# Untitled3

Telha H. Rehman

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

## START

```
Sys.time()
```

```
## [1] "2022-02-21 16:02:29 PST"
```

```
library(tinytex)
library(ggplot2)
library(tidyverse)
library(cowplot)
library(Cairo)
library(modelr)
library(gridExtra)
library(mixtools)
library(nlme)
library(car)
library(emmeans)
library(MuMIn)
library(ggpmisc)
library(gridExtra)
library(gtable)
library(grid)
library(RColorBrewer)
library(segmented)
library(data.table)
library(scales)
```

#DATA

##GreenSeeker NDVI Data

```
gs_ndvi_data <- read_csv(file = "DATA/PI_greenseeker_data.csv")
```

```
## Rows: 328 Columns: 17
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): site_year, NDVI_1, NDVI_2, NDVI_3, NDVI_4
```

```
## dbl (12): year, exp_plot_number, block, plot, N_level, N_level_kgha, biomass...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
str(gs_ndvi_data , give.attr = FALSE)
```

```
## spec_tbl_df[,17] [328 x 17] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ site_year         : chr [1:328] "Arbuckle-15" "Arbuckle-15" "Arbuckle-15" "Arbuckle-15" ...
##  $ year              : num [1:328] 2015 2015 2015 2015 2015 ...
##  $ exp_plot_number   : num [1:328] 101 102 103 104 105 201 202 203 204 205 ...
##  $ block             : num [1:328] 1 1 1 1 1 2 2 2 2 2 ...
##  $ plot              : num [1:328] 1 2 3 4 5 1 2 3 4 5 ...
##  $ N_level           : num [1:328] 125 225 0 75 175 0 175 125 225 75 ...
##  $ N_level_kgha      : num [1:328] 125 225 0 75 175 0 175 125 225 75 ...
##  $ biomass_plus_bag_g: num [1:328] 414 472 281 386 455 304 402 322 418 336 ...
##  $ paper_bag_g       : num [1:328] 45 45 45 45 45 45 45 45 45 45 ...
##  $ ring_size         : num [1:328] 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
##  $ num_of_paper_bags : num [1:328] 1 1 1 1 1 1 1 1 1 1 ...
##  $ sample_weight_mg  : num [1:328] 4.84 5.12 4.78 5.15 4.93 ...
##  $ sample_N_ug       : num [1:328] 117.1 153.4 64.9 92.9 116 ...
##  $ NDVI_1            : chr [1:328] "0.77" "0.82" "0.56" "0.72" ...
##  $ NDVI_2            : chr [1:328] "n/a" "n/a" "n/a" "n/a" ...
##  $ NDVI_3            : chr [1:328] "n/a" "n/a" "n/a" "n/a" ...
##  $ NDVI_4            : chr [1:328] "n/a" "n/a" "n/a" "n/a" ...
```

```
gs_ndvi_data <- gs_ndvi_data %>%
  filter(!year %in% c("2015" , "2016"),
         N_level_kgha != 275) #remove the years we don't need for this analysis and also the N rate tha

str(gs_ndvi_data , give.attr = FALSE)
```

```
## spec_tbl_df[,17] [240 x 17] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ site_year         : chr [1:240] "Nicolaus-17" "Nicolaus-17" "Nicolaus-17" "Nicolaus-17" ...
##  $ year              : num [1:240] 2017 2017 2017 2017 2017 ...
##  $ exp_plot_number   : num [1:240] 101 102 103 104 106 107 202 203 204 205 ...
##  $ block             : num [1:240] 1 1 1 1 1 1 2 2 2 2 ...
##  $ plot              : num [1:240] 1 2 3 4 6 7 2 3 4 5 ...
##  $ N_level           : num [1:240] 225 0 125 175 45 75 75 125 0 175 ...
##  $ N_level_kgha      : num [1:240] 225 0 125 175 45 75 75 125 0 175 ...
##  $ biomass_plus_bag_g: num [1:240] 361 264 318 360 285 ...
##  $ paper_bag_g       : num [1:240] 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 ...
##  $ ring_size         : num [1:240] 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
##  $ num_of_paper_bags : num [1:240] 1 1 1 1 1 1 1 1 1 1 ...
##  $ sample_weight_mg  : num [1:240] 3.17 3.42 3.09 3.07 3.2 ...
##  $ sample_N_ug       : num [1:240] 79.4 61.1 80.7 95.9 63.6 ...
##  $ NDVI_1            : chr [1:240] "0.77" "0.50" "0.67" "0.79" ...
##  $ NDVI_2            : chr [1:240] "0.77" "0.52" "0.67" "0.78" ...
##  $ NDVI_3            : chr [1:240] "0.79" "0.46" "0.71" "0.79" ...
##  $ NDVI_4            : chr [1:240] "n/a" "n/a" "n/a" "n/a" ...
```

```
gs_ndvi_data <- gs_ndvi_data[c(1:240), c(1:17)] #removes the empty rows and columns from the data frame

gs_ndvi_data$block <- factor(gs_ndvi_data$block)
gs_ndvi_data$year <- factor(gs_ndvi_data$year)
```

```
gs_ndvi_data$plot <- factor(gs_ndvi_data$plot)
gs_ndvi_data$N_level_kgha_f <- factor(gs_ndvi_data$N_level_kgha)
gs_ndvi_data$exp_plot_number <- factor(gs_ndvi_data$exp_plot_number)
gs_ndvi_data$site_year <- factor(gs_ndvi_data$site_year , levels = c( "Nicolaus-17" , "Williams-17" , "
gs_ndvi_data$NDVI_1 <- as.numeric(as.character(gs_ndvi_data$NDVI_1))
```

```
## Warning: NAs introduced by coercion
```

```
gs_ndvi_data$NDVI_2 <- as.numeric(as.character(gs_ndvi_data$NDVI_2))
```

```
## Warning: NAs introduced by coercion
```

```
gs_ndvi_data$NDVI_3 <- as.numeric(as.character(gs_ndvi_data$NDVI_3))
gs_ndvi_data$NDVI_4 <- as.numeric(as.character(gs_ndvi_data$NDVI_4)) #gets the data right
```

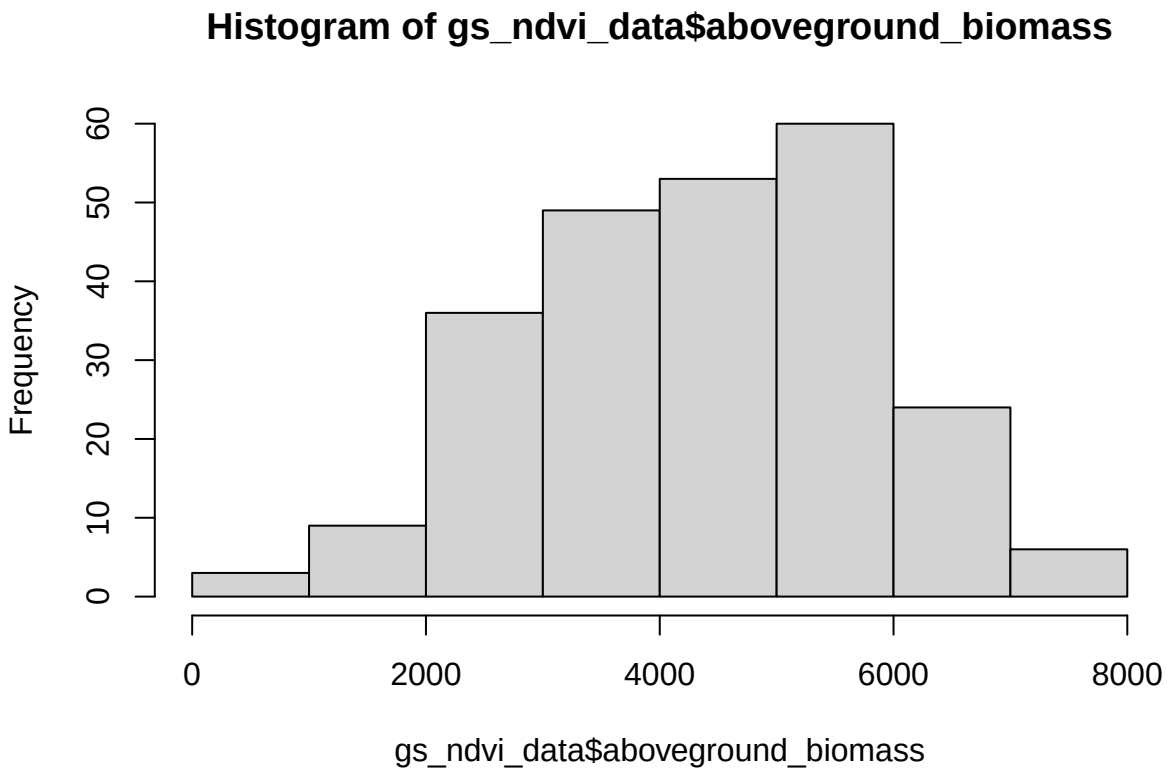
```
## Warning: NAs introduced by coercion
```

```
str(gs_ndvi_data , give.attr = FALSE)
```

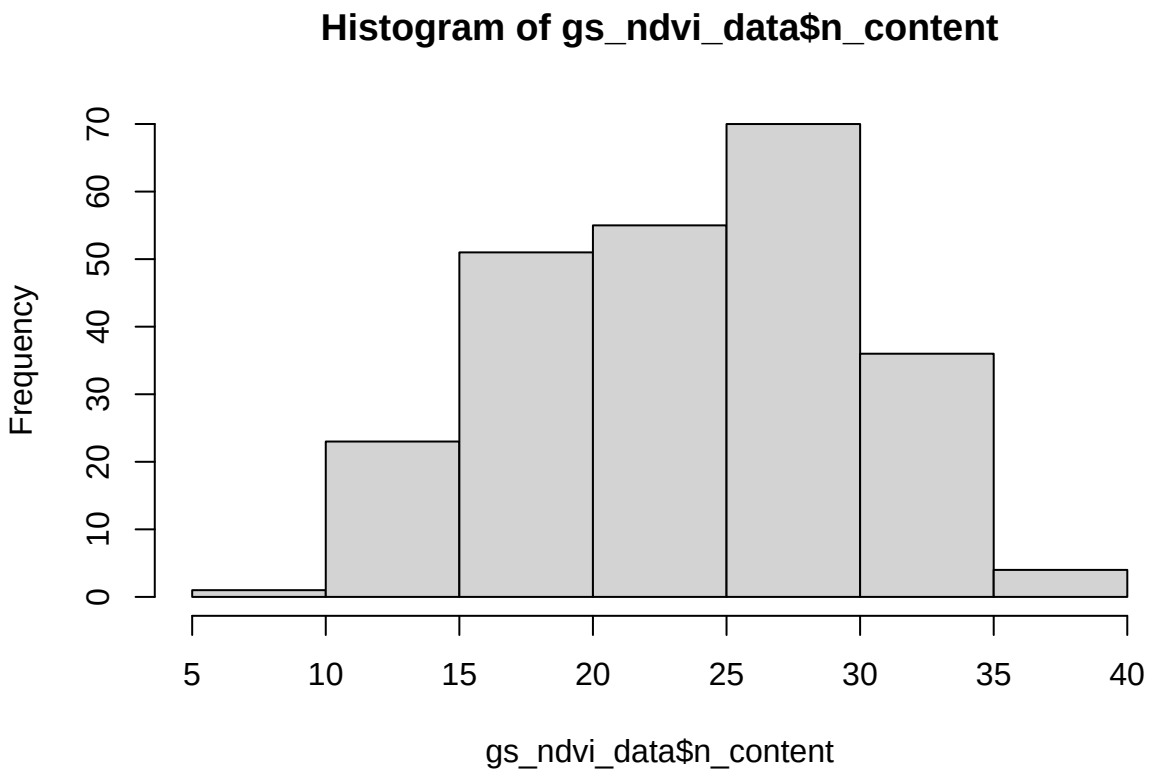
```
## tibble[,18] [240 x 18] (S3: tbl_df/tbl/data.frame)
##  $ site_year         : Factor w/ 10 levels "Nicolaus-17",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year              : Factor w/ 3 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ exp_plot_number   : Factor w/ 27 levels "101","102","103",..: 1 2 3 4 6 7 9 10 11 12 ...
##  $ block             : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 2 2 2 2 ...
##  $ plot              : Factor w/ 7 levels "1","2","3","4",..: 1 2 3 4 6 7 2 3 4 5 ...
##  $ N_level           : num [1:240] 225 0 125 175 45 75 75 125 0 175 ...
##  $ N_level_kgha      : num [1:240] 225 0 125 175 45 75 75 125 0 175 ...
##  $ biomass_plus_bag_g: num [1:240] 361 264 318 360 285 ...
##  $ paper_bag_g       : num [1:240] 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 ...
##  $ ring_size         : num [1:240] 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
##  $ num_of_paper_bags : num [1:240] 1 1 1 1 1 1 1 1 1 1 ...
##  $ sample_weight_mg  : num [1:240] 3.17 3.42 3.09 3.07 3.2 ...
##  $ sample_N_ug       : num [1:240] 79.4 61.1 80.7 95.9 63.6 ...
##  $ NDVI_1            : num [1:240] 0.77 0.5 0.67 0.79 0.68 0.63 0.66 0.68 0.54 0.77 ...
##  $ NDVI_2            : num [1:240] 0.77 0.52 0.67 0.78 0.69 0.56 0.65 0.68 0.5 0.75 ...
##  $ NDVI_3            : num [1:240] 0.79 0.46 0.71 0.79 0.68 0.53 0.63 0.64 0.5 0.77 ...
##  $ NDVI_4            : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ N_level_kgha_f    : Factor w/ 11 levels "0","45","75",..: 10 1 5 8 2 3 3 5 1 8 ...
```

```
gs_ndvi_data <-  gs_ndvi_data %>%
  mutate( biomass_dry_wt = biomass_plus_bag_g - (paper_bag_g * num_of_paper_bags) ,
          aboveground_biomass = (biomass_dry_wt / ring_size) * 10 ,
          n_content = sample_N_ug / sample_weight_mg ,
          PI_N_Uptake = (aboveground_biomass * n_content) / 1000 #n uptake in kg per ha
          )#processes the data


gs_ndvi_data <- gs_ndvi_data %>%
  rowwise() %>%
  mutate(NDVI = mean(c( NDVI_1 , NDVI_2 , NDVI_3 , NDVI_4) , na.rm = T)) #takes average of four NDVI re

gs_ndvi_data <- dplyr::select(gs_ndvi_data ,
                      site_year,
                      year,
                      exp_plot_number,
                      block,
```

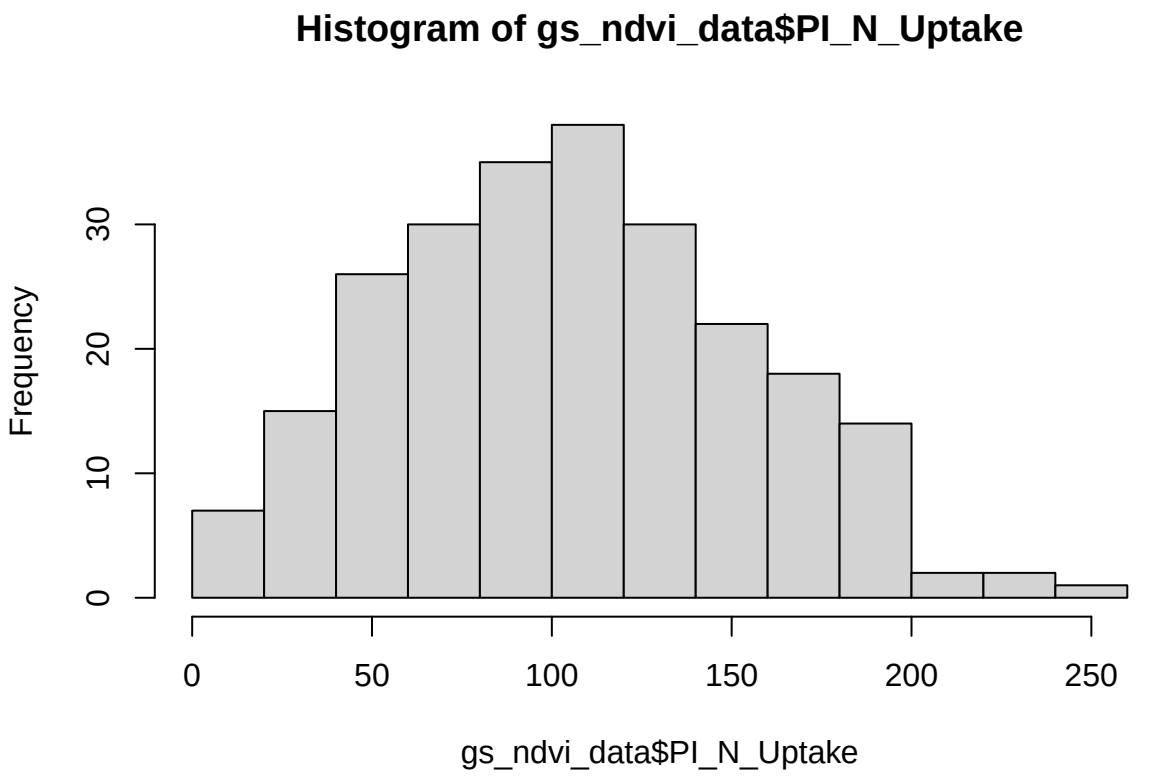
```
                      plot,
                      N_level_kgha,
                      aboveground_biomass,
                      n_content,
                      PI_N_Uptake,
                      NDVI) #selects the relevant columns

gs_ndvi_data$site_year <- factor(gs_ndvi_data$site_year , levels = c( "Nicolaus-17" , "Williams-17" , "

gs_ndvi_data$NDVI <- round(gs_ndvi_data$NDVI , digits = 2) #rounds to the precision of the instrument

hist(gs_ndvi_data$aboveground_biomass)
```




```
hist(gs_ndvi_data$n_content)
```




```
hist(gs_ndvi_data$PI_N_Uptake)
```



## Yield Data

```
yield_data <- read_csv(file = "DATA/yield_data.csv")
```

```
## Rows: 672 Columns: 34
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (2): site_year, Received_TopDress
## dbl (32): year, Block, MainPlot, SubPlot, exp_plot_number, N_level, TopDress...
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
str(yield_data , give.attr = FALSE)
```

```
## spec_tbl_df[,34] [672 x 34] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ site_year            : chr [1:672] "Davis-16" "Davis-16" "Davis-16" "Davis-16" ...
##  $ year                 : num [1:672] 2016 2016 2016 2016 2016 ...
##  $ Block                : num [1:672] 1 1 1 1 1 2 2 2 2 2 ...
##  $ MainPlot             : num [1:672] 1 2 3 4 5 1 2 3 4 5 ...
##  $ SubPlot              : num [1:672] 3 1 3 2 1 2 3 2 3 2 ...
##  $ exp_plot_number      : num [1:672] 101 102 103 104 105 201 202 203 204 205 ...
##  $ N_level              : num [1:672] 75 225 0 125 175 75 0 225 175 125 ...
##  $ TopDress             : num [1:672] 0 0 0 0 0 0 0 0 0 0 ...
##  $ SeasonalNRate        : num [1:672] 75 225 0 125 175 75 0 225 175 125 ...
##  $ N_level_kgha         : num [1:672] 75 225 0 125 175 75 0 225 175 125 ...
##  $ TopDress_kgha        : num [1:672] 0 0 0 0 0 0 0 0 0 0 ...
##  $ SeasonalNRate_kgha   : num [1:672] 75 225 0 125 175 75 0 225 175 125 ...
##  $ Received_TopDress    : chr [1:672] "NO" "NO" "NO" "NO" ...
##  $ tare                 : num [1:672] 1220 1220 1220 1220 1220 1220 1220 1220 1220 1220 ...
##  $ FW1PlusTare          : num [1:672] 5662 5298 5256 4846 5220 ...
##  $ FW2PlusTare          : num [1:672] 4610 5438 1220 4278 5156 ...
##  $ SSFWPlusTare         : num [1:672] 3070 3262 2266 2482 2650 ...
##  $ SSODW                : num [1:672] 692 705 519 541 538 ...
##  $ HarvestBagPlusTie    : num [1:672] 121 122 122 121 122 ...
##  $ Grain1PlusPaperBag1  : num [1:672] 292 271 188 215 208 ...
##  $ PaperBag1            : num [1:672] 7.92 7.92 7.92 7.92 7.92 7.92 7.92 7.92 7.92 7.92 ...
##  $ Grain2PlusPaperBag2  : num [1:672] 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 ...
##  $ PaperBag2            : num [1:672] 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 ...
##  $ StrawSampleSize      : num [1:672] 8.27 7.74 7.9 8.38 7.52 ...
##  $ StrawN               : num [1:672] 55.1 55.2 49.4 47.1 45.3 ...
##  $ GrainSampleSize      : num [1:672] 4.95 4.97 5.04 5.02 4.57 ...
##  $ GrainN               : num [1:672] 54.5 58.6 47.4 52.5 46.1 ...
##  $ DaysPI2Harvest       : num [1:672] 102 102 102 102 102 102 102 102 102 102 ...
##  $ SeedTray1.1          : num [1:672] 243 243 243 243 243 243 243 243 243 243 ...
##  $ SeedTray1.2          : num [1:672] 0.62 0.62 0.62 0.64 0.64 0.62 0.62 0.62 0.62 0.62 ...
##  $ Grain3PlusSeedTray1  : num [1:672] 435 514 426 455 447 ...
##  $ MoistureContentGrain3: num [1:672] 11.2 11.2 11.2 11.2 11.2 11.2 11.2 11.2 11.2 11.2 ...
##  $ SeedTray2            : num [1:672] 244 244 244 244 244 ...
##  $ Grain4PlusSeedTray2  : num [1:672] 254 254 254 254 254 ...
```

