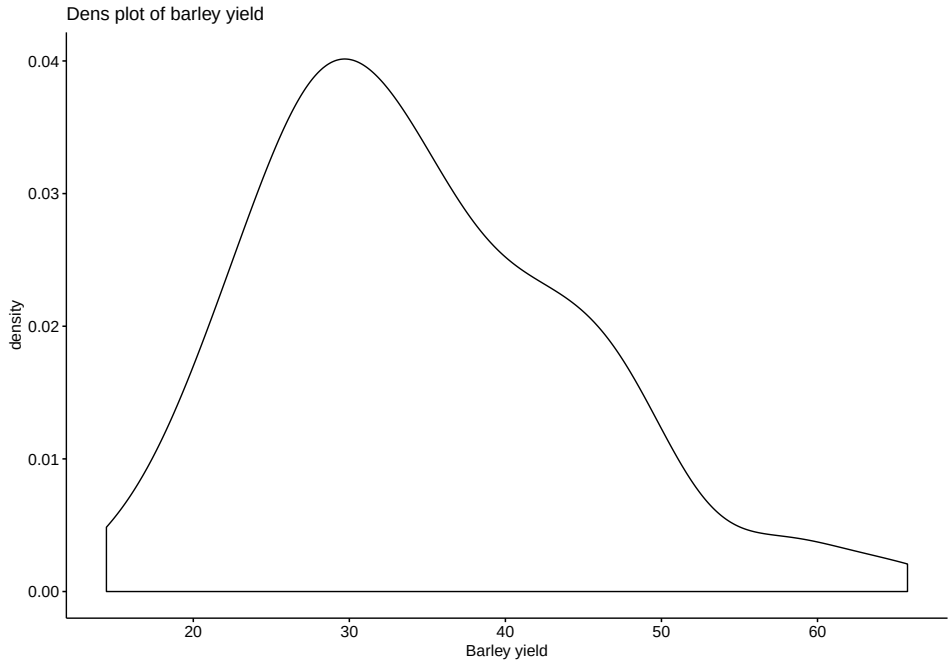
```
leveneTest(barley$yield, barley$variety)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##        Df F value Pr(>F)
## group   9   0.747 0.6652
##       110
```

Normality

- Density plot: the density plot provides a visual judgment about whether the distribution is bell shaped.

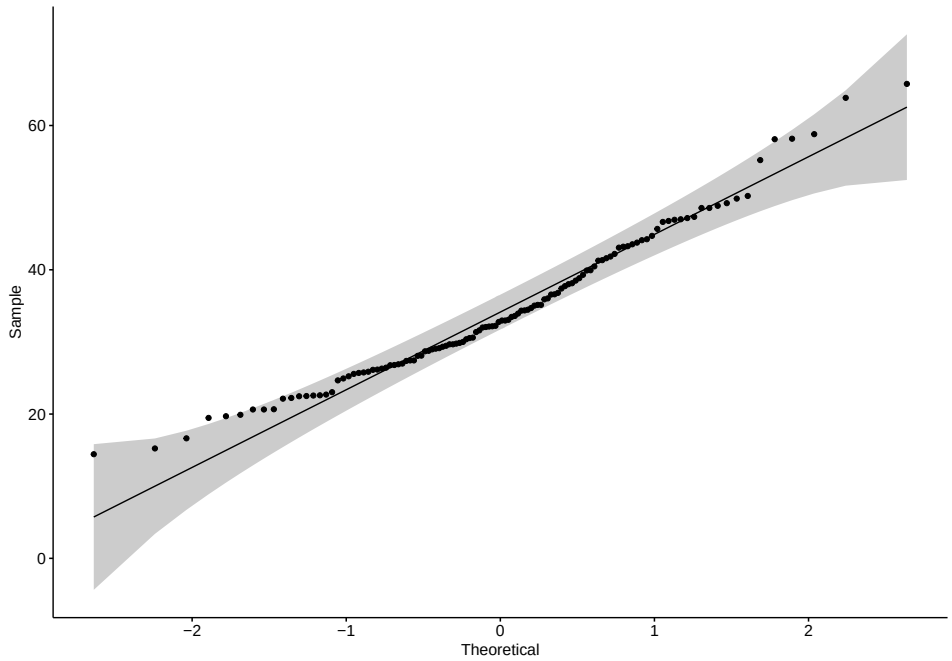
```
ggdensity(barley$yield,
          main = "Dens plot of barley yield",
          xlab = "Barley yield")
```



Q-Q plot

- Q-Q plot: Q-Q plot (or quantile-quantile plot) draws the correlation between a given sample and the normal distribution. A 45-degree reference line is also plotted.

```
ggqqplot(barley$yield)
```



Normality test

```
library(nortest)
pearson.test(barley$yield)
```

```
##
##  Pearson chi-square normality test
##
## data:  barley$yield
## P = 16.5, p-value = 0.1236
```

- From the output, the p-value > 0.05 implying that the distribution of the data are not significantly different from normal distribution. In other words, we can assume the normality.
- Other tests of normality (Shapiro-Wilk)