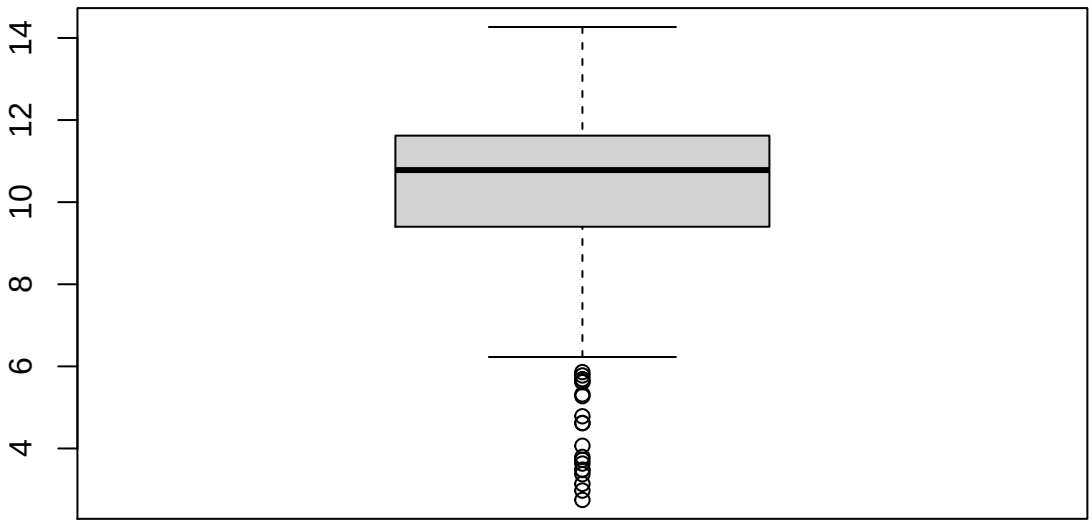
```
yield_data <- yield_data %>%
  filter(!year %in% c( "2016"),
         N_level != "275") #removing the years and  N rate to match with NDVI data
```

```r
yield_data <- yield_data %>%
  mutate(
    site_year = factor(site_year),
    year = factor(year),
    Block = factor(Block),
    MainPlot = factor(MainPlot),
    exp_plot_number = factor(exp_plot_number),
    N_level = factor(N_level),
    SubPlot = factor(SubPlot),
    TopDress = factor(TopDress),
    SeasonalNRate_f = factor(SeasonalNRate),
    N_level_kgha_f = factor(N_level_kgha),
    TopDress_kgha_f = factor(TopDress_kgha),
    SeasonalNRate_kgha_f = factor(SeasonalNRate_kgha)
    ) #changes these columns to factor

str(yield_data , give.attr = FALSE)
```

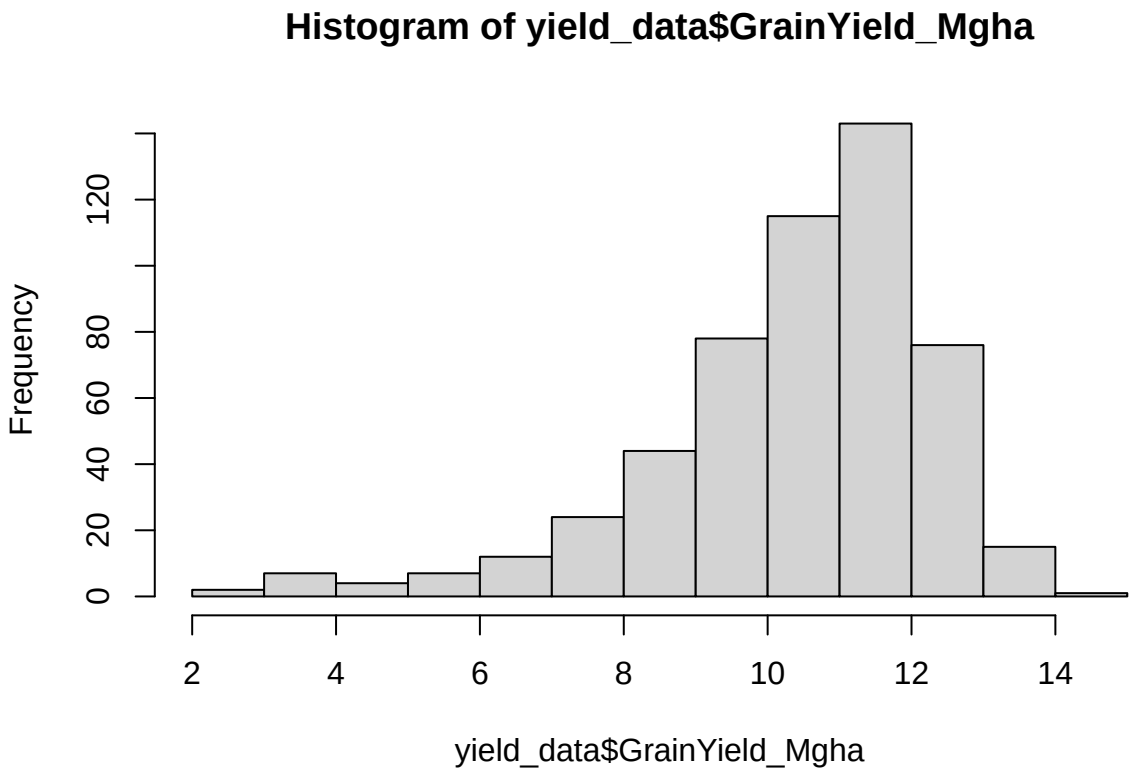
```
## spec_tbl_df[,38] [528 x 38] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ site_year            : Factor w/ 10 levels "Arbuckle-18",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ year                 : Factor w/ 3 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Block                : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 2 2 2 2 ...
##  $ MainPlot             : Factor w/ 7 levels "1","2","3","4",..: 1 2 3 4 6 7 2 3 4 5 ...
##  $ SubPlot              : Factor w/ 3 levels "1","2","3": 2 3 1 1 1 2 1 1 2 3 ...
##  $ exp_plot_number      : Factor w/ 27 levels "101","102","103",..: 1 2 3 4 6 7 9 10 11 12 ...
##  $ N_level              : Factor w/ 11 levels "0","45","75",..: 11 1 6 8 2 3 3 6 1 8 ...
##  $ TopDress             : Factor w/ 4 levels "0","25","30",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ SeasonalNRate        : num [1:528] 225 0 125 175 45 75 75 125 0 175 ...
##  $ N_level_kgha         : num [1:528] 225 0 125 175 45 75 75 125 0 175 ...
##  $ TopDress_kgha        : num [1:528] 0 0 0 0 0 0 0 0 0 0 ...
##  $ SeasonalNRate_kgha   : num [1:528] 225 0 125 175 45 75 75 125 0 175 ...
##  $ Received_TopDress    : chr [1:528] "NO" "NO" "NO" "NO" ...
##  $ tare                 : num [1:528] 1220 1220 1220 1220 1220 1220 1220 1220 1220 1220 ...
##  $ FW1PlusTare          : num [1:528] 4360 4818 5376 5598 4852 ...
##  $ FW2PlusTare          : num [1:528] 3254 1220 1220 1220 1220 ...
##  $ SSFWPlusTare         : num [1:528] 2324 1814 1994 2126 2106 ...
##  $ SSODW                : num [1:528] 632 479 570 562 583 ...
##  $ HarvestBagPlusTie    : num [1:528] 162 176 169 121 120 ...
##  $ Grain1PlusPaperBag1  : num [1:528] 230 173 211 233 265 ...
##  $ PaperBag1            : num [1:528] 7.96 7.96 7.96 7.96 7.96 7.96 7.96 7.96 7.96 7.96 ...
##  $ Grain2PlusPaperBag2  : num [1:528] 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 ...
##  $ PaperBag2            : num [1:528] 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 7.01 ...
##  $ StrawSampleSize      : num [1:528] 8.25 7.96 7.76 7.89 8.45 ...
##  $ StrawN               : num [1:528] 63.5 36 46.1 55.9 39.7 ...
##  $ GrainSampleSize      : num [1:528] 4.97 5.52 5.23 5.72 5.82 ...
##  $ GrainN               : num [1:528] 60.5 50 56 66.7 51.1 ...
##  $ DaysPI2Harvest       : num [1:528] 76 76 76 76 76 76 76 76 76 76 ...
##  $ SeedTray1.1          : num [1:528] 243 243 243 243 243 243 243 243 243 243 ...
##  $ SeedTray1.2          : num [1:528] 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6 ...
##  $ Grain3PlusSeedTray1  : num [1:528] 470 410 449 471 504 ...
##  $ MoistureContentGrain3: num [1:528] 11.2 11.2 11.2 11.2 11.2 11.2 11.2 11.2 11.2 11.2 ...
##  $ SeedTray2            : num [1:528] 244 244 244 244 244 ...
##  $ Grain4PlusSeedTray2  : num [1:528] 254 254 254 254 254 ...
```

```
##  $ SeasonalNRate_f      : Factor w/ 21 levels "0","25","30",..: 18 1 12 14 4 7 7 12 1 14 ...
##  $ N_level_kgha_f       : Factor w/ 11 levels "0","45","75",..: 10 1 5 8 2 3 3 5 1 8 ...
##  $ TopDress_kgha_f      : Factor w/ 4 levels "0","25","34",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ SeasonalNRate_kgha_f : Factor w/ 22 levels "0","25","34",..: 18 1 11 15 4 7 7 11 1 15 ...
```

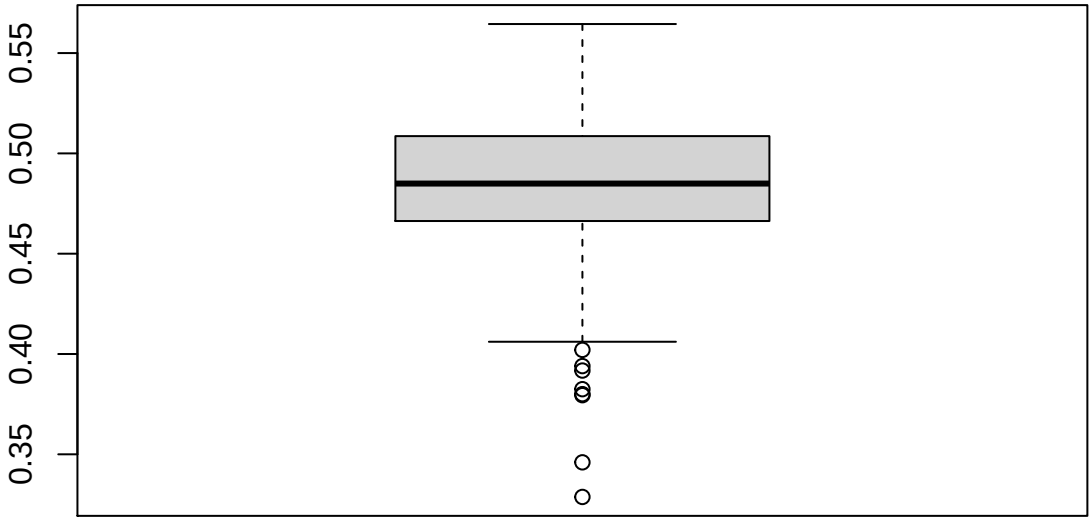
```
yield_data <- yield_data %>%
  mutate(
    FW1net = FW1PlusTare - tare,
    FW2net = FW2PlusTare - tare,
    TotalFW = FW1net + FW2net,
    SSFWnet = SSFWPlusTare - tare,
    Ratio = SSFWnet / TotalFW,
    SSODWnet = SSODW - HarvestBagPlusTie,
    SeedTray1 = SeedTray1.1 + SeedTray1.2, #adds the decimal to the 243g to get the tare weight for the
    Grain3net = Grain3PlusSeedTray1 - SeedTray1, #subtract tare of seed tray from grain3. Grain3 is the
    Grain4net = Grain4PlusSeedTray2 - SeedTray2, #grain4 is the amount of grain removed for ballmilling
    Grain2net = Grain2PlusPaperBag2 - PaperBag2, #yield component grain sample
    Grain2net = Grain2net * Ratio, #this essentially subsamples the yield component grain sample
    GrainNet = Grain3net + Grain4net + Grain2net, #add the grain removed for ball milling and yield com
    GrainRing = GrainNet / Ratio, #the amount of grain in the entire m^2 ring in grams
    GrainYield = GrainRing * 10, #g/m^2 to kg/ha
    GrainYield_kgha = GrainYield * ((100-MoistureContentGrain3)/86),#corrects for 14% moisture based on
    GrainYield_Mgha = GrainYield_kgha / 1000 , #converts kg/ha to Mg/ha
    Grain5 = GrainRing * ((100-MoistureContentGrain3)/98.1), #grain in the ring if the subsample was at
    Grain6 = GrainNet * ((100-MoistureContentGrain3)/98.1), #grain in the subsample if it was at 1.9% m
    StrawSS = SSODWnet - Grain6 , #just straw in subsample in grams
    StrawRing = StrawSS / Ratio, #straw in ring in grams i.e g/m2
    StrawNcon = StrawN / StrawSampleSize,
    StrawNup = (StrawRing * StrawNcon) / 100, #straw Nup divide by 100 to convert mg/m2 to kg/ha - this
    GrainNcon = (GrainN / GrainSampleSize), #grain in ring in kg/ha
    GrainNup = (Grain5 * GrainNcon) / 100, #grain Nup divide by 100 to convert mg N/m2 to kg N/ha
    TotalSeasonalNup = StrawNup + GrainNup, #in kg/ha
    HarvestIndex = Grain5 / (Grain5 + StrawRing),
    Moisture = SSFWnet / SSODWnet
    )

boxplot(yield_data$GrainYield_Mgha)
```



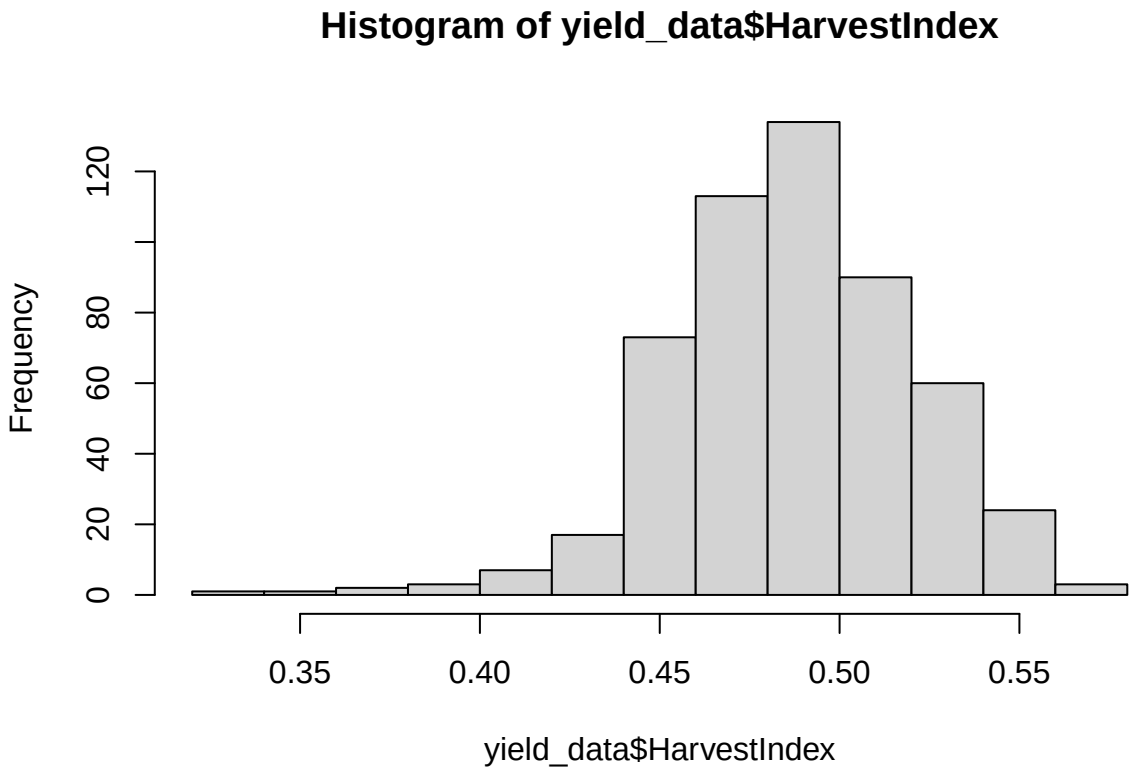
```
hist(yield_data$GrainYield_Mgha)
```



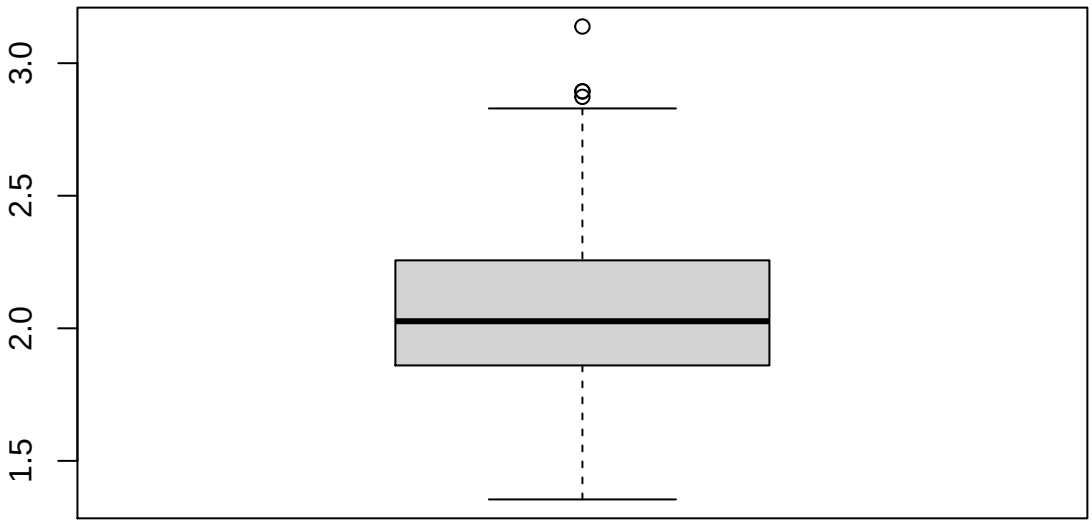

```
boxplot(yield_data$HarvestIndex)
```



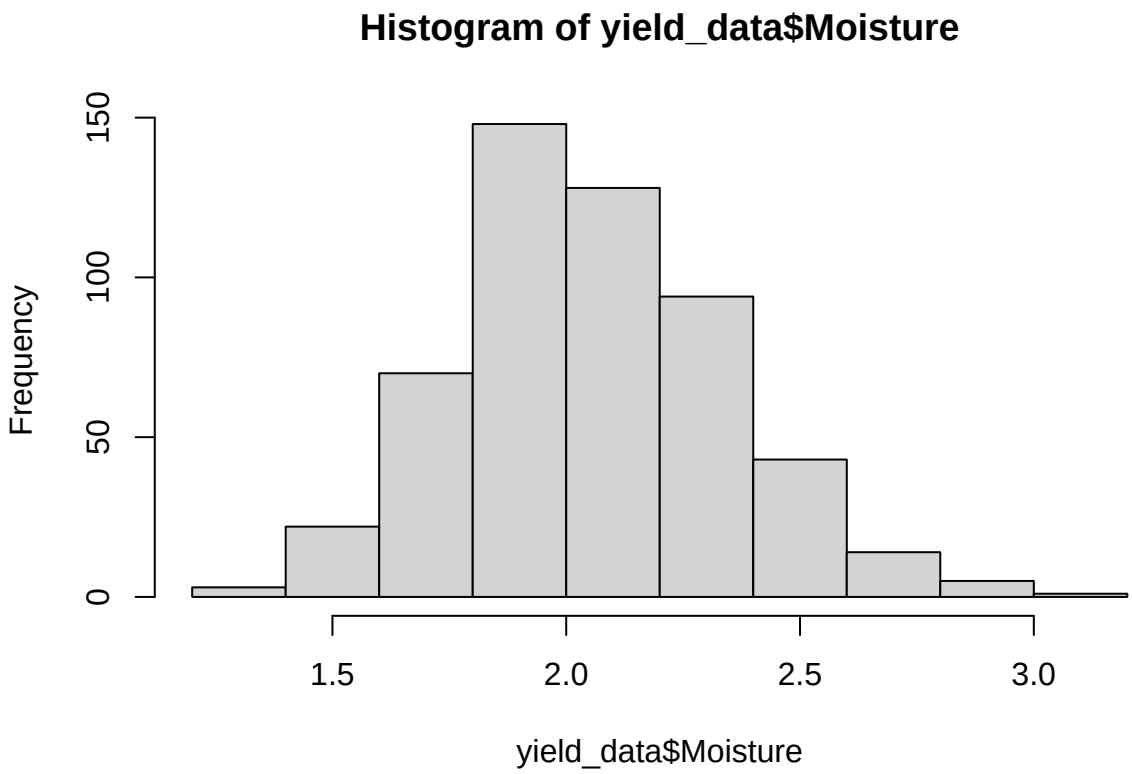

```
hist(yield_data$HarvestIndex)
```




```
boxplot(yield_data$Moisture)
```




```
hist(yield_data$Moisture)
```




```
#the data looks good - don't see any unusual values


yield_data <- dplyr::select(yield_data,
                            site_year,
                            year,
                            Block,
                            MainPlot,
                            SubPlot,
                            exp_plot_number,
                            N_level_kgha,
                            N_level_kgha_f,
                            TopDress_kgha,
                            TopDress_kgha_f,
                            GrainYield_Mgha
                            )


gs_data <- full_join(gs_ndvi_data , yield_data)
```

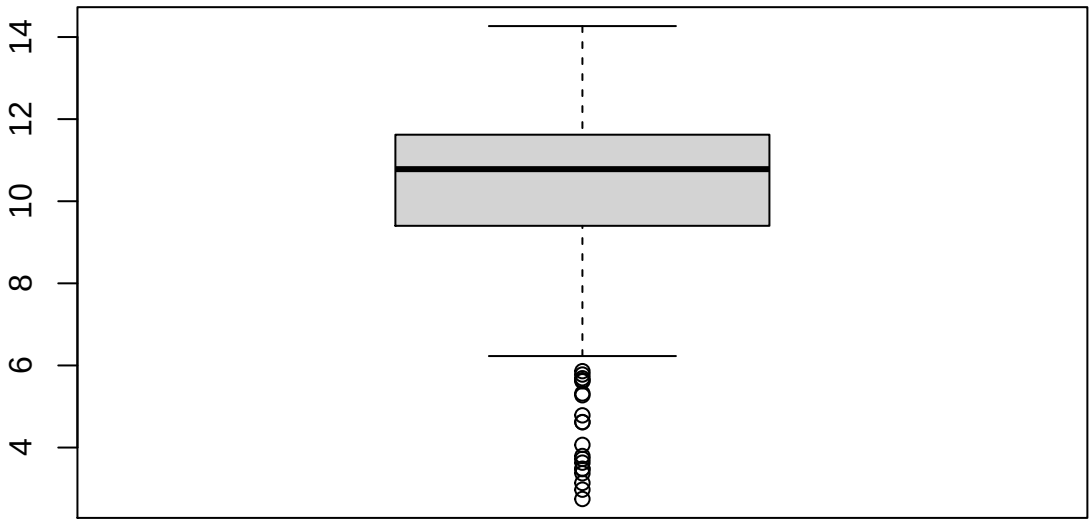
```
## Joining, by = c("site_year", "year", "exp_plot_number", "N_level_kgha")
```

```
gs_data <- dplyr::select(gs_data,
                site_year,
                year,
                exp_plot_number,
                Block,
                MainPlot,
                SubPlot,
                N_level_kgha,
```

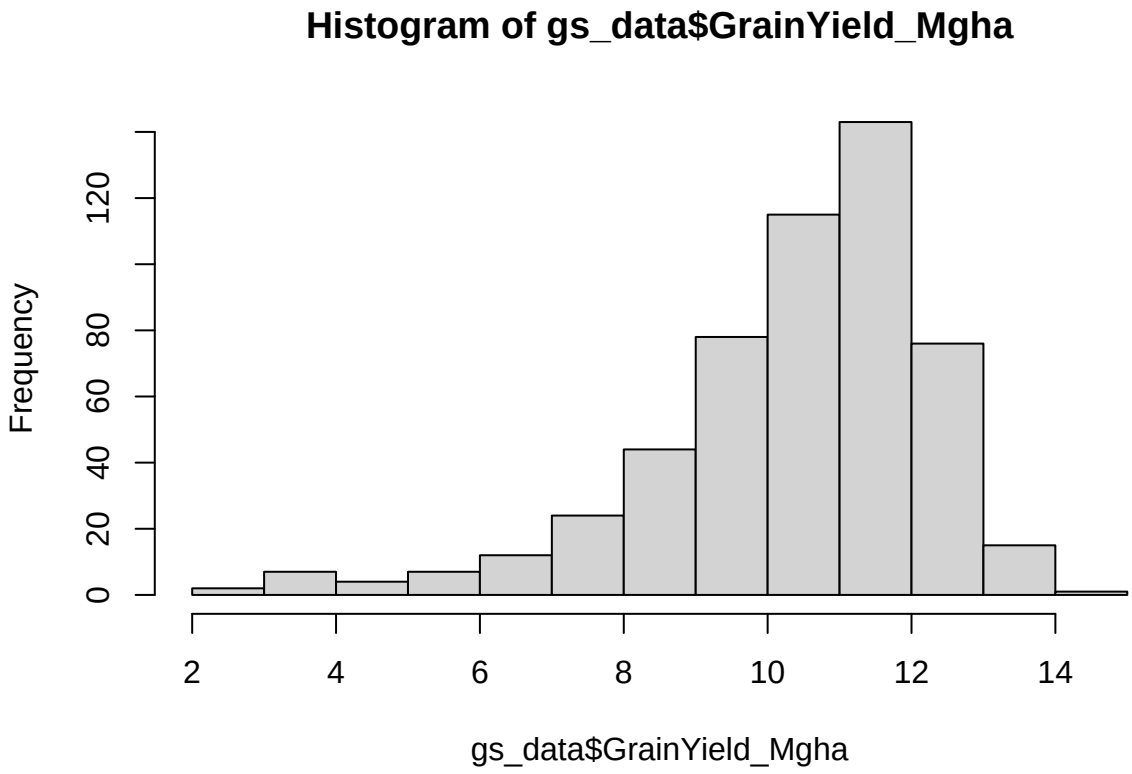
```
                N_level_kgha_f,
                TopDress_kgha,
                TopDress_kgha_f ,
                PI_N_Uptake,
                NDVI,
                GrainYield_Mgha) #reorders the columns

gs_data$site_year <- as.factor(gs_data$site_year)

str(gs_data , give.attr = FALSE)
```

```
## rowwise_df[,13] [528 x 13] (S3: rowwise_df/tbl_df/tbl/data.frame)
##  $ site_year      : Factor w/ 10 levels "Nicolaus-17",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year           : Factor w/ 3 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ exp_plot_number: Factor w/ 27 levels "101","102","103",..: 1 1 1 2 2 2 3 3 3 4 ...
##  $ Block          : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...
##  $ MainPlot       : Factor w/ 7 levels "1","2","3","4",..: 1 1 1 2 2 2 3 3 3 4 ...
##  $ SubPlot        : Factor w/ 3 levels "1","2","3": 2 1 3 3 2 1 1 3 2 1 ...
##  $ N_level_kgha   : num [1:528] 225 225 225 0 0 0 125 125 125 175 ...
##  $ N_level_kgha_f : Factor w/ 11 levels "0","45","75",..: 10 10 10 1 1 1 5 5 5 8 ...
##  $ TopDress_kgha  : num [1:528] 0 25 50 0 25 50 0 25 50 0 ...
##  $ TopDress_kgha_f: Factor w/ 4 levels "0","25","34",..: 1 2 4 1 2 4 1 2 4 1 ...
##  $ PI_N_Uptake    : num [1:528] 158 158 158 78 78 ...
##  $ NDVI           : num [1:528] 0.78 0.78 0.78 0.49 0.49 0.49 0.68 0.68 0.68 0.79 ...
##  $ GrainYield_Mgha: num [1:528] 11.4 12.2 12.1 11 10.9 ...
```

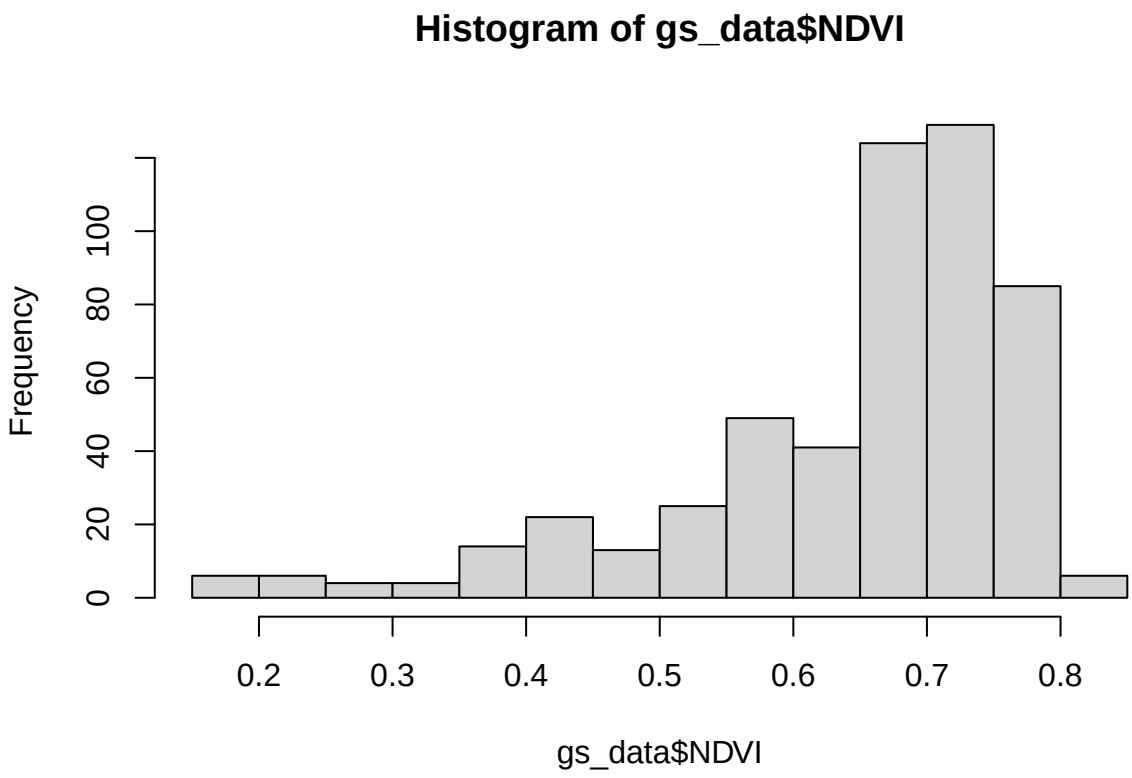
```
boxplot(gs_data$GrainYield_Mgha)
```



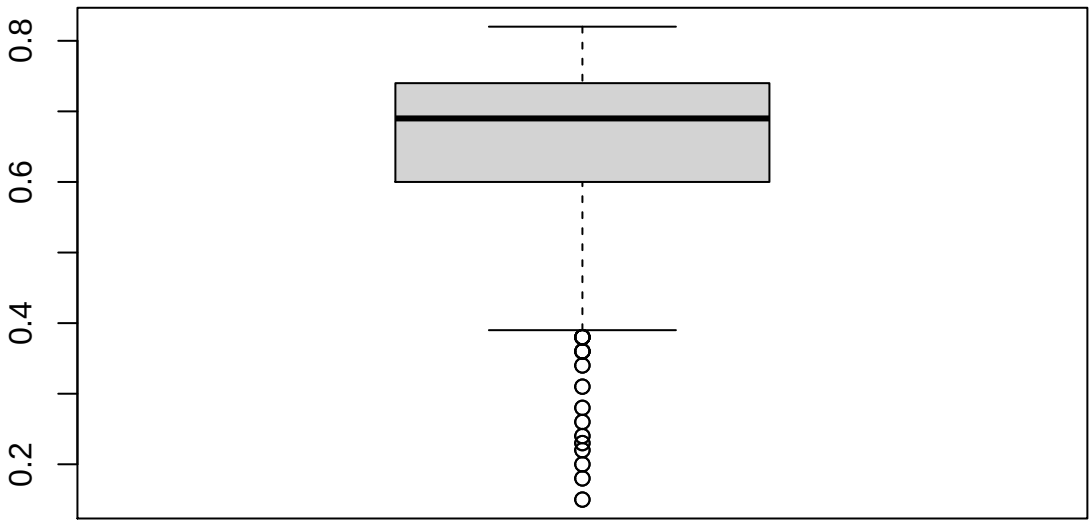

```
hist(gs_data$GrainYield_Mgha)
```



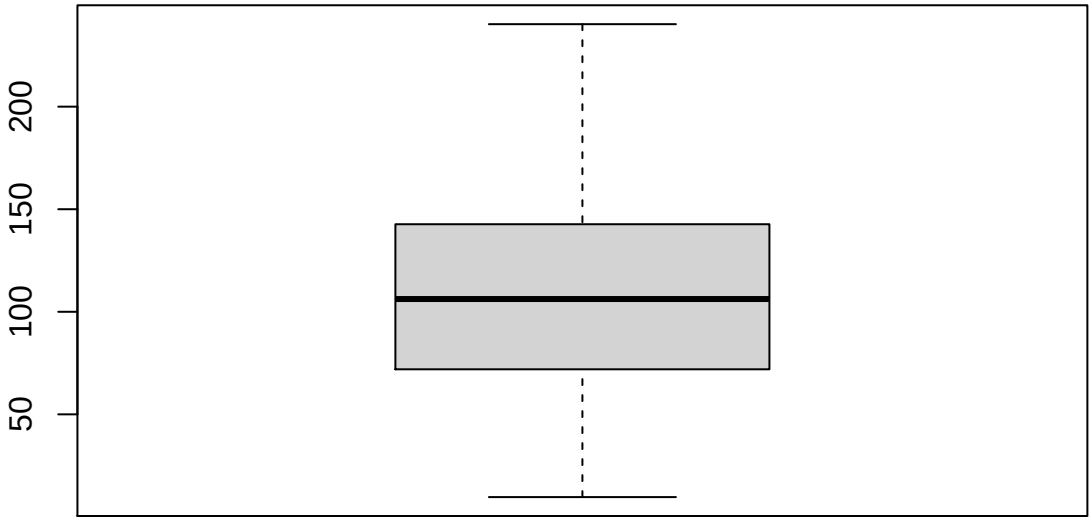

```
hist(gs_data$NDVI)
```



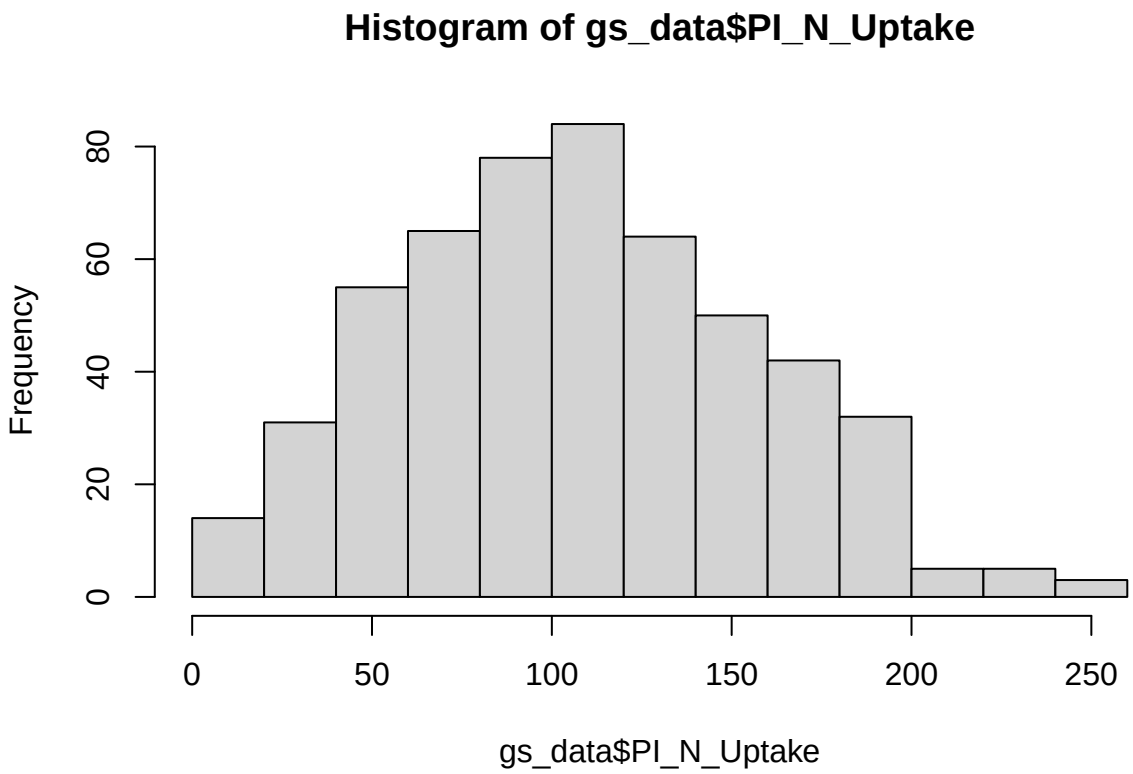

```
boxplot(gs_data$NDVI)
```




```
boxplot(gs_data$PI_N_Uptake)
```




```
hist(gs_data$PI_N_Uptake)
```




```
#Overall data looks good -- no errors of data entry
```

##Data for Table 3

```
table_3_data <- gs_data %>%
  select(site_year , TopDress_kgha_f , GrainYield_Mgha) %>%
  group_by(site_year , TopDress_kgha_f) %>%
  summarise(GrainYield_min = min(GrainYield_Mgha),
            GrainYield_max = max(GrainYield_Mgha),
            GrainYield_mean = mean(GrainYield_Mgha)) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'site_year'. You can override using the `.groups` argument.
```

```
table_3_data$GrainYield_min <- round(table_3_data$GrainYield_min , digits = 1)
table_3_data$GrainYield_max <- round(table_3_data$GrainYield_max , digits = 1)
table_3_data$GrainYield_mean <- round(table_3_data$GrainYield_mean , digits = 1)
```

## Calculating GS RI

```
max_gs_ndvi <- gs_data %>%
  filter(N_level_kgha_f %in% c(225, 235) & TopDress_kgha_f == 0) %>%
  select(site_year , NDVI) %>%
  group_by(site_year) %>%
  summarise(mean_NDVI = mean(NDVI))

max_gs_ndvi$mean_NDVI <- round(max_gs_ndvi$mean_NDVI , digits = 2)

nic17 <- subset(max_gs_ndvi, site_year == "Nicolaus-17")
nic17maxNDVI <- nic17$mean_NDVI
```

```
nic17maxNDVI <- as.numeric(nic17maxNDVI)
nic17maxNDVI
```

```
## [1] 0.78
```

```
wil17 <- subset(max_gs_ndvi, site_year == "Williams-17")
wil17maxNDVI <- wil17$mean_NDVI
wil17maxNDVI <- as.numeric(wil17maxNDVI)
wil17maxNDVI
```

```
## [1] 0.79
```

```
arb18 <- subset(max_gs_ndvi, site_year == "Arbuckle-18")
arb18maxNDVI <- arb18$mean_NDVI
arb18maxNDVI <- as.numeric(arb18maxNDVI)
arb18maxNDVI
```

```
## [1] 0.74
```

```
biggs18 <- subset(max_gs_ndvi, site_year == "Biggs-18")
biggs18maxNDVI <- biggs18$mean_NDVI
biggs18maxNDVI <- as.numeric(biggs18maxNDVI)
biggs18maxNDVI
```

```
## [1] 0.78
```

```
mry18 <- subset(max_gs_ndvi, site_year == "Marysville-18")
mry18maxNDVI <- mry18$mean_NDVI
mry18maxNDVI <- as.numeric(mry18maxNDVI)
mry18maxNDVI
```

```
## [1] 0.73
```

```
nic18 <- subset(max_gs_ndvi, site_year == "Nicolaus-18")
nic18maxNDVI <- nic18$mean_NDVI
nic18maxNDVI <- as.numeric(nic18maxNDVI)
nic18maxNDVI
```

```
## [1] 0.76
```

```
arb19 <- subset(max_gs_ndvi, site_year == "Arbuckle-19")
arb19maxNDVI <- arb19$mean_NDVI
arb19maxNDVI <- as.numeric(arb19maxNDVI)
arb19maxNDVI
```

```
## [1] 0.72
```

```
davis19 <- subset(max_gs_ndvi, site_year == "Davis-19")
davis19maxNDVI <- davis19$mean_NDVI
davis19maxNDVI <- as.numeric(davis19maxNDVI)
davis19maxNDVI
```

```
## [1] 0.61
```

```
mry19 <- subset(max_gs_ndvi, site_year == "Marysville-19")
mry19maxNDVI <- mry19$mean_NDVI
mry19maxNDVI <- as.numeric(mry19maxNDVI)
mry19maxNDVI
```

```
## [1] 0.75
```