Data transformation

- Assuming data (barley yield) is non-gaussean (non-normal)
- Load the package *bestNormalize*

```
#install.packages("bestNormalize") # if needed
library(bestNormalize)
```

Package 'bestNormalize'

August 20, 2019

Type Package

Title Normalizing Transformation Functions

Version 1.4.2

Date 2019-08-20

Description Estimate a suite of normalizing transformations, including a new adaptation of a technique based on ranks which can guarantee normally distributed transformed data if there are no ties: ordered quantile normalization (ORQ). ORQ normalization combines a rank-mapping approach with a shifted logit approximation that allows the transformation to work on data outside the original domain. It is also able to handle new data within the original domain via linear interpolation. The package is built to estimate the best normalizing transformation for a vector consistently and accurately. It implements the Box-Cox transformation, the Yeo-Johnson transformation, three types of Lambert WxF transformations, and the ordered quantile normalization transformation. It also estimates the normalization efficacy of other commonly used transformations.

URL https://github.com/petersonR/bestNormalize

Transforming the data

-

```
data <- bestNormalize(barley$yield, loo = TRUE)
```

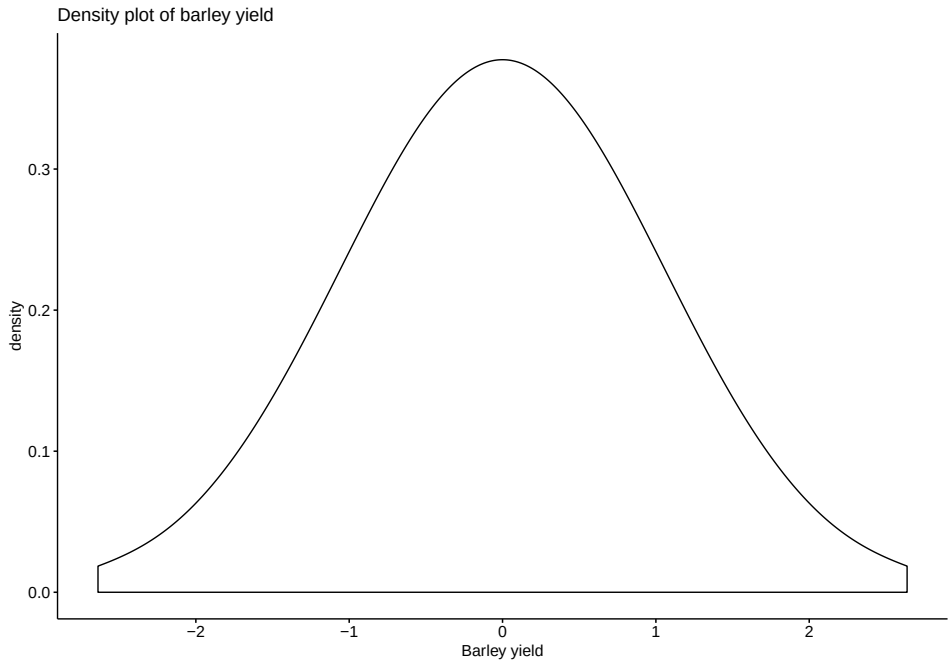
```
data
```

```
## Best Normalizing transformation with 120 Observations
##  Estimated Normality Statistics (Pearson P / df, lower => n
##  - No transform: 1.5
##  - Box-Cox: 0.4818
##  - Log_b(x+a): 0.4818
##  - sqrt(x+a): 0.9909
##  - exp(x): 136.9621
##  - arcsinh(x): 0.4818
##  - Yeo-Johnson: 0.4818
##  - orderNorm: 0.1212
## Estimation method: Out-of-sample via leave-one-out CV
##
```

Visual plots of transformed yield date

- Density plot: the density plot provides a visual judgment about whether the distribution is bell shaped (transformed data)

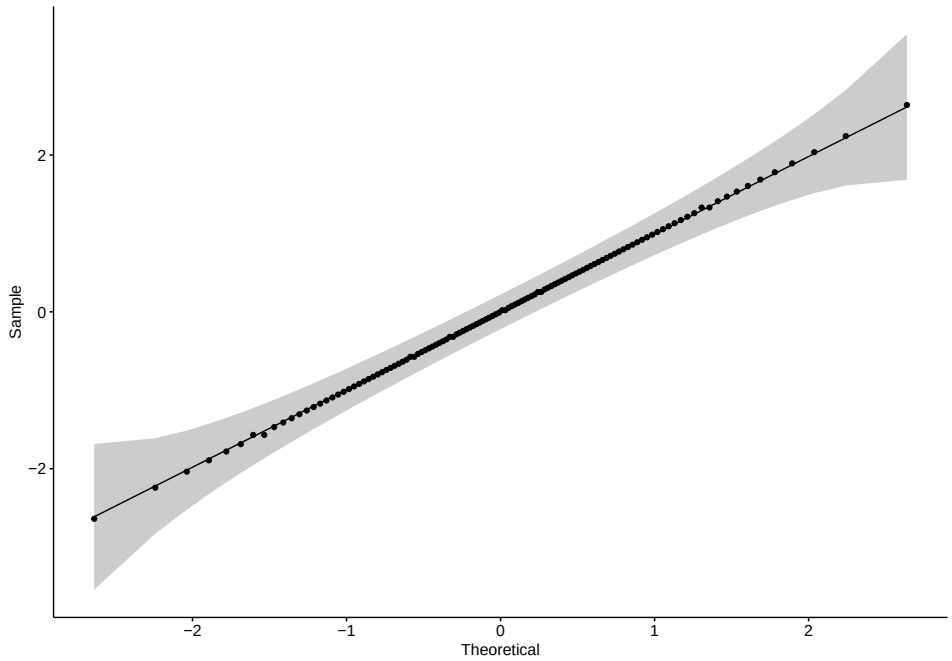
```
ggdensity(data$x.t,
          main = "Density plot of barley yield",
          xlab = "Barley yield")
```