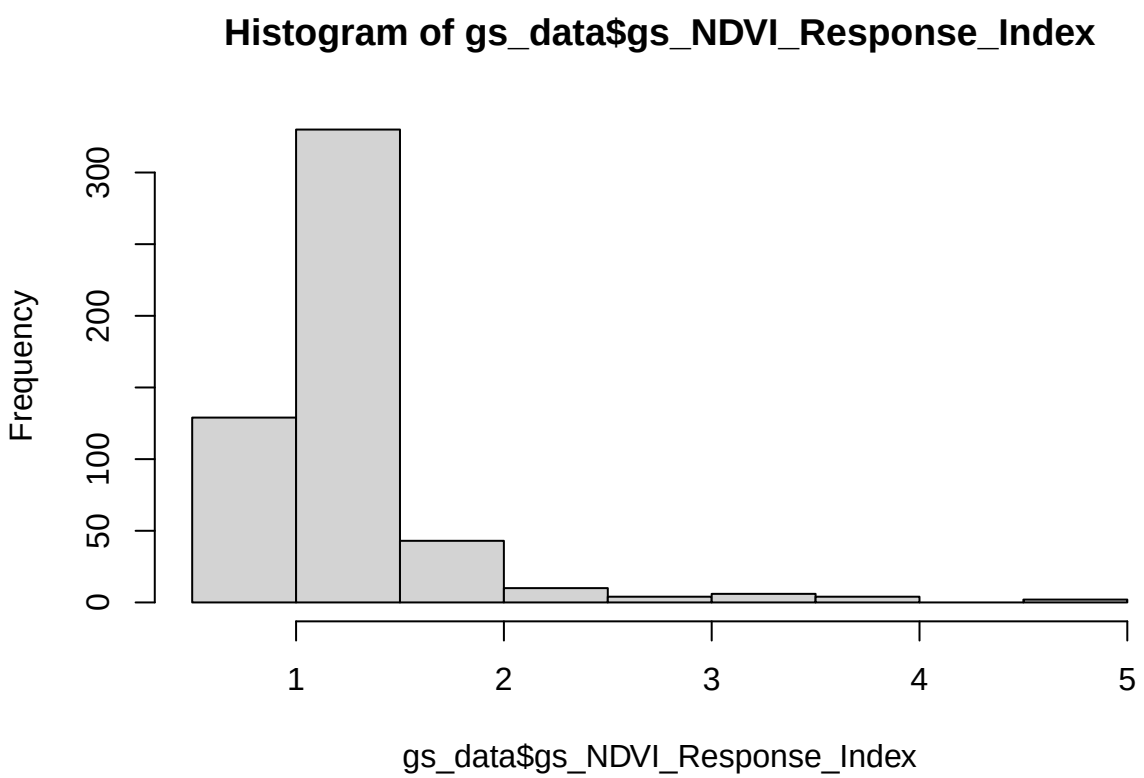
```
res19 <- subset(max_gs_ndvi, site_year == "RES-19")
res19maxNDVI <- res19$mean_NDVI
res19maxNDVI <- as.numeric(res19maxNDVI)
res19maxNDVI
```

```
## [1] 0.67
```

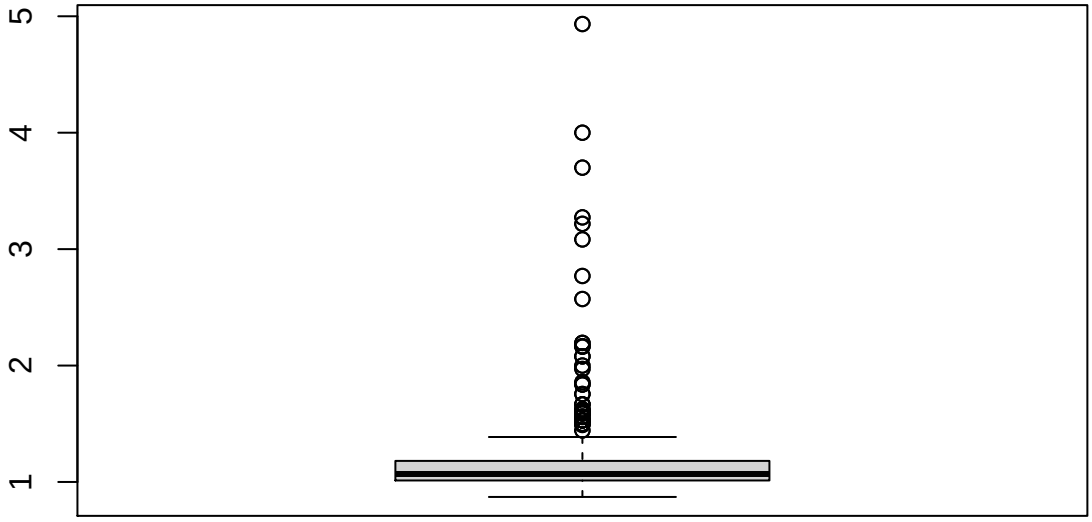
```
gs_data <- gs_data %>%
  mutate(max_NDVI = case_when(
                           site_year == "Nicolaus-17" ~ nic17maxNDVI ,
                           site_year == "Williams-17" ~ wil17maxNDVI ,
                           site_year == "Arbuckle-18" ~ arb18maxNDVI ,
                           site_year == "Biggs-18" ~ biggs18maxNDVI ,
                           site_year == "Marysville-18" ~ mry18maxNDVI ,
                           site_year == "Nicolaus-18" ~ nic18maxNDVI,
                           site_year == "Arbuckle-19" ~ arb19maxNDVI ,
                           site_year == "Davis-19" ~ davis19maxNDVI ,
                           site_year == "Marysville-19" ~ mry19maxNDVI ,
                           site_year == "RES-19" ~ res19maxNDVI)
  )

gs_data <- gs_data %>%
  mutate(gs_NDVI_Response_Index = case_when(
                           site_year == "Nicolaus-17" ~ nic17maxNDVI / NDVI,
                           site_year == "Williams-17" ~ wil17maxNDVI / NDVI,
                           site_year == "Arbuckle-18" ~ arb18maxNDVI / NDVI ,
                           site_year == "Biggs-18" ~ biggs18maxNDVI / NDVI ,
                           site_year == "Marysville-18" ~ mry18maxNDVI / NDVI ,
                           site_year == "Nicolaus-18" ~ nic18maxNDVI / NDVI,
                           site_year == "Arbuckle-19" ~ arb19maxNDVI / NDVI,
                           site_year == "Davis-19" ~ davis19maxNDVI / NDVI,
                           site_year == "Marysville-19" ~ mry19maxNDVI / NDVI,
                           site_year == "RES-19" ~ res19maxNDVI / NDVI
                         )) #calculates NDVI response index

str(gs_data , give.attr = FALSE)
```

```
## rowwise_df[,15] [528 x 15] (S3: rowwise_df/tbl_df/tbl/data.frame)
##  $ site_year             : Factor w/ 10 levels "Nicolaus-17",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year                  : Factor w/ 3 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ exp_plot_number       : Factor w/ 27 levels "101","102","103",..: 1 1 1 2 2 2 3 3 3 4 ...
##  $ Block                 : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...
##  $ MainPlot              : Factor w/ 7 levels "1","2","3","4",..: 1 1 1 2 2 2 3 3 3 4 ...
##  $ SubPlot               : Factor w/ 3 levels "1","2","3": 2 1 3 3 2 1 1 3 2 1 ...
##  $ N_level_kgha          : num [1:528] 225 225 225 0 0 0 125 125 125 175 ...
##  $ N_level_kgha_f        : Factor w/ 11 levels "0","45","75",..: 10 10 10 1 1 1 5 5 5 8 ...
##  $ TopDress_kgha         : num [1:528] 0 25 50 0 25 50 0 25 50 0 ...
##  $ TopDress_kgha_f       : Factor w/ 4 levels "0","25","34",..: 1 2 4 1 2 4 1 2 4 1 ...
##  $ PI_N_Uptake           : num [1:528] 158 158 158 78 78 ...
##  $ NDVI                  : num [1:528] 0.78 0.78 0.78 0.49 0.49 0.49 0.68 0.68 0.68 0.79 ...
##  $ GrainYield_Mgha       : num [1:528] 11.4 12.2 12.1 11 10.9 ...
##  $ max_NDVI              : num [1:528] 0.78 0.78 0.78 0.78 0.78 0.78 0.78 0.78 0.78 0.78 ...
##  $ gs_NDVI_Response_Index: num [1:528] 1 1 1 1.59 1.59 ...
```

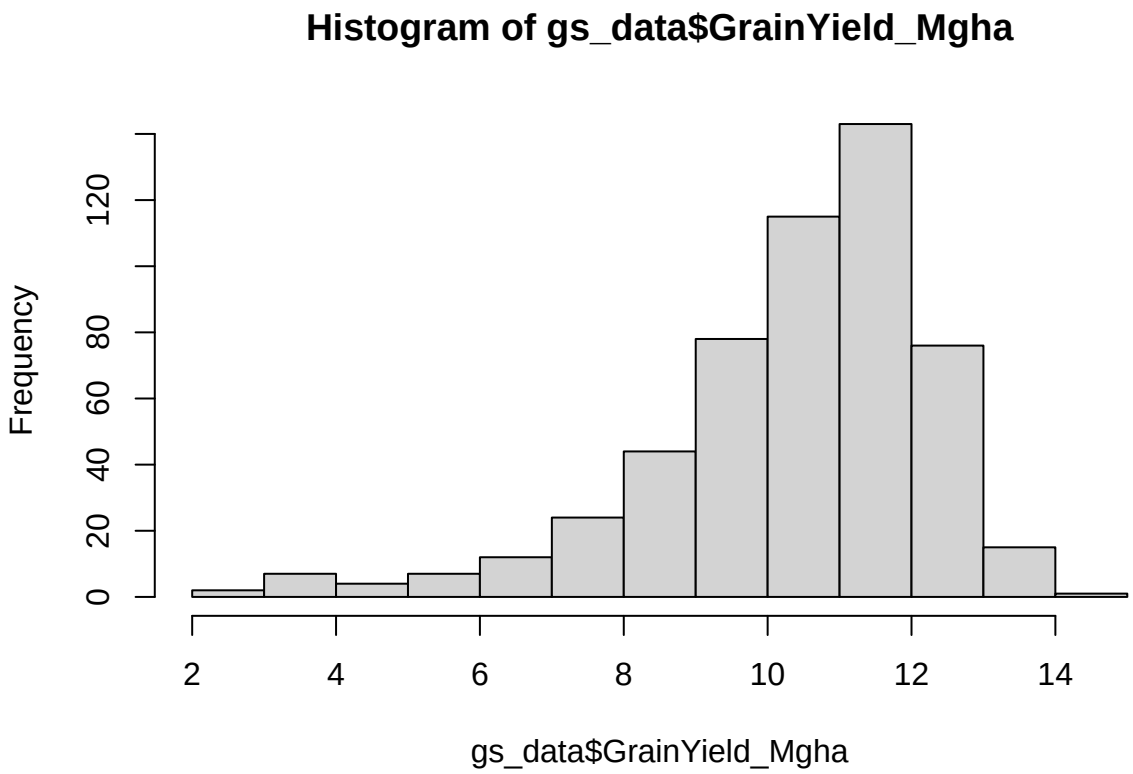
```
hist(gs_data$gs_NDVI_Response_Index)
```



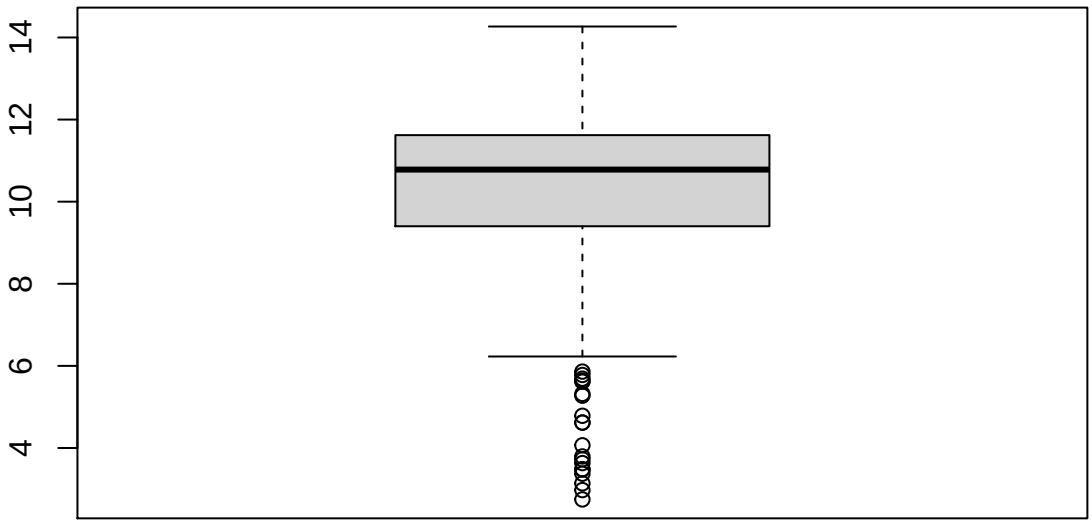

```
boxplot(gs_data$gs_NDVI_Response_Index)
```




```
hist(gs_data$GrainYield_Mgha)
```




```
boxplot(gs_data$GrainYield_Mgha)
```




```
gs_data$gs_NDVI_Response_Index[gs_data$gs_NDVI_Response_Index < 1] <- 1 #converts values less than 1, t
```

##Drone Data

```
drone_data <- read_csv(file = "DATA/PI_drone_data.csv")
```

```
## Rows: 248 Columns: 38
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): site_year
## dbl (37): year, exp_plot_number, Block, MainPlot, N_level, N_level_kgha, bio...
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
drone_data <- drone_data %>%
  filter(N_level_kgha != 275)

str(drone_data , give.attr = FALSE)
```

```
## spec_tbl_df[,38] [240 x 38] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ site_year         : chr [1:240] "Nicolaus-17" "Nicolaus-17" "Nicolaus-17" "Nicolaus-17" ...
##  $ year              : num [1:240] 2017 2017 2017 2017 2017 ...
##  $ exp_plot_number   : num [1:240] 101 102 103 104 106 107 202 203 204 205 ...
##  $ Block             : num [1:240] 1 1 1 1 1 1 2 2 2 2 ...
##  $ MainPlot          : num [1:240] 1 2 3 4 6 7 2 3 4 5 ...
##  $ N_level           : num [1:240] 225 0 125 175 45 75 75 125 0 175 ...
##  $ N_level_kgha      : num [1:240] 225 0 125 175 45 75 75 125 0 175 ...
##  $ biomass_plus_bag_g: num [1:240] 361 264 318 360 285 ...
##  $ ring_size         : num [1:240] 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
##  $ paper_bag_g       : num [1:240] 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 ...
##  $ num_of_paper_bags : num [1:240] 1 1 1 1 1 1 1 1 1 1 ...
##  $ sample_weight_mg  : num [1:240] 3.17 3.42 3.09 3.07 3.2 ...
##  $ sample_N_ug       : num [1:240] 79.4 61.1 80.7 95.9 63.6 ...
##  $ greenmean         : num [1:240] 0.0467 0.0581 0.0498 0.0488 0.0608 0.0589 0.0537 0.0488 0.0598 0.0
##  $ greenmedia        : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ greenstdev        : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ greenmin          : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ greenmax          : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ bluemean          : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ bluemedian        : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ bluestdev         : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ bluemin           : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ bluemax           : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ redmean           : num [1:240] 0.023 0.0286 0.0243 0.0237 0.0292 0.0281 0.0263 0.0237 0.0293 0.02
##  $ redmedian         : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ redstdev          : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ redmin            : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ redmax            : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ edgemean          : num [1:240] 0.0738 0.0867 0.0768 0.0764 0.0903 0.0877 0.0812 0.0754 0.0873 0.0
##  $ edgemedian        : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ edgestdev         : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ edgemin           : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ edgemax           : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ nirmean           : num [1:240] 0.55 0.357 0.482 0.537 0.431 ...
##  $ nirmedian         : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ nirstdev          : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ nirmin            : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ nirmax            : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
```

```
drone_data <- drone_data %>%
  mutate(year = factor(year) ,
         exp_plot_number = factor(exp_plot_number) ,
         Block = factor(Block) ,
         MainPlot = factor(MainPlot) ,
         N_level = factor(N_level) ,
```

```
        N_level_kgha_f = factor(N_level_kgha)
  )

drone_data$site_year <- factor(drone_data$site_year , levels = c("Nicolaus-17" , "Williams-17" , "Arbuc

str(drone_data , give.attr = FALSE)
```

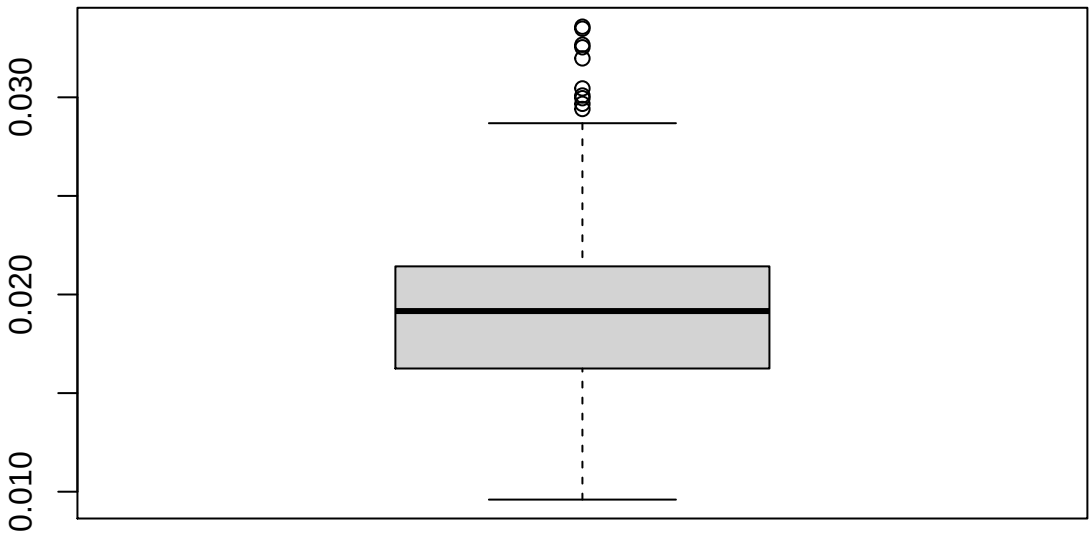
```
## spec_tbl_df[,39] [240 x 39] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ site_year         : Factor w/ 10 levels "Nicolaus-17",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year              : Factor w/ 3 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ exp_plot_number   : Factor w/ 27 levels "101","102","103",..: 1 2 3 4 6 7 9 10 11 12 ...
##  $ Block             : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 2 2 2 2 ...
##  $ MainPlot          : Factor w/ 7 levels "1","2","3","4",..: 1 2 3 4 6 7 2 3 4 5 ...
##  $ N_level           : Factor w/ 11 levels "0","45","75",..: 11 1 6 8 2 3 3 6 1 8 ...
##  $ N_level_kgha      : num [1:240] 225 0 125 175 45 75 75 125 0 175 ...
##  $ biomass_plus_bag_g: num [1:240] 361 264 318 360 285 ...
##  $ ring_size         : num [1:240] 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
##  $ paper_bag_g       : num [1:240] 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 ...
##  $ num_of_paper_bags : num [1:240] 1 1 1 1 1 1 1 1 1 1 ...
##  $ sample_weight_mg  : num [1:240] 3.17 3.42 3.09 3.07 3.2 ...
##  $ sample_N_ug       : num [1:240] 79.4 61.1 80.7 95.9 63.6 ...
##  $ greenmean         : num [1:240] 0.0467 0.0581 0.0498 0.0488 0.0608 0.0589 0.0537 0.0488 0.0598 0.0
##  $ greenmedia        : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ greenstdev        : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ greenmin          : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ greenmax          : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ bluemean          : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ bluemedian        : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ bluestdev         : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ bluemin           : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ bluemax           : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ redmean           : num [1:240] 0.023 0.0286 0.0243 0.0237 0.0292 0.0281 0.0263 0.0237 0.0293 0.02
##  $ redmedian         : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ redstdev          : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ redmin            : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ redmax            : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ edgemean          : num [1:240] 0.0738 0.0867 0.0768 0.0764 0.0903 0.0877 0.0812 0.0754 0.0873 0.0
##  $ edgemedian        : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ edgestdev         : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ edgemin           : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ edgemax           : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ nirmean           : num [1:240] 0.55 0.357 0.482 0.537 0.431 ...
##  $ nirmedian         : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ nirstdev          : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ nirmin            : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ nirmax            : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ N_level_kgha_f    : Factor w/ 11 levels "0","45","75",..: 10 1 5 8 2 3 3 5 1 8 ...
```

```
drone_data <- dplyr::select(drone_data ,
                     site_year,
                     year,
                     exp_plot_number,
                     Block,
                     MainPlot,
```

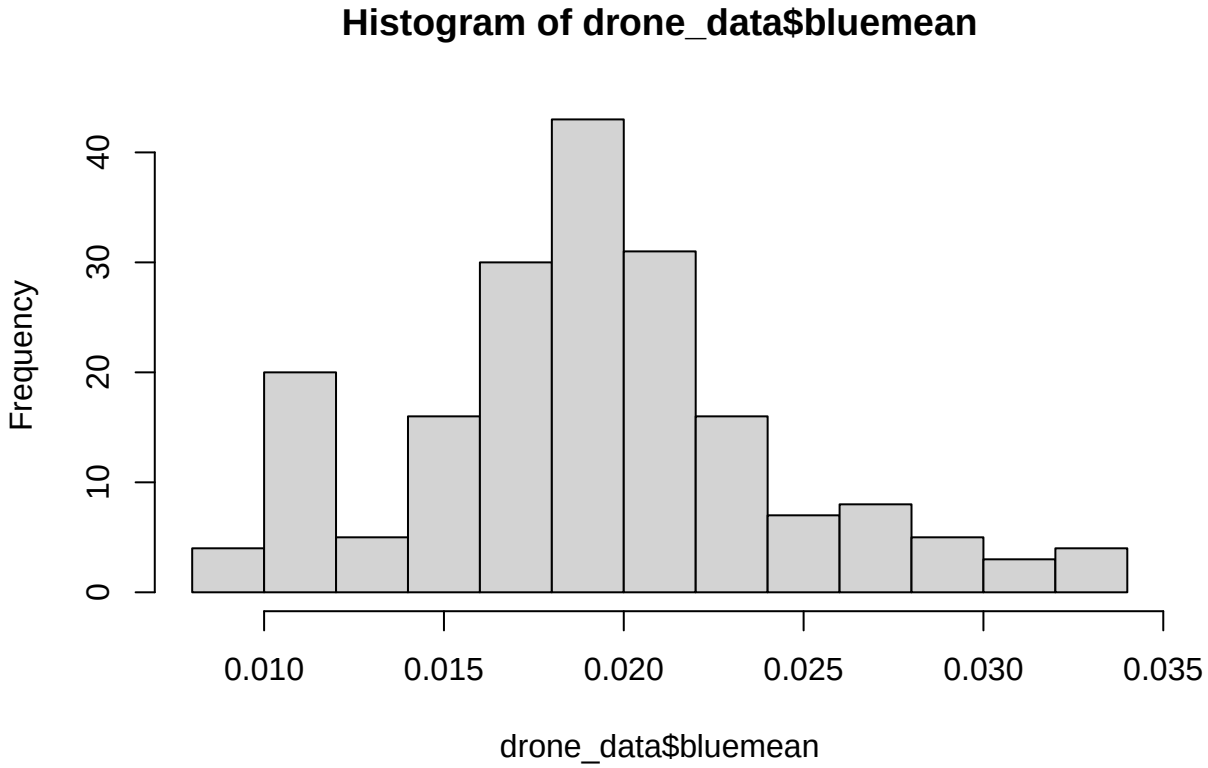
```
                    N_level,
                    N_level_kgha,
                    N_level_kgha_f,
                    biomass_plus_bag_g,
                    ring_size,
                    paper_bag_g,
                    num_of_paper_bags,
                    sample_weight_mg,
                    sample_N_ug,
                    bluemean,
                    greenmean,
                    redmean,
                    edgemean,
                    nirmean
                    )#selects the relevant columns

str(drone_data , give.attr = FALSE)
```

```
## tibble[,19] [240 x 19] (S3: tbl_df/tbl/data.frame)
##  $ site_year         : Factor w/ 10 levels "Nicolaus-17",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year              : Factor w/ 3 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ exp_plot_number   : Factor w/ 27 levels "101","102","103",..: 1 2 3 4 6 7 9 10 11 12 ...
##  $ Block             : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 2 2 2 2 ...
##  $ MainPlot          : Factor w/ 7 levels "1","2","3","4",..: 1 2 3 4 6 7 2 3 4 5 ...
##  $ N_level           : Factor w/ 11 levels "0","45","75",..: 11 1 6 8 2 3 3 6 1 8 ...
##  $ N_level_kgha      : num [1:240] 225 0 125 175 45 75 75 125 0 175 ...
##  $ N_level_kgha_f    : Factor w/ 11 levels "0","45","75",..: 10 1 5 8 2 3 3 5 1 8 ...
##  $ biomass_plus_bag_g: num [1:240] 361 264 318 360 285 ...
##  $ ring_size         : num [1:240] 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
##  $ paper_bag_g       : num [1:240] 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 ...
##  $ num_of_paper_bags : num [1:240] 1 1 1 1 1 1 1 1 1 1 ...
##  $ sample_weight_mg  : num [1:240] 3.17 3.42 3.09 3.07 3.2 ...
##  $ sample_N_ug       : num [1:240] 79.4 61.1 80.7 95.9 63.6 ...
##  $ bluemean          : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ greenmean         : num [1:240] 0.0467 0.0581 0.0498 0.0488 0.0608 0.0589 0.0537 0.0488 0.0598 0.0
##  $ redmean           : num [1:240] 0.023 0.0286 0.0243 0.0237 0.0292 0.0281 0.0263 0.0237 0.0293 0.01
##  $ edgemean          : num [1:240] 0.0738 0.0867 0.0768 0.0764 0.0903 0.0877 0.0812 0.0754 0.0873 0.0
##  $ nirmean           : num [1:240] 0.55 0.357 0.482 0.537 0.431 ...
```

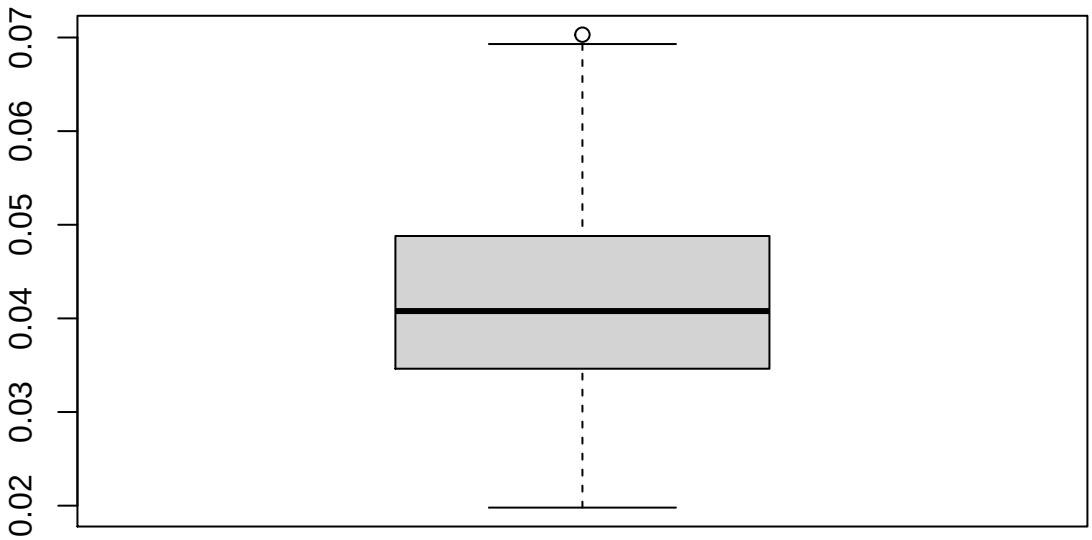
```
#visualize drone_data to look for outliers

boxplot(drone_data$bluemean)
```