Q-Q plot of transformed yield date

- Q-Q plot: Q-Q plot (or quantile-quantile plot) draws the correlation between a given sample and the normal distribution. A 45-degree reference line is also plotted

```
ggqqplot(data$x.t)
```



Normality test after data transformation

```
pearson.test(data$x.t)
```

```
##
##  Pearson chi-square normality test
##
## data:  data$x.t
## P = 0.4, p-value = 1
```

- From the output, the p-value > 0.05 implying that thedistribution of the data are not significantly different from normal distribution. In other words, we can assume the normality.

Post-Hoc ANOVA

- Use package: lme4

Package 'lme4'

March 5, 2019

Version 1.1-21

Title Linear Mixed-Effects Models using 'Eigen' and S4

Contact LME4 Authors <lme4-authors@lists.r-forge.r-project.org>

Description Fit linear and generalized linear mixed-effects models. The models and their components are represented using S4 classes and methods. The core computational algorithms are implemented using the 'Eigen' C++ library for numerical linear algebra and 'RcppEigen' ``glue''.

Depends R (>= 3.2.0), Matrix (>= 1.2-1), methods, stats

LinkingTo Rcpp (>= 0.10.5), RcppEigen

Imports graphics, grid, splines, utils, parallel, MASS, lattice, boot, nlme (>= 3.1-123), minqa (>= 1.1.15), nloptr (>= 1.0.4)

Suggests knitr, rmarkdown, PKPDmodels, MEMSS, testthat (>= 0.8.1), ggplot2, mlmRev, optimx (>= 2013.8.6), gamm4, pbkrtest, HSAUR2, numDeriv, car, dfoptim

VignetteBuilder knitr

LazyData yes

License GPL (>= 2)

URL https://github.com/lme4/lme4/

Post-Hoc ANOVA

Packages

```
library(lme4) # model
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Registered S3 methods overwritten by 'lme4':
##   method                          from
##   cooks.distance.influence.merMod car
##   influence.merMod                car
##   dfbeta.influence.merMod         car
##   dfbetas.influence.merMod        car
```

```
library(emmeans) # anova
library(lmerTest) # lsmeans
```

```
##
## Attaching package: 'lmerTest'
```

Model

- Mixed model:

  Fixed: Variety

  Random: Year

```
fit <- lmer(yield ~ variety * site + (1|year), data=barley)
```

Summary

```
summary(fit)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: yield ~ variety * site + (1 | year)
##    Data: barley
##
## REML criterion at convergence: 449.8
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -1.877 -0.491  0.000  0.491  1.877
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  year     (Intercept) 13.29    3.645
##  Residual             50.35    7.096
## Number of obs: 120, groups:  year, 2
##
## Fixed effects:
##                                    Estimate Std. Error      df
## (Intercept)                         23.1500     5.6406 16.7881
## varietyNo. 462                      -0.7333     7.0955 59.0002
## varietyManchuria                    -4.4000     7.0955 59.0002
```

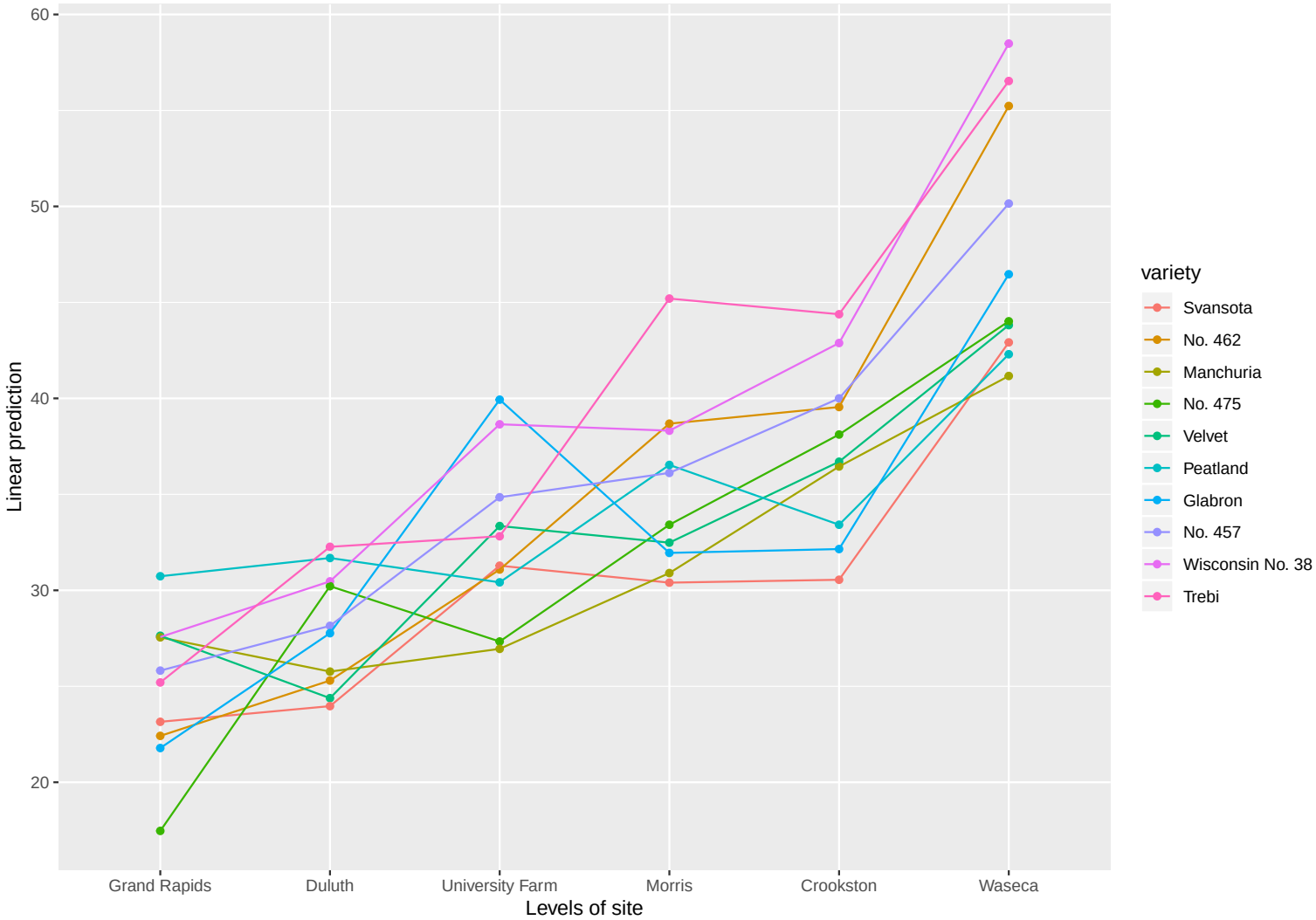
ANOVA

```
anova(fit)
```

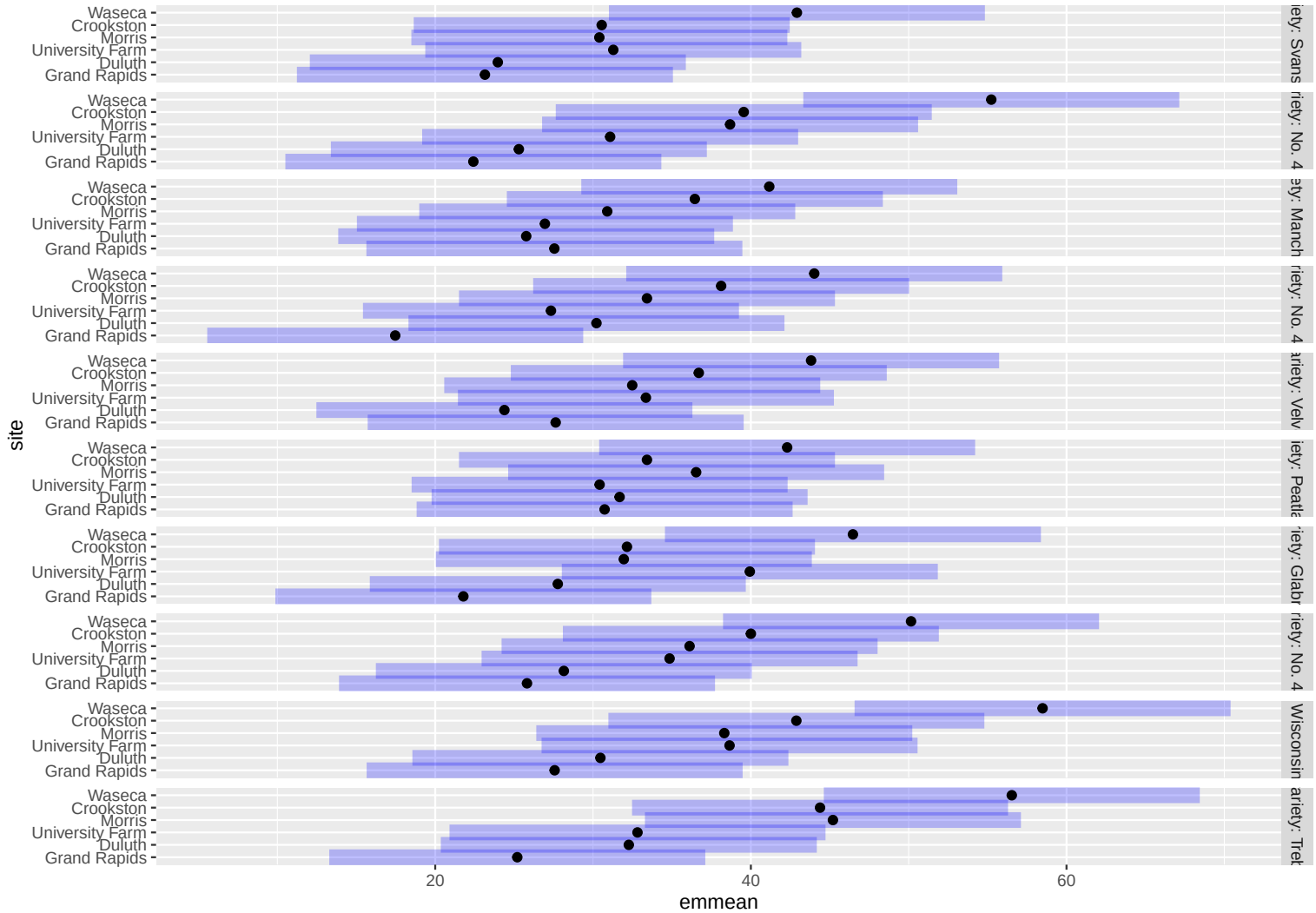
```
## Type III Analysis of Variance Table with Satterthwaite's method
##              Sum Sq Mean Sq NumDF DenDF F value   Pr(>F)
## variety      1052.6  116.95     9    59  2.3230  0.02592 *
## site         6633.9 1326.77     5    59 26.3529 6.95e-14 ***
## variety:site 1205.8   26.79    45    59  0.5322  0.98544
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Visual interaction