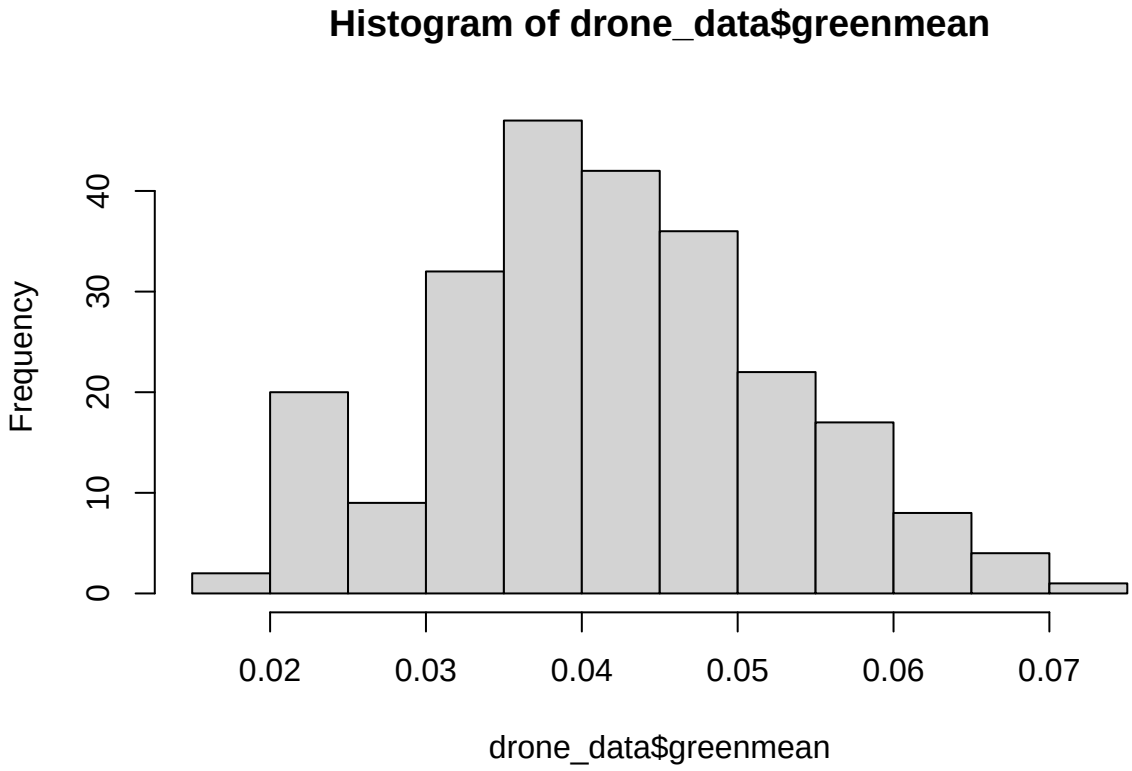

```
hist(drone_data$bluemean)
```



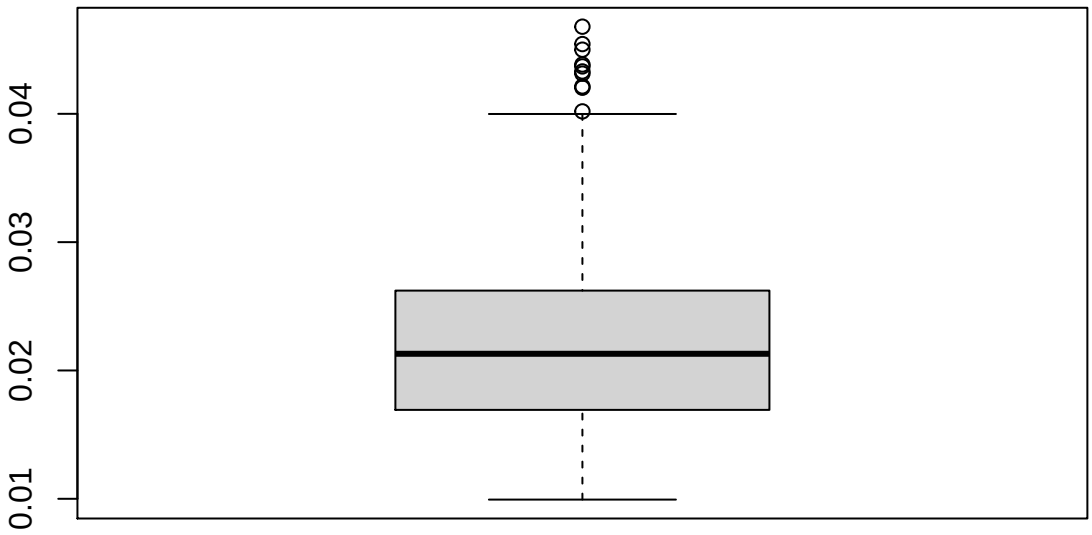

```
boxplot(drone_data$greenmean)
```



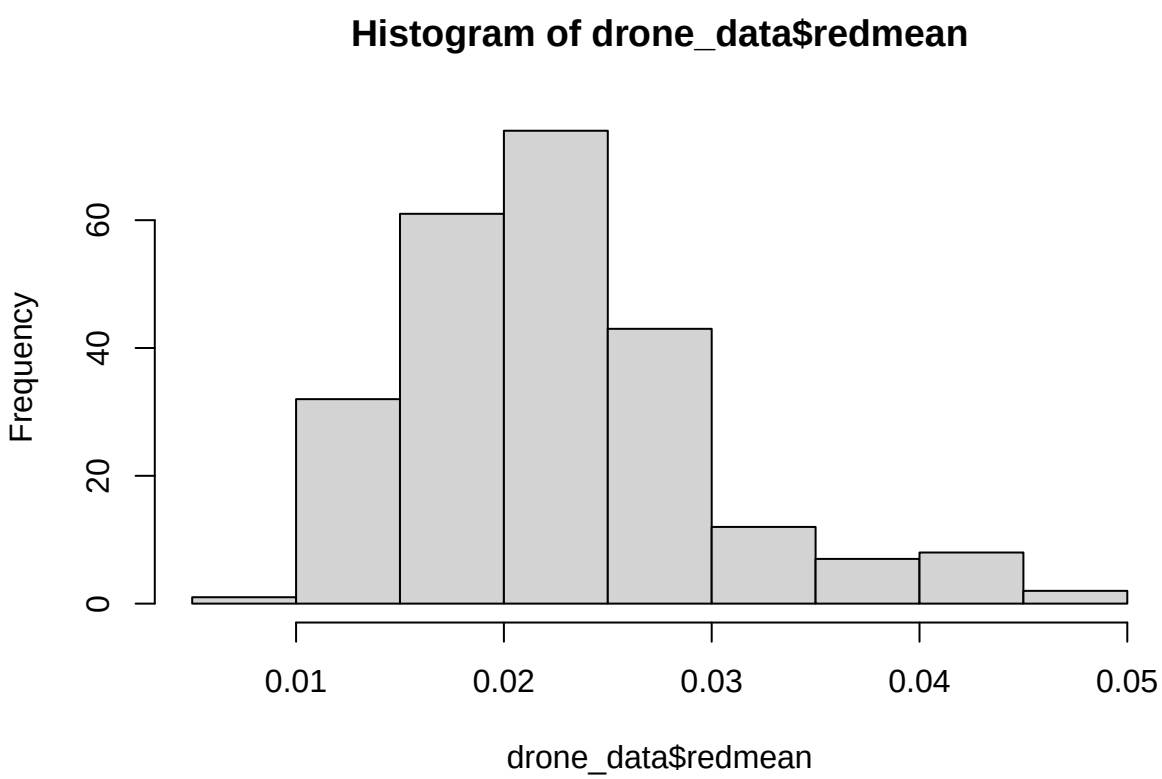

```
hist(drone_data$greenmean)
```



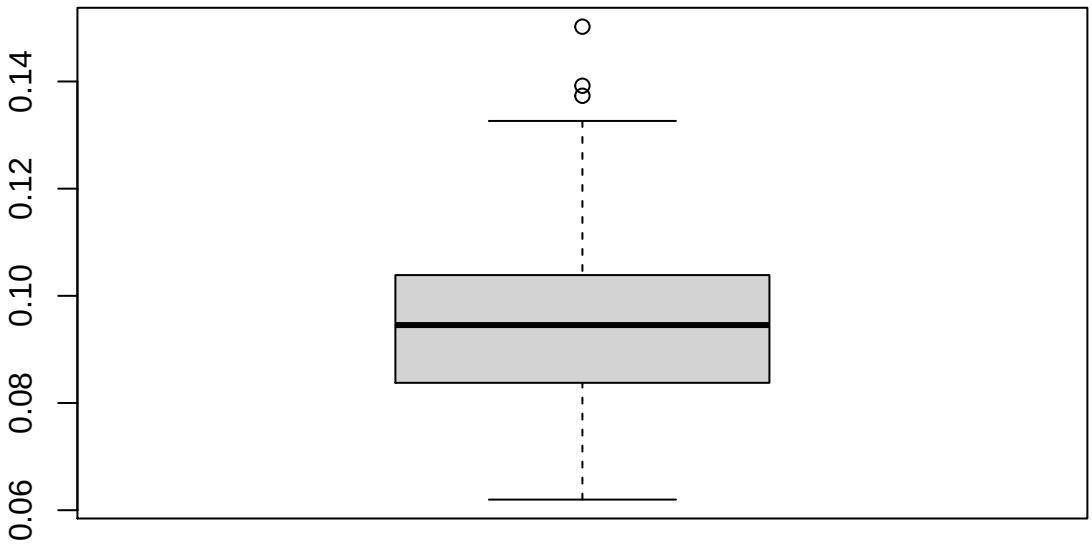

```
boxplot(drone_data$redmean)
```



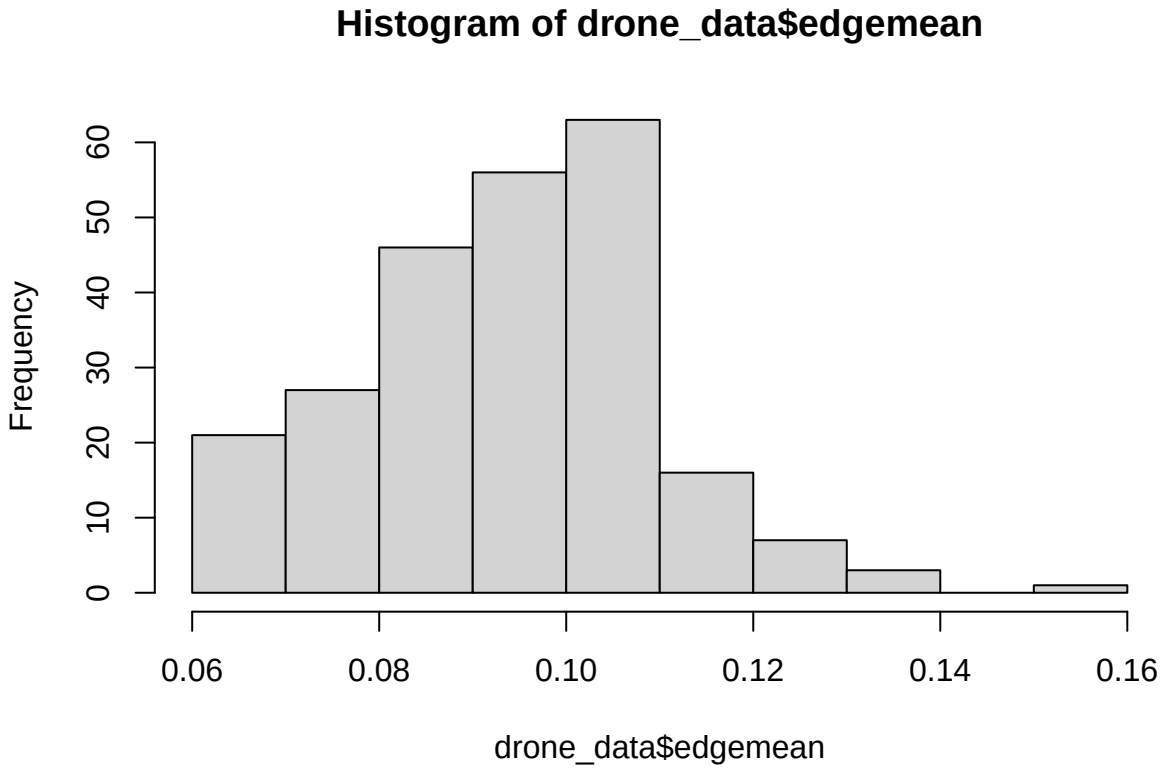

```
hist(drone_data$redmean)
```



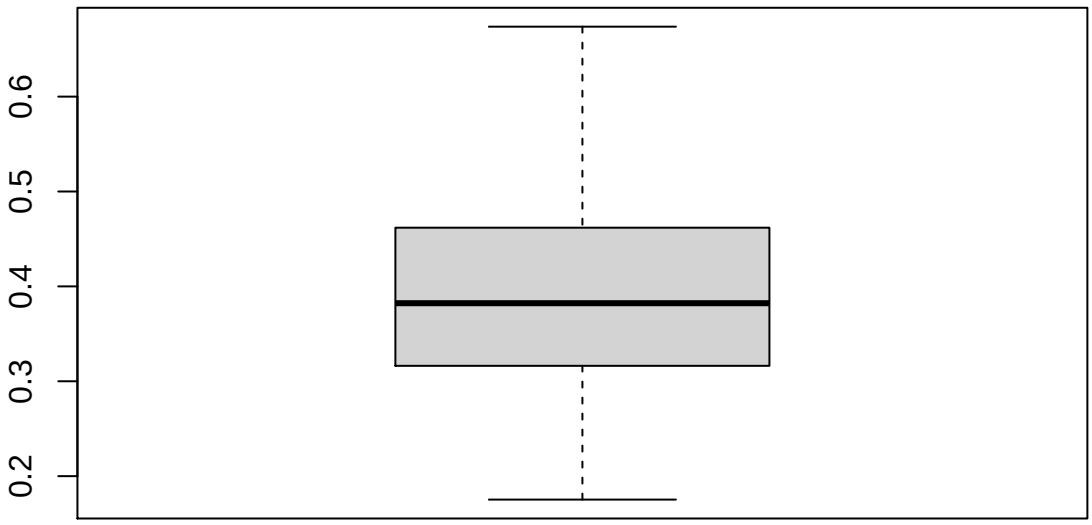

```
boxplot(drone_data$edgemean)
```



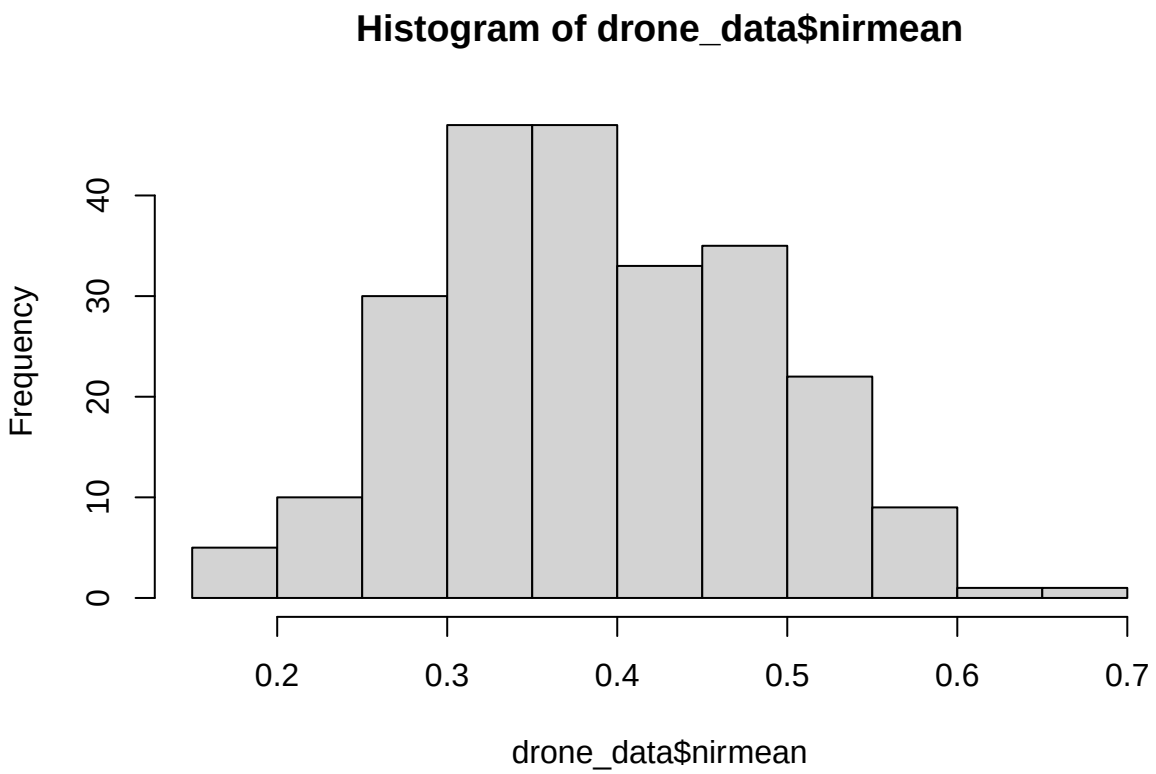

```
hist(drone_data$edgemean)
```



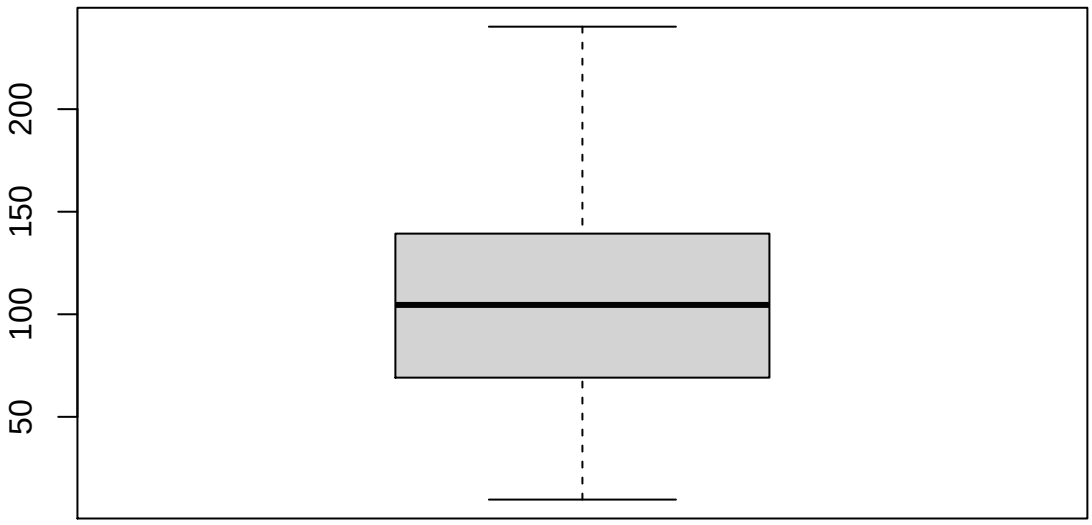

```
boxplot(drone_data$nirmean)
```



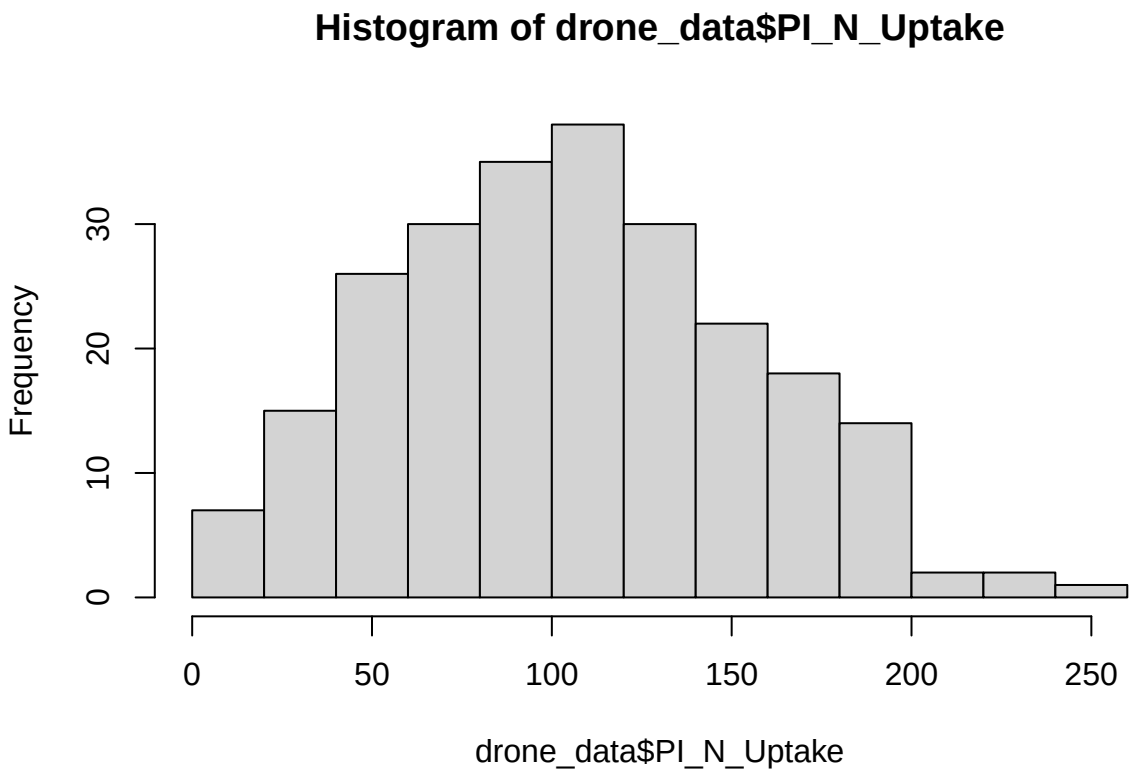

```
hist(drone_data$nirmean)
```



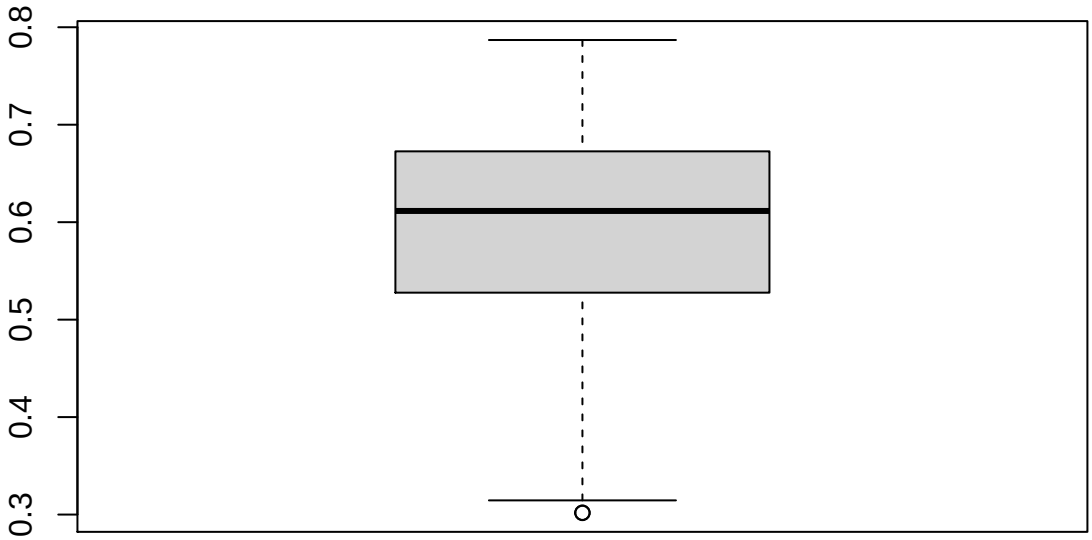

```
drone_data <-  drone_data %>%
  mutate( biomass_dry_wt = biomass_plus_bag_g - (paper_bag_g * num_of_paper_bags) ,
          aboveground_biomass = (biomass_dry_wt / ring_size) * 10 , #ring size 0.5 m^2 biomass in kg pe
          n_content = sample_N_ug / sample_weight_mg ,
          PI_N_Uptake = (aboveground_biomass * n_content) / 1000 #n uptake in kg per ha
          )#processes the data2

boxplot(drone_data$PI_N_Uptake)
```



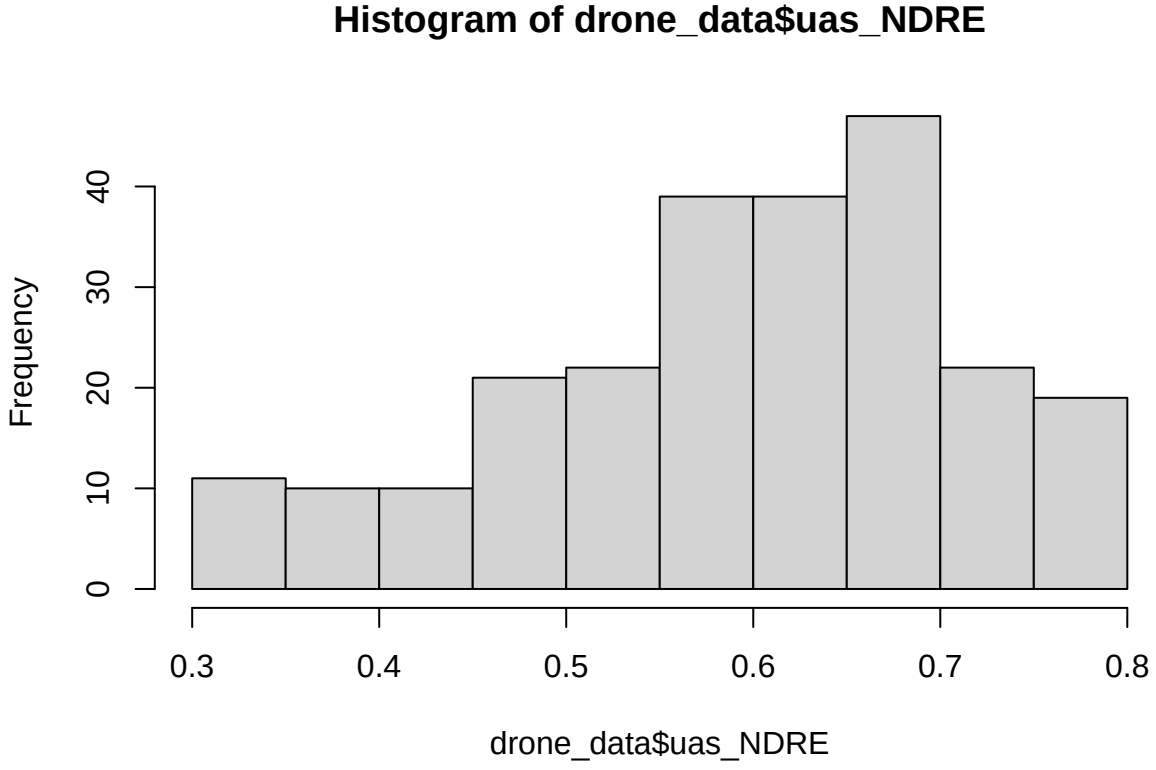

```
hist(drone_data$PI_N_Uptake)
```



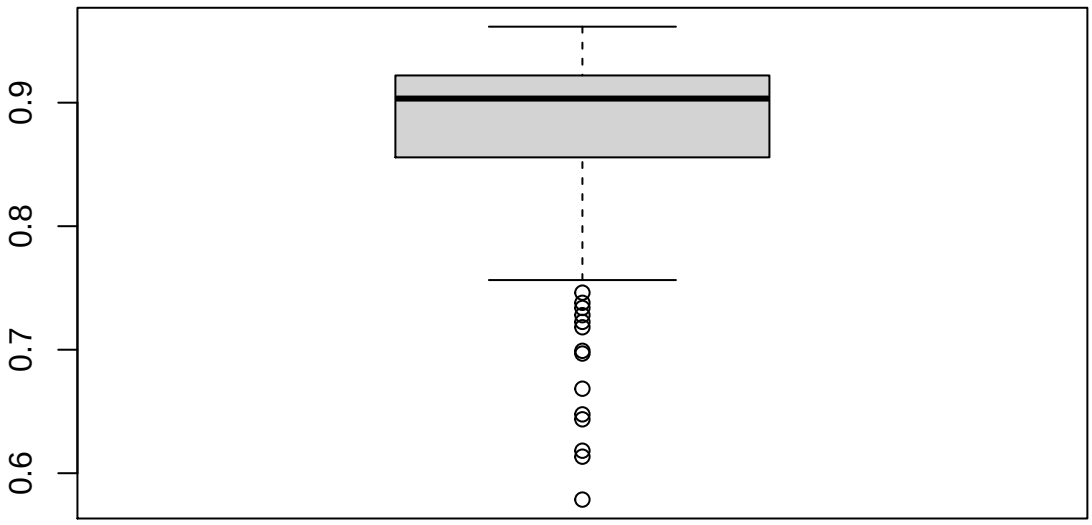

```
drone_data <- drone_data %>%
  mutate(uas_NDRE = ((nirmean - edgemean) / (nirmean + edgemean)) ,
         uas_NDVI = ((nirmean - redmean) / (nirmean + redmean))
         ) #calculates NDRE and NDVI

boxplot(drone_data$uas_NDRE)
```



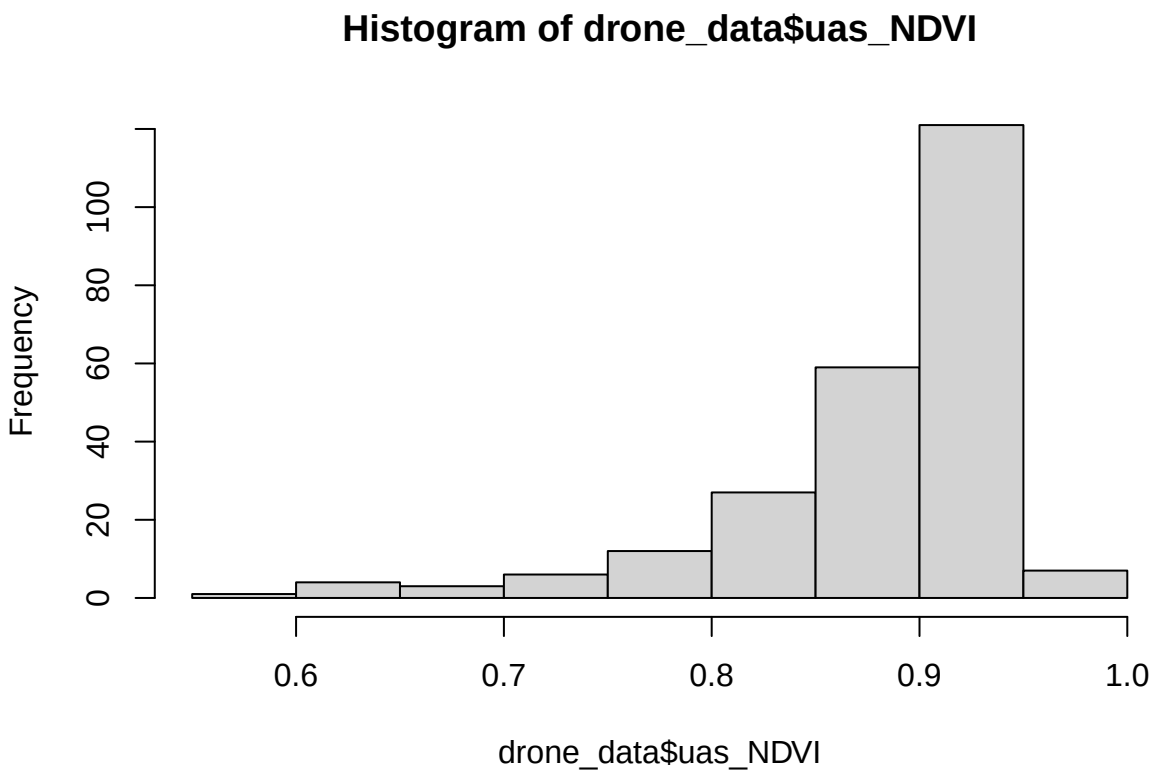

```
hist(drone_data$uas_NDRE)
```



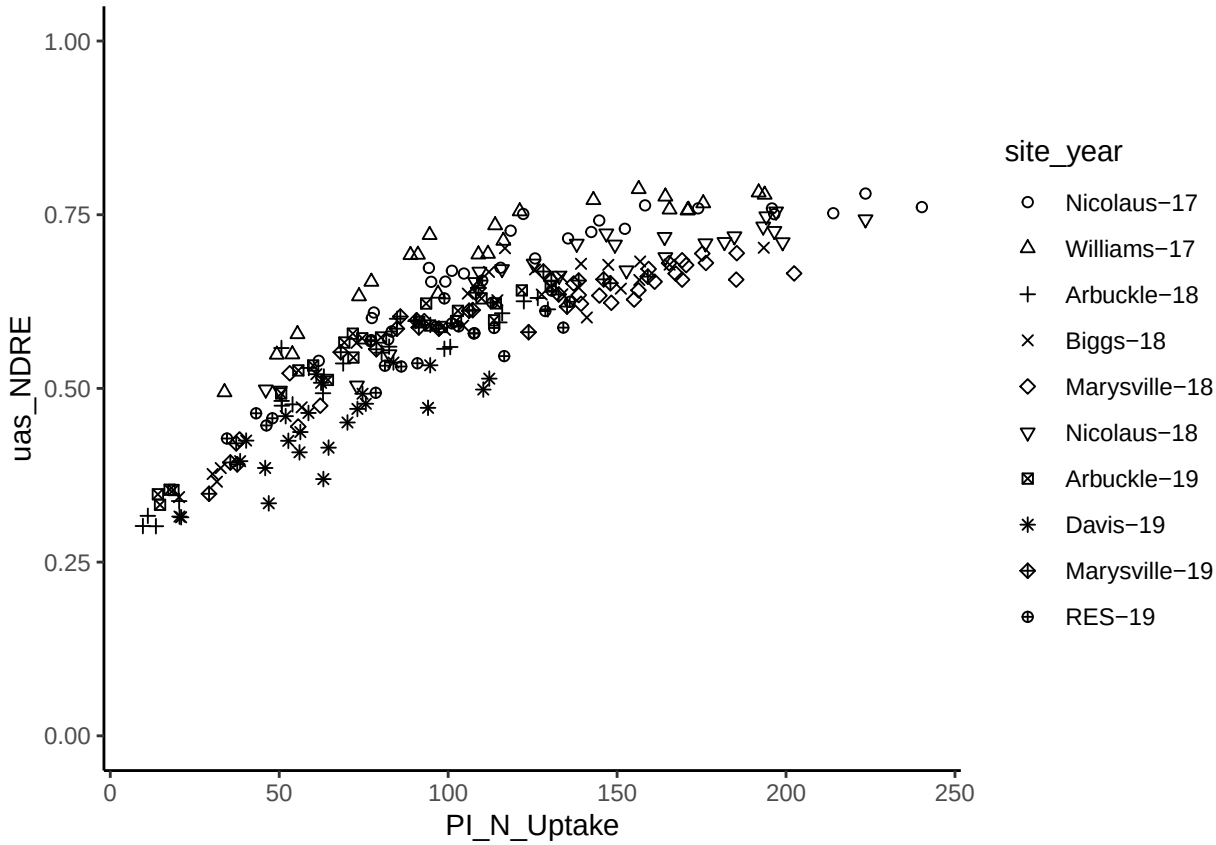

```
boxplot(drone_data$uas_NDVI)
```



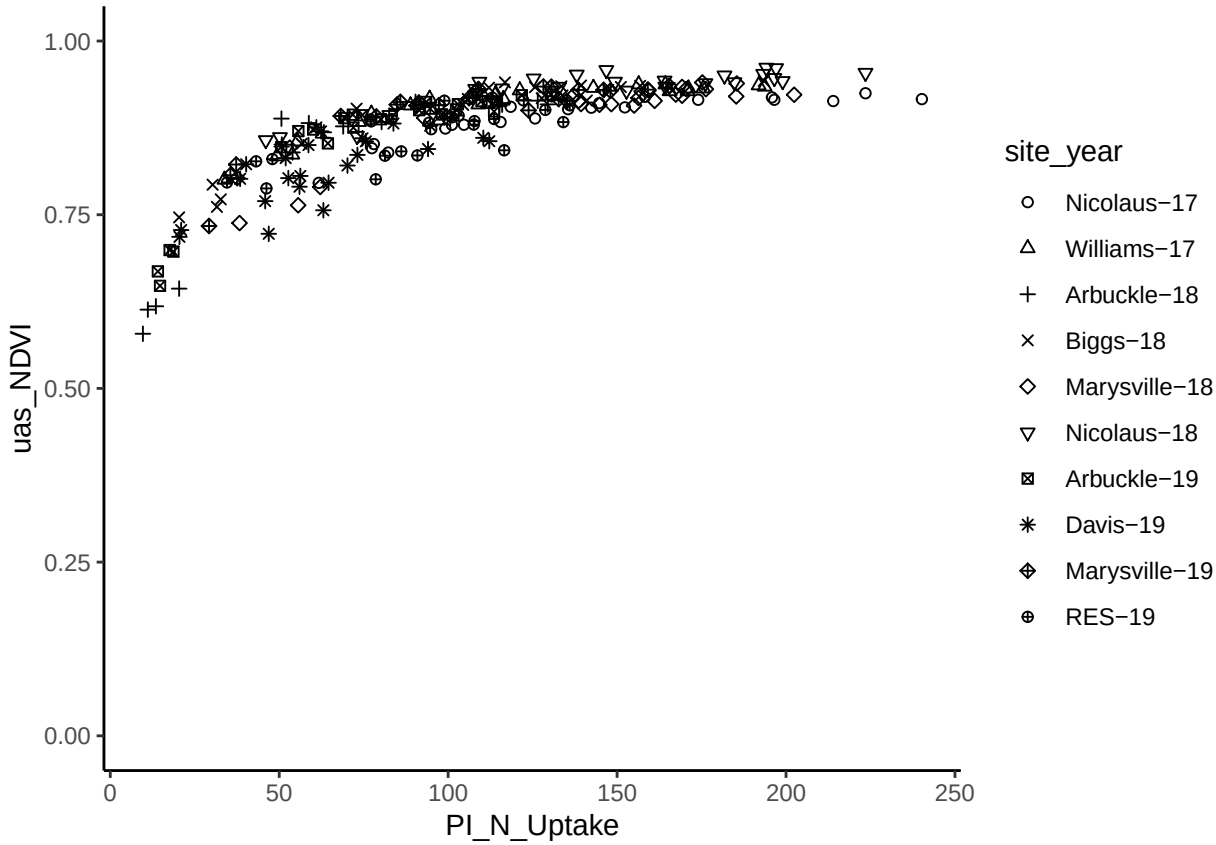

```
hist(drone_data$uas_NDVI)
```




```
ggplot(data = drone_data, aes(x = PI_N_Uptake , y = uas_NDRE )) +
  geom_point(mapping = aes(x = PI_N_Uptake , y = uas_NDRE , shape = site_year) , data = drone_data) +
  theme_classic() +
  scale_shape_manual(values = seq(4:20)) +
  coord_cartesian(ylim = c(0,1))
```




```
ggplot(data = drone_data, aes(x = PI_N_Uptake , y = uas_NDVI )) +
  geom_point(mapping = aes(x = PI_N_Uptake , y = uas_NDVI , shape = site_year) , data = drone_data) +
  theme_classic() +
  scale_shape_manual(values = seq(4:20)) +
  coord_cartesian(ylim = c(0,1))
```




```
drone_data$uas_NDRE <- round(drone_data$uas_NDRE , digits = 2) #rounds to the precision of the instrume
drone_data$uas_NDVI <- round(drone_data$uas_NDVI , digits = 2) #rounds to the precision of the instrume
```

### Calculating UAS RI

```
#gets the max NDRE value for each site

max_drone_data <- drone_data %>%
  filter(N_level_kgha_f %in% c(225, 235)) %>%
  select(site_year , uas_NDVI , uas_NDRE) %>%
  group_by(site_year) %>%
  summarise_all(.funs = mean) %>%
  ungroup()

max_drone_data$uas_NDVI <- round(max_drone_data$uas_NDVI , digits = 2)
max_drone_data$uas_NDRE <- round(max_drone_data$uas_NDRE , digits = 2)

nic17 <- subset(max_drone_data, site_year == "Nicolaus-17")
nic17maxuas_NDRE <- as.numeric(nic17$uas_NDRE)
nic17maxuas_NDRE
```

```
## [1] 0.76
```

```
nic17maxuas_NDVI <- as.numeric(nic17$uas_NDVI)
nic17maxuas_NDVI
```

```
## [1] 0.92
```

```r
wil17 <- subset(max_drone_data, site_year == "Williams-17")
wil17maxuas_NDRE <- as.numeric(wil17$uas_NDRE)
wil17maxuas_NDRE
```

```
## [1] 0.78
```

```r
wil17maxuas_NDVI <- as.numeric(wil17$uas_NDVI)
wil17maxuas_NDVI
```

```
## [1] 0.94
```

```r
arb18 <- subset(max_drone_data, site_year == "Arbuckle-18")
arb18maxuas_NDRE <- as.numeric(arb18$uas_NDRE)
arb18maxuas_NDRE
```

```
## [1] 0.62
```

```r
arb18maxuas_NDVI <- as.numeric(arb18$uas_NDVI)
arb18maxuas_NDVI
```

```
## [1] 0.91
```

```r
biggs18 <- subset(max_drone_data, site_year == "Biggs-18")
biggs18maxuas_NDRE <- as.numeric(biggs18$uas_NDRE)
biggs18maxuas_NDRE
```

```
## [1] 0.68
```

```r
biggs18maxuas_NDVI <- as.numeric(biggs18$uas_NDVI)
biggs18maxuas_NDVI
```

```
## [1] 0.93
```

```r
mry18 <- subset(max_drone_data, site_year == "Marysville-18")
mry18maxuas_NDRE <- as.numeric(mry18$uas_NDRE)
mry18maxuas_NDRE
```

```
## [1] 0.68
```

```r
mry18maxuas_NDVI <- as.numeric(mry18$uas_NDVI)
mry18maxuas_NDVI
```

```
## [1] 0.93
```

```r
nic18 <- subset(max_drone_data, site_year == "Nicolaus-18")
nic18maxuas_NDRE <- as.numeric(nic18$uas_NDRE)
nic18maxuas_NDRE
```

```
## [1] 0.74
```

```r
nic18maxuas_NDVI <- as.numeric(nic18$uas_NDVI)
nic18maxuas_NDVI
```

```
## [1] 0.96
```

```r
arb19 <- subset(max_drone_data, site_year == "Arbuckle-19")
arb19maxuas_NDRE <- as.numeric(arb19$uas_NDRE)
arb19maxuas_NDRE
```

```
## [1] 0.63
```

```
arb19maxuas_NDVI <- as.numeric(arb19$uas_NDVI)
arb19maxuas_NDVI
```

```
## [1] 0.91
```

```
davis19 <- subset(max_drone_data, site_year == "Davis-19")
davis19maxuas_NDRE <- as.numeric(davis19$uas_NDRE)
davis19maxuas_NDRE
```

```
## [1] 0.5
```

```
davis19maxuas_NDVI <- as.numeric(davis19$uas_NDVI)
davis19maxuas_NDVI
```

```
## [1] 0.86
```

```
mry19 <- subset(max_drone_data, site_year == "Marysville-19")
mry19maxuas_NDRE <- as.numeric(mry19$uas_NDRE)
mry19maxuas_NDRE
```

```
## [1] 0.66
```

```
mry19maxuas_NDVI <- as.numeric(mry19$uas_NDVI)
mry19maxuas_NDVI
```

```
## [1] 0.93
```

```
res19 <- subset(max_drone_data, site_year == "RES-19")
res19maxuas_NDRE <- as.numeric(res19$uas_NDRE)
res19maxuas_NDRE
```

```
## [1] 0.6
```

```
res19maxuas_NDVI <- as.numeric(res19$uas_NDVI)
res19maxuas_NDVI
```

```
## [1] 0.88
```

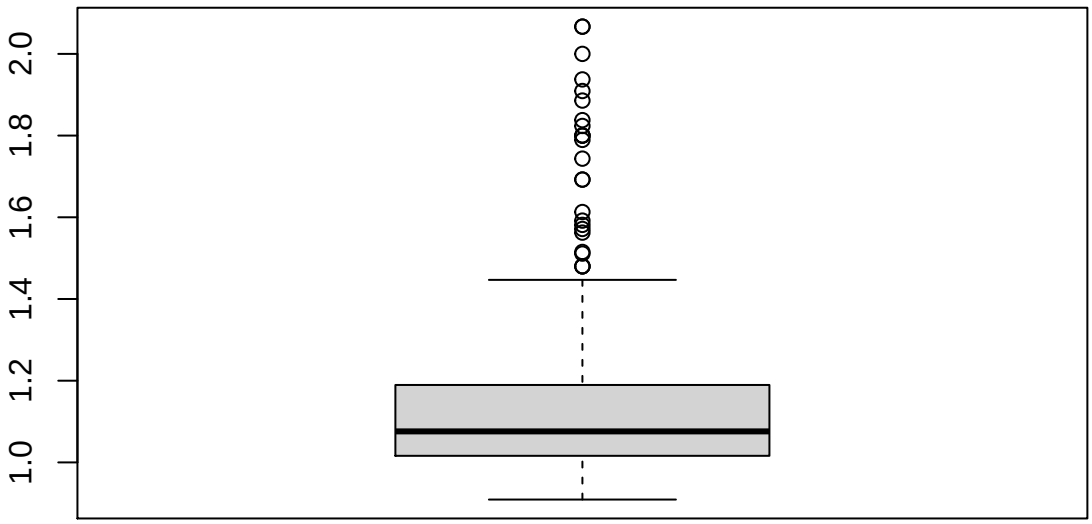
```
drone_data <- drone_data %>%
  mutate(max_uas_NDRE = case_when(
                          site_year == "Nicolaus-17" ~ nic17maxuas_NDRE ,
                          site_year == "Williams-17" ~ wil17maxuas_NDRE ,
                          site_year == "Arbuckle-18" ~ arb18maxuas_NDRE ,
                          site_year == "Biggs-18" ~ biggs18maxuas_NDRE ,
                          site_year == "Marysville-18" ~ mry18maxuas_NDRE ,
                          site_year == "Nicolaus-18" ~ nic18maxuas_NDRE,
                          site_year == "Arbuckle-19" ~ arb19maxuas_NDRE ,
                          site_year == "Davis-19" ~ davis19maxuas_NDRE ,
                          site_year == "Marysville-19" ~ mry19maxuas_NDRE ,
                          site_year == "RES-19" ~ res19maxuas_NDRE) #assign the max NDRE value for each
  )

drone_data <- drone_data %>%
  mutate(uas_NDRE_Response_Index = case_when(
                          site_year == "Nicolaus-17" ~ nic17maxuas_NDRE / uas_NDRE,
                          site_year == "Williams-17" ~ wil17maxuas_NDRE / uas_NDRE,
                          site_year == "Arbuckle-18" ~ arb18maxuas_NDRE / uas_NDRE  ,
                          site_year == "Biggs-18" ~ biggs18maxuas_NDRE / uas_NDRE ,
                          site_year == "Marysville-18" ~ mry18maxuas_NDRE / uas_NDRE ,
```

```
                          site_year == "Nicolaus-18" ~ nic18maxuas_NDRE / uas_NDRE,
                          site_year == "Arbuckle-19" ~ arb19maxuas_NDRE / uas_NDRE,
                          site_year == "Davis-19" ~ davis19maxuas_NDRE / uas_NDRE,
                          site_year == "Marysville-19" ~ mry19maxuas_NDRE / uas_NDRE,
                          site_year == "RES-19" ~ res19maxuas_NDRE / uas_NDRE
                        )) #calculates uas_NDRE response index

drone_data <- drone_data %>%
  mutate(max_uas_NDVI = case_when(
                          site_year == "Nicolaus-17" ~ nic17maxuas_NDVI ,
                          site_year == "Williams-17" ~ wil17maxuas_NDVI ,
                          site_year == "Arbuckle-18" ~ arb18maxuas_NDVI ,
                          site_year == "Biggs-18" ~ biggs18maxuas_NDVI ,
                          site_year == "Marysville-18" ~ mry18maxuas_NDVI ,
                          site_year == "Nicolaus-18" ~ nic18maxuas_NDVI,
                          site_year == "Arbuckle-19" ~ arb19maxuas_NDVI ,
                          site_year == "Davis-19" ~ davis19maxuas_NDVI ,
                          site_year == "Marysville-19" ~ mry19maxuas_NDVI ,
                          site_year == "RES-19" ~ res19maxuas_NDVI) #assign max ndvi value for each sit
  )

drone_data <- drone_data %>%
  mutate(uas_NDVI_Response_Index = case_when(
                          site_year == "Nicolaus-17" ~ nic17maxuas_NDVI / uas_NDVI,
                          site_year == "Williams-17" ~ wil17maxuas_NDVI / uas_NDVI,
                          site_year == "Arbuckle-18" ~ arb18maxuas_NDVI / uas_NDVI  ,
                          site_year == "Biggs-18" ~ biggs18maxuas_NDVI / uas_NDVI ,
                          site_year == "Marysville-18" ~ mry18maxuas_NDVI / uas_NDVI ,
                          site_year == "Nicolaus-18" ~ nic18maxuas_NDVI / uas_NDVI,
                          site_year == "Arbuckle-19" ~ arb19maxuas_NDVI / uas_NDVI,
                          site_year == "Davis-19" ~ davis19maxuas_NDVI / uas_NDVI,
                          site_year == "Marysville-19" ~ mry19maxuas_NDVI / uas_NDVI,
                          site_year == "RES-19" ~ res19maxuas_NDVI / uas_NDVI
                        )) #calculates uas_NDVI response index

str(drone_data , give.attr =  F)
```

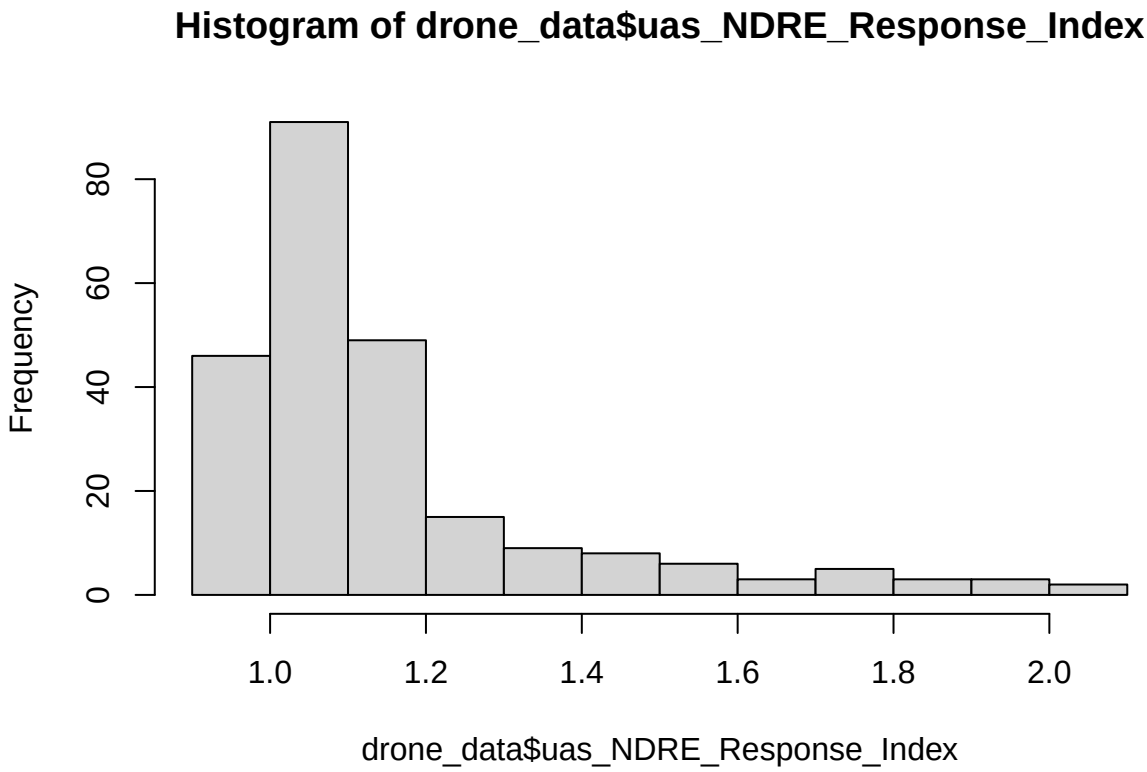
```
## tibble[,29] [240 x 29] (S3: tbl_df/tbl/data.frame)
##  $ site_year              : Factor w/ 10 levels "Nicolaus-17",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year                   : Factor w/ 3 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ exp_plot_number        : Factor w/ 27 levels "101","102","103",..: 1 2 3 4 6 7 9 10 11 12 ...
##  $ Block                  : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 2 2 2 2 ...
##  $ MainPlot               : Factor w/ 7 levels "1","2","3","4",..: 1 2 3 4 6 7 2 3 4 5 ...
##  $ N_level                : Factor w/ 11 levels "0","45","75",..: 11 1 6 8 2 3 3 6 1 8 ...
##  $ N_level_kgha           : num [1:240] 225 0 125 175 45 75 75 125 0 175 ...
##  $ N_level_kgha_f         : Factor w/ 11 levels "0","45","75",..: 10 1 5 8 2 3 3 5 1 8 ...
##  $ biomass_plus_bag_g     : num [1:240] 361 264 318 360 285 ...
##  $ ring_size              : num [1:240] 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
##  $ paper_bag_g            : num [1:240] 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 45.5 ...
##  $ num_of_paper_bags      : num [1:240] 1 1 1 1 1 1 1 1 1 1 ...
##  $ sample_weight_mg       : num [1:240] 3.17 3.42 3.09 3.07 3.2 ...
##  $ sample_N_ug            : num [1:240] 79.4 61.1 80.7 95.9 63.6 ...
##  $ bluemean               : num [1:240] NA NA NA NA NA NA NA NA NA NA ...
##  $ greenmean              : num [1:240] 0.0467 0.0581 0.0498 0.0488 0.0608 0.0589 0.0537 0.0488 0.059
```

```
##  $ redmean                : num [1:240] 0.023 0.0286 0.0243 0.0237 0.0292 0.0281 0.0263 0.0237 0.029
##  $ edgemean               : num [1:240] 0.0738 0.0867 0.0768 0.0764 0.0903 0.0877 0.0812 0.0754 0.08
##  $ nirmean                : num [1:240] 0.55 0.357 0.482 0.537 0.431 ...
##  $ biomass_dry_wt         : num [1:240] 316 219 273 315 239 ...
##  $ aboveground_biomass    : num [1:240] 6316 4376 5456 6296 4782 ...
##  $ n_content              : num [1:240] 25.1 17.8 26.1 31.2 19.9 ...
##  $ PI_N_Uptake            : num [1:240] 158 78 142 197 95 ...
##  $ uas_NDRE               : num [1:240] 0.76 0.61 0.73 0.75 0.65 0.67 0.67 0.73 0.6 0.76 ...
##  $ uas_NDVI               : num [1:240] 0.92 0.85 0.9 0.92 0.87 0.88 0.88 0.91 0.85 0.92 ...
##  $ max_uas_NDRE           : num [1:240] 0.76 0.76 0.76 0.76 0.76 0.76 0.76 0.76 0.76 0.76 ...
##  $ uas_NDRE_Response_Index: num [1:240] 1 1.25 1.04 1.01 1.17 ...
##  $ max_uas_NDVI           : num [1:240] 0.92 0.92 0.92 0.92 0.92 0.92 0.92 0.92 0.92 0.92 ...
##  $ uas_NDVI_Response_Index: num [1:240] 1 1.08 1.02 1 1.06 ...
```