```
emmip(fit, variety ~ site | site) # include CIs = TRUE
```



```
plot(emmeans(fit, ~ site | site*variety))
```



LS Means

- Looking the **site** level (Factor)

```
lssite<-emmeans(fit, ~site, contr="pairwise", adjust="none", type="response")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

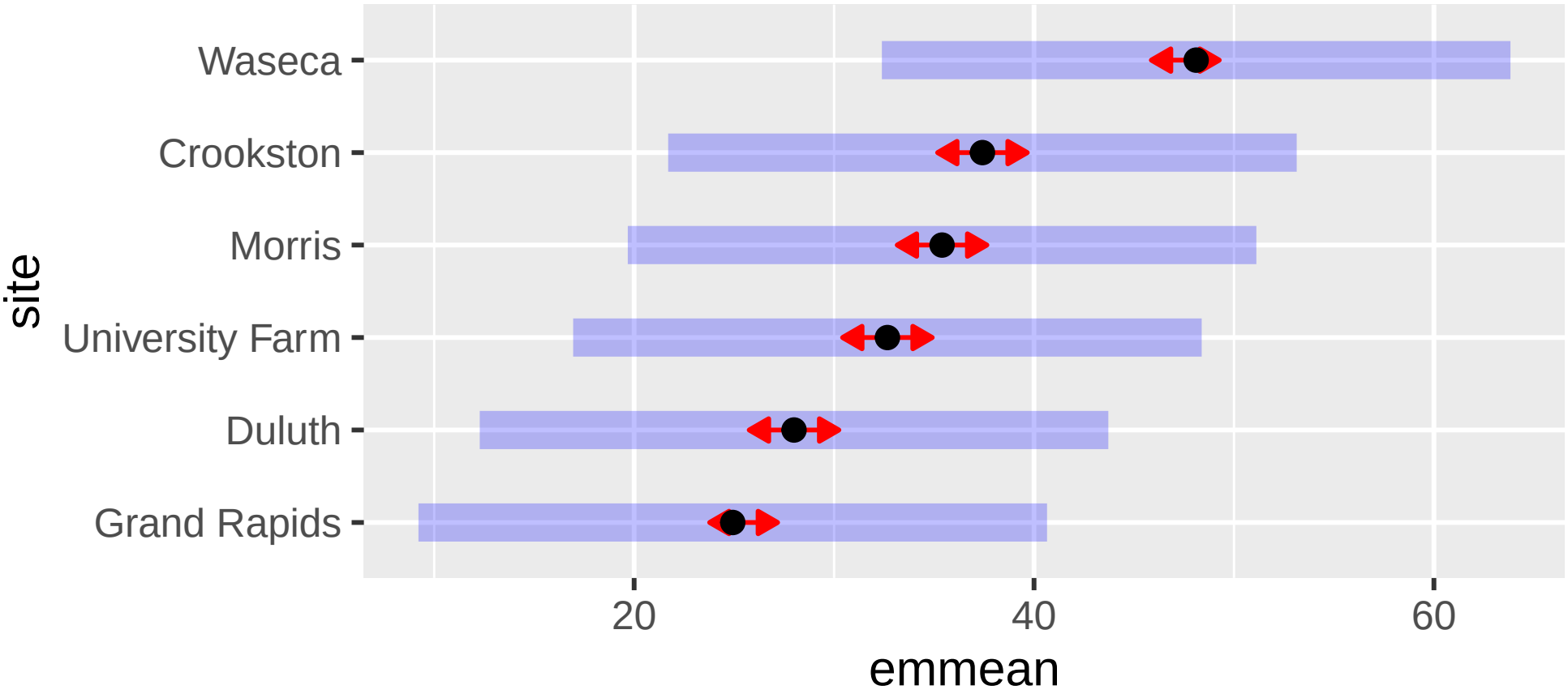
```
lssite$emmeans
```

```
##  site            emmean   SE   df lower.CL upper.CL
##  Grand Rapids      24.9 3.03 1.68     9.21     40.6
##  Duluth            28.0 3.03 1.68    12.28     43.7
##  University Farm   32.7 3.03 1.68    16.95     48.4
##  Morris            35.4 3.03 1.68    19.68     51.1
##  Crookston         37.4 3.03 1.68    21.70     53.1
##  Waseca            48.1 3.03 1.68    32.39     63.8
##
## Results are averaged over the levels of: variety
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
```

Plot

- Plotting the site *level* (Factor)

```
plot(lssite, comparisons =TRUE, adjust="none")
```



LS Means

- Looking the **variety** level (Factor)

```
lsvar<-lsmeans(fit, ~variety, contr="pairwise", adjust="none", type="response")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

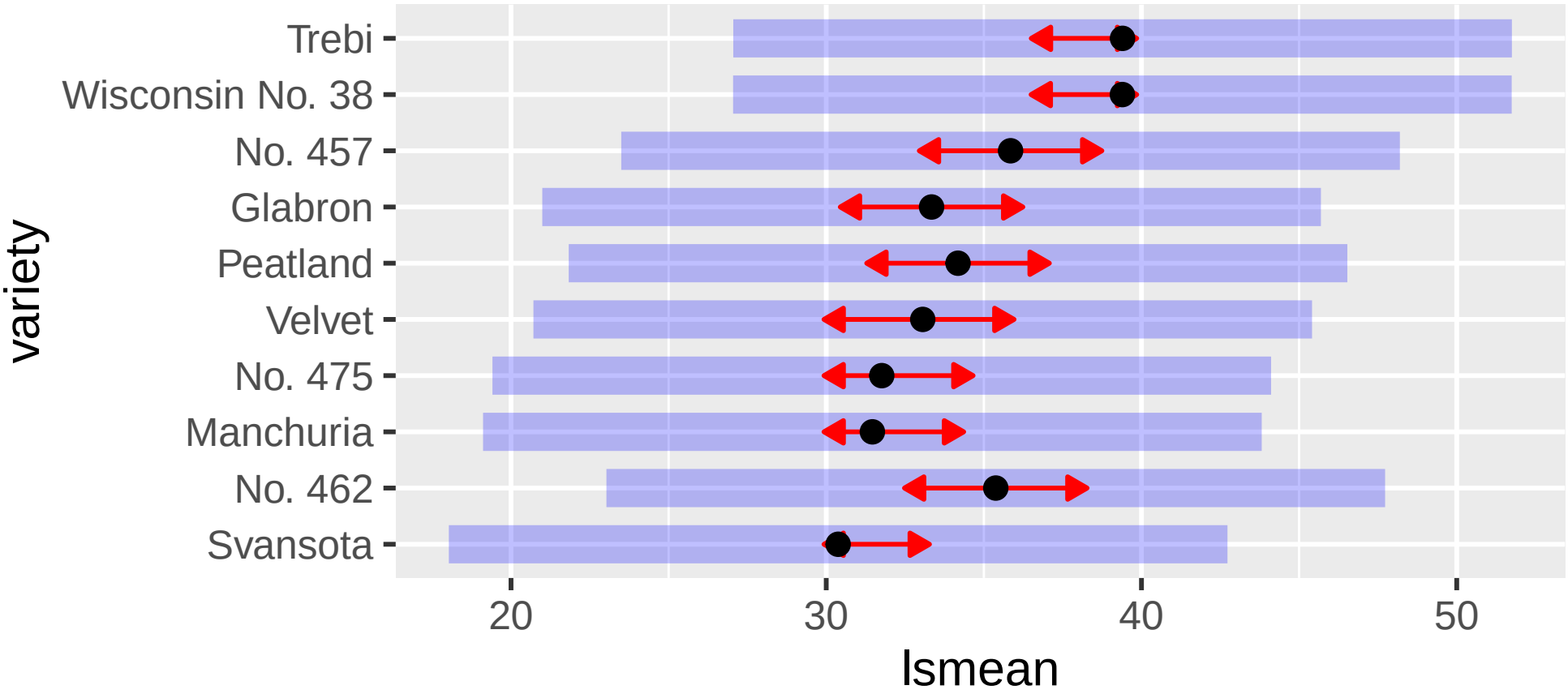
```
lsvar$lsmeans
```

```
##  variety          lsmean   SE   df lower.CL upper.CL
##  Svansota           30.4 3.29 2.34     18.0     42.7
##  No. 462            35.4 3.29 2.34     23.0     47.7
##  Manchuria          31.5 3.29 2.34     19.1     43.8
##  No. 475            31.8 3.29 2.34     19.4     44.1
##  Velvet             33.1 3.29 2.34     20.7     45.4
##  Peatland           34.2 3.29 2.34     21.8     46.5
##  Glabron            33.3 3.29 2.34     21.0     45.7
##  No. 457            35.8 3.29 2.34     23.5     48.2
##  Wisconsin No. 38   39.4 3.29 2.34     27.0     51.7
##  Trebi              39.4 3.29 2.34     27.0     51.8
##
## Results are averaged over the levels of: site
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
```

LS Means

- Plotting the **variety** level (factor)

```
plot(lsvar, comparisons =TRUE, adjust="none")
```



Non-Parametric Tests

- Case of study: Percentage data (0 to 100%)
  - Weed Control
  - Disease
  - Insect damage

Non-Parametric Tests

- Cover crop management (Kolby's study)

```
sample_n(Data, size=5)
```

```
## # A tibble: 5 x 5
##   trt                                    trtn    rep crop    control
##   <chr>                                  <fct> <dbl> <chr>     <dbl>
## 1 No-till + POST                         3         4 Soybean    0.9
## 2 Cereal Rye terminated at plant + POST  4         3 Soybean    0.99
## 3 No-till + POST                         3         2 Soybean    0.7
## 4 Cerealy Rye forage harvest + PRE fb POST 11      3 Soybean    0.99
## 5 Cereal Rye terminated at plant + POST  4         1 Soybean    0.9
```

- Values must be between 0 and 1

Generalized Linear Mixed Models using Template Model Builder

- Fit linear and generalized linear mixed models with various extensions, including zero-inflation. The models are fitted using maximum likelihood estimation via 'TMB' (Template Model Builder)
- Random effects are assumed to be Gaussian on the scale of the linear predictor and are integrated out using the Laplace approximation. Gradients are calculated using automatic differentiation

Homogeneity of variances

- Bartlett test

```
bartlett.test(control ~ trtn, data=Data)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  control by trtn
## Bartlett's K-squared = Inf, df = 10, p-value < 2.2e-16
```

- levene test

```
leveneTest(Data$control, Data$trtn)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value  Pr(>F)
## group 10  1.8176 0.09158 .
##       37
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model

```
library(glmmTMB)
model <- glmmTMB(control ~ trtn + (1|rep), data=Data,  beta_family(link = "logit"))
```

Summary

```
Sum <- summary(model)
Sum$coefficients
```

```
## $cond
##                Estimate Std. Error   z value     Pr(>|z|)
## (Intercept) -0.8771015  0.3208830 -2.733399 6.268431e-03
## trtn3        2.5672054  0.4223463  6.078437 1.213594e-09
## trtn4        3.4124799  0.5599591  6.094159 1.100145e-09
## trtn5        4.0050302  0.5967980  6.710864 1.934760e-11
## trtn6        4.0050179  0.5967956  6.710870 1.934668e-11
## trtn7        3.3517779  0.5556180  6.032522 1.614203e-09
## trtn8        1.9697344  0.4639821  4.245281 2.183199e-05
## trtn9        3.6705235  0.5774255  6.356704 2.061281e-10
## trtn10       3.4741377  0.5642906  6.156646 7.430164e-10
## trtn11       3.8424874  0.5879081  6.535864 6.324327e-11
## trtn12       4.0050179  0.5967956  6.710870 1.934668e-11
##
## $zi
## NULL
##
## $disp
## NULL
```

ANOVA

- We should use Anova.glmmTBM

```
Anova(model, test.statistic = "Chisq", type = "II")
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: control
##       Chisq Df Pr(>Chisq)
## trtn 97.126 10  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

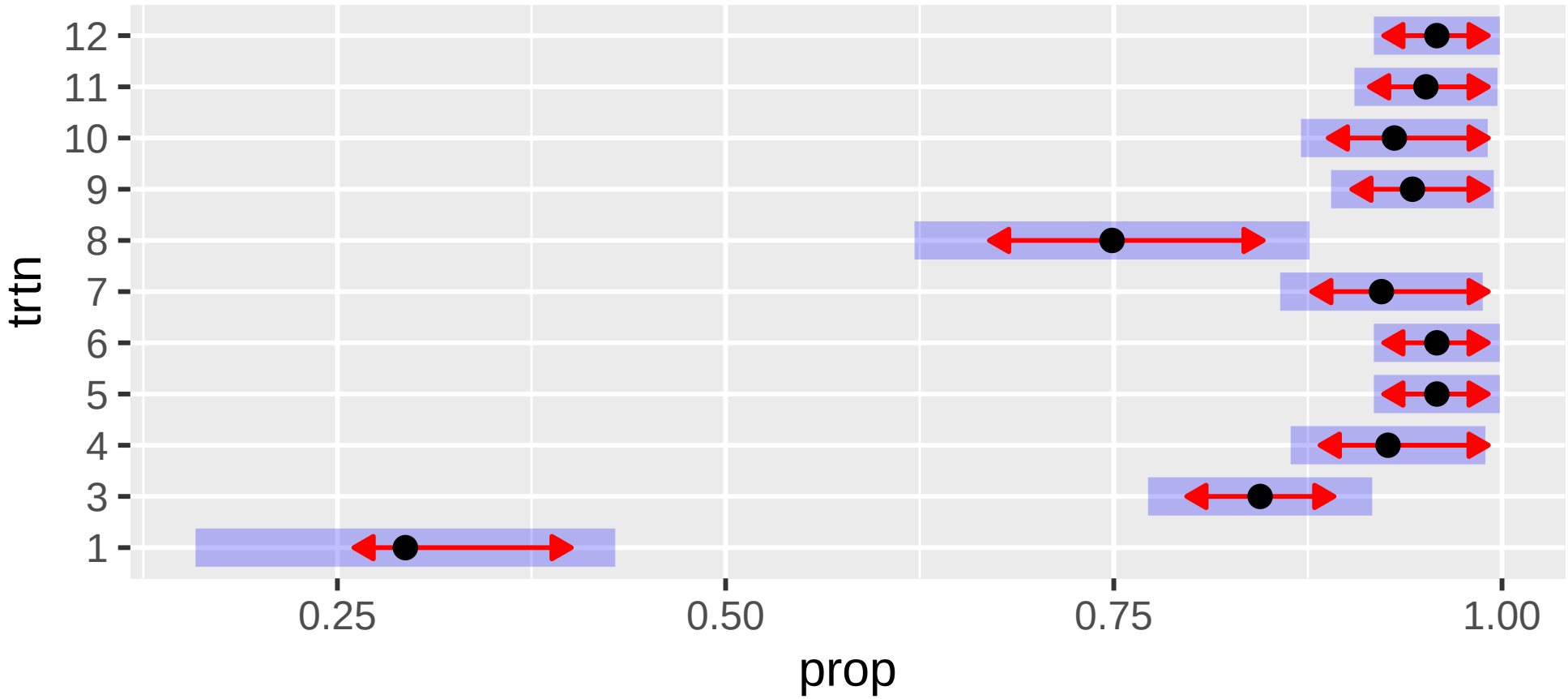
LS Means

```
lsm=emmeans(model , ~ trtn, contr="pairwise", adjust="none", type = "response")
lsm
```

```
## $emmeans
##  trtn  prop     SE df lower.CL upper.CL
##  1    0.294 0.0666 35    0.178    0.444
##  3    0.844 0.0355 35    0.758    0.904
##  4    0.927 0.0309 35    0.834    0.969
##  5    0.958 0.0200 35    0.893    0.984
##  6    0.958 0.0200 35    0.893    0.984
##  7    0.922 0.0321 35    0.827    0.967
##  8    0.749 0.0627 35    0.603    0.854
##  9    0.942 0.0258 35    0.862    0.977
##  10   0.931 0.0296 35    0.841    0.971
##  11   0.951 0.0227 35    0.878    0.981
##  12   0.958 0.0200 35    0.893    0.984
##
## Confidence level used: 0.95
## Intervals are back-transformed from the logit scale
##
## $contrasts
##  contrast odds.ratio     SE df t.ratio p.value
##  1 / 3        0.0767 0.0324 35 -6.078  <.0001
##  1 / 4        0.0330 0.0185 35 -6.094  <.0001
```

Plot

```
plot(lsm, comparisons =TRUE, adjust="none")
```



```
CLD(lsm, alpha=0.05, Letters=letters, adjust="none", reversed = TRUE)
```

```
## Warning: 'CLD' will be deprecated. Its use is discouraged.
## See '? CLD' for an explanation. Use 'pwpp' or 'multcomp::cld' instead.

## Warning in CLD.emm_list(lsm, alpha = 0.05, Letters = letters, adjust =
## "none", : `CLD()` called with a list of 2 objects. Only the first one was
## used.

## Warning: 'CLD' will be deprecated. Its use is discouraged.
## See '? CLD' for an explanation. Use 'pwpp' or 'multcomp::cld' instead.

##  trtn  prop     SE df lower.CL upper.CL .group
##  5    0.958 0.0200 35    0.893    0.984  a
##  12   0.958 0.0200 35    0.893    0.984  a
##  6    0.958 0.0200 35    0.893    0.984  a
##  11   0.951 0.0227 35    0.878    0.981  a
##  9    0.942 0.0258 35    0.862    0.977  a
##  10   0.931 0.0296 35    0.841    0.971  ab
##  4    0.927 0.0309 35    0.834    0.969  ab
##  7    0.922 0.0321 35    0.827    0.967  ab
##  3    0.844 0.0355 35    0.758    0.904   bc
##  8    0.749 0.0627 35    0.603    0.854    c
##  1    0.294 0.0666 35    0.178    0.444     d
##
## Confidence level used: 0.95
## Intervals are back-transformed from the logit scale
## Tests are performed on the log odds ratio scale
```

Plotting