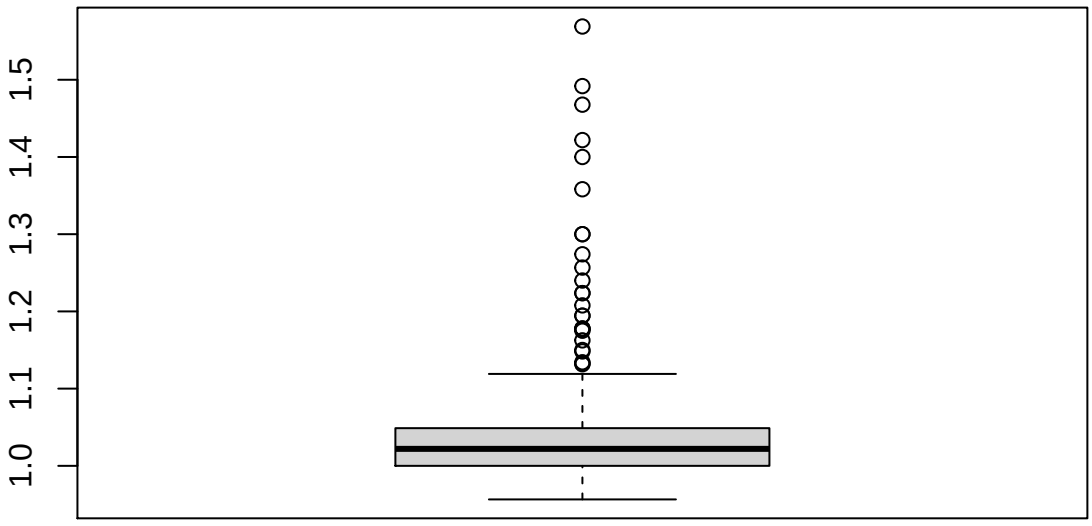
```
boxplot(drone_data$uas_NDRE_Response_Index)
```



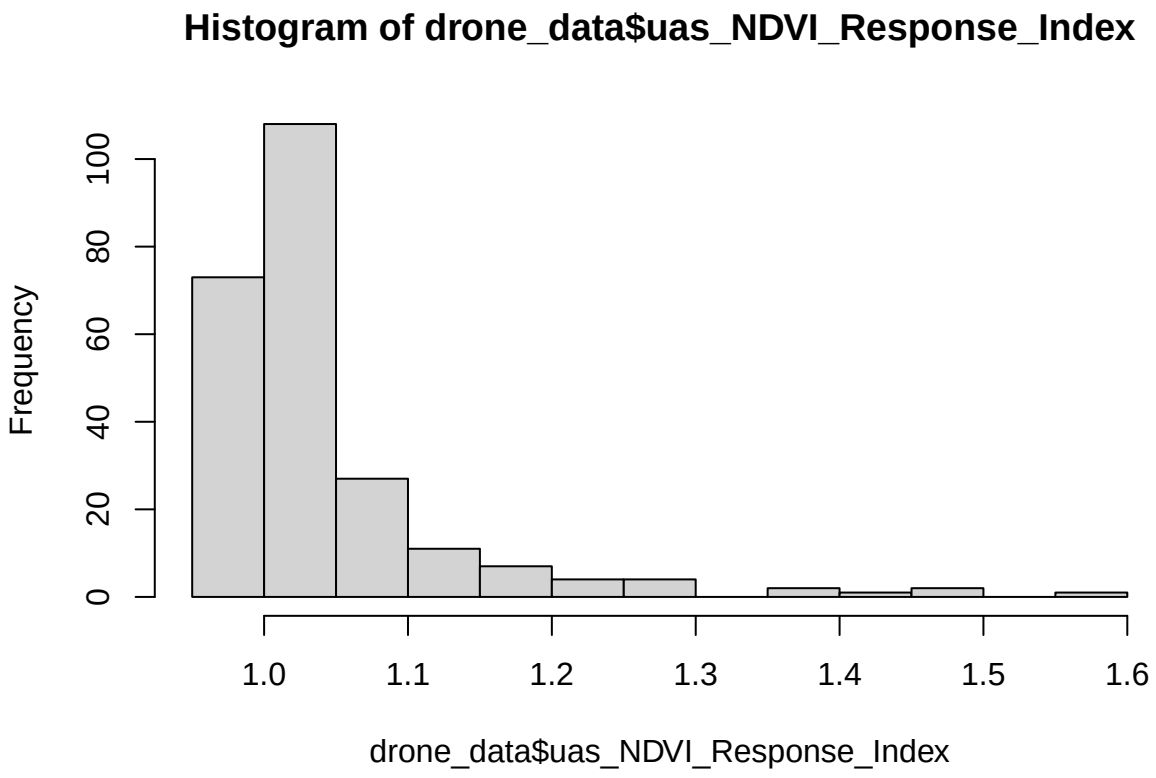

```
hist(drone_data$uas_NDRE_Response_Index)
```




```
boxplot(drone_data$uas_NDVI_Response_Index)
```




```
hist(drone_data$uas_NDVI_Response_Index)
```




```
drone_data$uas_NDVI_Response_Index[drone_data$uas_NDVI_Response_Index < 1] <- 1 #converts values less t
drone_data$uas_NDRE_Response_Index[drone_data$uas_NDRE_Response_Index < 1] <- 1 #converts values less t
```

```
drone_data <- dplyr::select(drone_data ,
                      site_year,
                      year,
                      exp_plot_number,
                      Block,
                      MainPlot,
                      N_level_kgha,
                      N_level_kgha_f,
                      PI_N_Uptake,
                      uas_NDRE,
                      uas_NDVI,
                      uas_NDRE_Response_Index,
                      uas_NDVI_Response_Index
                      )#selects the relevant columns

sUAS_yield_data <- yield_data

sUAS_yield_data$site_year <- factor(sUAS_yield_data$site_year , levels = c("Nicolaus-17" , "Williams-17

str(drone_data , give.attr = F)
```

```
## tibble[,12] [240 x 12] (S3: tbl_df/tbl/data.frame)
##  $ site_year              : Factor w/ 10 levels "Nicolaus-17",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year                   : Factor w/ 3 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ exp_plot_number        : Factor w/ 27 levels "101","102","103",..: 1 2 3 4 6 7 9 10 11 12 ...
##  $ Block                  : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 2 2 2 2 ...
##  $ MainPlot               : Factor w/ 7 levels "1","2","3","4",..: 1 2 3 4 6 7 2 3 4 5 ...
```

```
##  $ N_level_kgha           : num [1:240] 225 0 125 175 45 75 75 125 0 175 ...
##  $ N_level_kgha_f         : Factor w/ 11 levels "0","45","75",..: 10 1 5 8 2 3 3 5 1 8 ...
##  $ PI_N_Uptake            : num [1:240] 158 78 142 197 95 ...
##  $ uas_NDRE               : num [1:240] 0.76 0.61 0.73 0.75 0.65 0.67 0.67 0.73 0.6 0.76 ...
##  $ uas_NDVI               : num [1:240] 0.92 0.85 0.9 0.92 0.87 0.88 0.88 0.91 0.85 0.92 ...
##  $ uas_NDRE_Response_Index: num [1:240] 1 1.25 1.04 1.01 1.17 ...
##  $ uas_NDVI_Response_Index: num [1:240] 1 1.08 1.02 1 1.06 ...
```

```
str(sUAS_yield_data , give.attr = F)
```

```
## tibble[,11] [528 x 11] (S3: tbl_df/tbl/data.frame)
##  $ site_year      : Factor w/ 10 levels "Nicolaus-17",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year           : Factor w/ 3 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Block          : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 2 2 2 2 ...
##  $ MainPlot       : Factor w/ 7 levels "1","2","3","4",..: 1 2 3 4 6 7 2 3 4 5 ...
##  $ SubPlot        : Factor w/ 3 levels "1","2","3": 2 3 1 1 1 2 1 1 2 3 ...
##  $ exp_plot_number: Factor w/ 27 levels "101","102","103",..: 1 2 3 4 6 7 9 10 11 12 ...
##  $ N_level_kgha   : num [1:528] 225 0 125 175 45 75 75 125 0 175 ...
##  $ N_level_kgha_f : Factor w/ 11 levels "0","45","75",..: 10 1 5 8 2 3 3 5 1 8 ...
##  $ TopDress_kgha  : num [1:528] 0 0 0 0 0 0 0 0 0 0 ...
##  $ TopDress_kgha_f: Factor w/ 4 levels "0","25","34",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ GrainYield_Mgha: num [1:528] 11.4 11 12 11.9 11.5 ...
```

```
sUAS_data <- full_join( drone_data , sUAS_yield_data)
```

```
## Joining, by = c("site_year", "year", "exp_plot_number", "Block", "MainPlot", "N_level_kgha", "N_level
```

```
sUAS_data <- dplyr::select(sUAS_data ,
                      site_year,
                      year,
                      exp_plot_number,
                      Block,
                      MainPlot,
                      SubPlot,
                      N_level_kgha,
                      N_level_kgha_f,
                      TopDress_kgha,
                      TopDress_kgha_f,
                      PI_N_Uptake,
                      uas_NDRE,
                      uas_NDVI,
                      GrainYield_Mgha,
                      uas_NDRE_Response_Index,
                      uas_NDVI_Response_Index
                      )#selects the relevant columns


str(sUAS_data , give.attr = F)
```

```
## tibble[,16] [528 x 16] (S3: tbl_df/tbl/data.frame)
##  $ site_year              : Factor w/ 10 levels "Nicolaus-17",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year                   : Factor w/ 3 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ exp_plot_number        : Factor w/ 27 levels "101","102","103",..: 1 1 1 2 2 2 3 3 3 4 ...
##  $ Block                  : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...
##  $ MainPlot               : Factor w/ 7 levels "1","2","3","4",..: 1 1 1 2 2 2 3 3 3 4 ...
##  $ SubPlot                : Factor w/ 3 levels "1","2","3": 2 1 3 3 2 1 1 3 2 1 ...
```

```
##  $ N_level_kgha           : num [1:528] 225 225 225 0 0 0 125 125 125 175 ...
##  $ N_level_kgha_f         : Factor w/ 11 levels "0","45","75",..: 10 10 10 1 1 1 5 5 5 8 ...
##  $ TopDress_kgha          : num [1:528] 0 25 50 0 25 50 0 25 50 0 ...
##  $ TopDress_kgha_f        : Factor w/ 4 levels "0","25","34",..: 1 2 4 1 2 4 1 2 4 1 ...
##  $ PI_N_Uptake            : num [1:528] 158 158 158 78 78 ...
##  $ uas_NDRE               : num [1:528] 0.76 0.76 0.76 0.61 0.61 0.61 0.73 0.73 0.73 0.75 ...
##  $ uas_NDVI               : num [1:528] 0.92 0.92 0.92 0.85 0.85 0.85 0.9 0.9 0.9 0.92 ...
##  $ GrainYield_Mgha        : num [1:528] 11.4 12.2 12.1 11 10.9 ...
##  $ uas_NDRE_Response_Index: num [1:528] 1 1 1 1.25 1.25 ...
##  $ uas_NDVI_Response_Index: num [1:528] 1 1 1 1.08 1.08 ...
```

##Combining the data

```
gs_data <- gs_data %>%
  mutate(gs_NDVI_SI = 1 / gs_NDVI_Response_Index)

sUAS_data <- sUAS_data %>%
  mutate(uas_NDVI_SI = 1 / uas_NDVI_Response_Index)

sUAS_data <- sUAS_data %>%
  mutate(uas_NDRE_SI = 1 / uas_NDRE_Response_Index)

gs_data <- gs_data %>%
  dplyr::select(site_year ,
                year,
                exp_plot_number,
                Block,
                MainPlot,
                SubPlot,
                N_level_kgha,
                N_level_kgha_f,
                TopDress_kgha,
                TopDress_kgha_f,
                PI_N_Uptake,
                NDVI,
                GrainYield_Mgha,
                gs_NDVI_Response_Index,
                gs_NDVI_SI
        )

sUAS_data <- sUAS_data %>%
  dplyr::select(site_year ,
                year,
                exp_plot_number,
                Block,
                MainPlot,
                SubPlot,
                N_level_kgha,
                N_level_kgha_f,
                TopDress_kgha,
                TopDress_kgha_f,
                uas_NDRE,
                uas_NDVI,
                uas_NDVI_Response_Index,
                uas_NDVI_SI,
```

```
                uas_NDRE_Response_Index,
                uas_NDRE_SI
        )

paper3_data <- full_join(gs_data , sUAS_data)
```

```
## Joining, by = c("site_year", "year", "exp_plot_number", "Block", "MainPlot", "SubPlot", "N_level_kgha
```

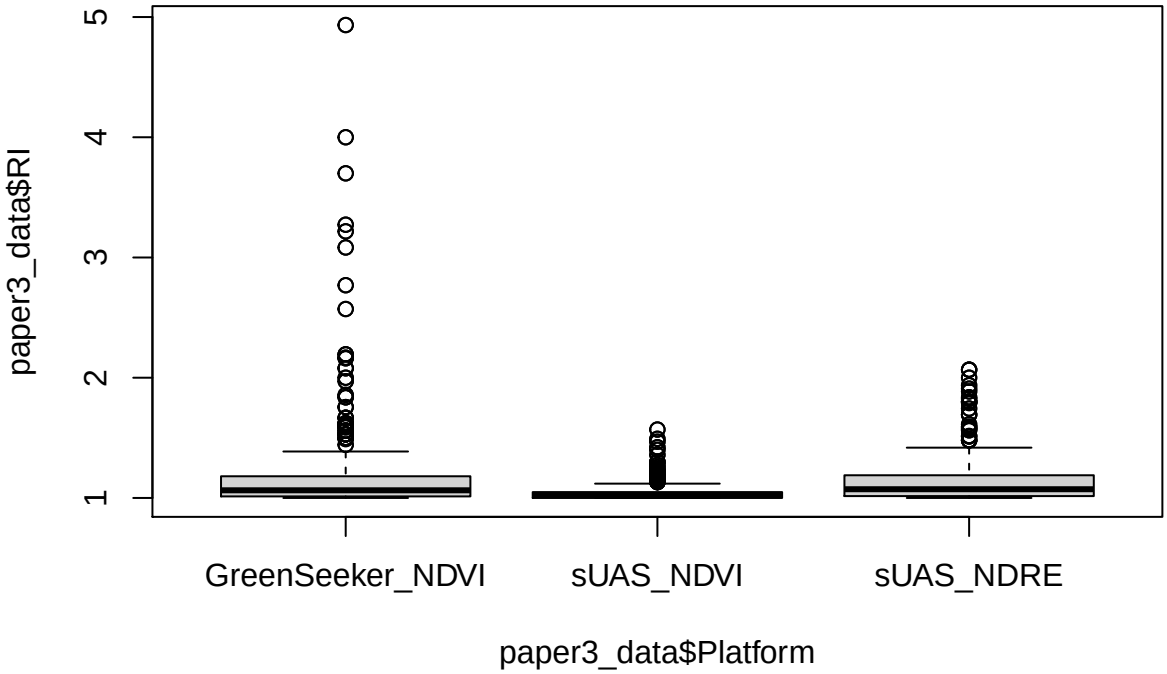
```
paper3_gsdata <- paper3_data %>%
  dplyr::select(site_year ,
                year ,
                exp_plot_number ,
                Block ,
                MainPlot ,
                SubPlot ,
                N_level_kgha ,
                N_level_kgha_f ,
                TopDress_kgha ,
                TopDress_kgha_f ,
                NDVI,
                gs_NDVI_Response_Index,
                gs_NDVI_SI ,
                GrainYield_Mgha,
                PI_N_Uptake) %>%
  mutate(Platform = "GreenSeeker_NDVI" ,
         SI_sq = (gs_NDVI_SI * gs_NDVI_SI)) %>%
  rename(Index = NDVI ,
         SI = gs_NDVI_SI,
         RI = gs_NDVI_Response_Index)

paper3_uas_ndvi_data <- paper3_data %>%
  dplyr::select(site_year ,
                year ,
                exp_plot_number ,
                Block ,
                MainPlot ,
                SubPlot ,
                N_level_kgha ,
                N_level_kgha_f ,
                TopDress_kgha ,
                TopDress_kgha_f ,
                uas_NDVI,
                uas_NDVI_Response_Index,
                uas_NDVI_SI ,
                GrainYield_Mgha,
                PI_N_Uptake) %>%
  mutate(Platform = "sUAS_NDVI" ,
         SI_sq = (uas_NDVI_SI * uas_NDVI_SI)) %>%
  rename(Index = uas_NDVI,
         SI = uas_NDVI_SI,
         RI = uas_NDVI_Response_Index)

paper3_uas_ndre_data <- paper3_data %>%
  dplyr::select(site_year ,
```

```
                year ,
                exp_plot_number ,
                Block ,
                MainPlot ,
                SubPlot ,
                N_level_kgha ,
                N_level_kgha_f ,
                TopDress_kgha ,
                TopDress_kgha_f ,
                uas_NDRE ,
                uas_NDRE_Response_Index ,
                uas_NDRE_SI ,
                GrainYield_Mgha,
                PI_N_Uptake) %>%
  mutate(Platform = "sUAS_NDRE" ,
         SI_sq = (uas_NDRE_SI * uas_NDRE_SI)) %>%
  rename(Index = uas_NDRE ,
         SI = uas_NDRE_SI,
         RI = uas_NDRE_Response_Index)

paper3_data <- rbind(paper3_gsdata ,
                     paper3_uas_ndvi_data ,
                     paper3_uas_ndre_data)

paper3_data <- paper3_data %>%
  dplyr::select(site_year ,
                year ,
                Platform,
                exp_plot_number ,
                Block ,
                MainPlot ,
                SubPlot ,
                N_level_kgha ,
                N_level_kgha_f ,
                TopDress_kgha ,
                TopDress_kgha_f ,
                Index,
                RI,
                SI ,
                SI_sq ,
                GrainYield_Mgha,
                PI_N_Uptake)

paper3_data$Platform <- as.factor(paper3_data$Platform)

str(paper3_data , give.attr = F)
```

```
## rowwise_df[,17] [1,584 x 17] (S3: rowwise_df/tbl_df/tbl/data.frame)
##  $ site_year      : Factor w/ 10 levels "Nicolaus-17",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year           : Factor w/ 3 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Platform       : Factor w/ 3 levels "GreenSeeker_NDVI",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ exp_plot_number: Factor w/ 27 levels "101","102","103",..: 1 1 1 2 2 2 3 3 3 4 ...
##  $ Block          : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...
##  $ MainPlot       : Factor w/ 7 levels "1","2","3","4",..: 1 1 1 2 2 2 3 3 3 4 ...
```

```
##  $ SubPlot        : Factor w/ 3 levels "1","2","3": 2 1 3 3 2 1 1 3 2 1 ...
##  $ N_level_kgha   : num [1:1584] 225 225 225 0 0 0 125 125 125 175 ...
##  $ N_level_kgha_f : Factor w/ 11 levels "0","45","75",..: 10 10 10 1 1 1 5 5 5 8 ...
##  $ TopDress_kgha  : num [1:1584] 0 25 50 0 25 50 0 25 50 0 ...
##  $ TopDress_kgha_f: Factor w/ 4 levels "0","25","34",..: 1 2 4 1 2 4 1 2 4 1 ...
##  $ Index          : num [1:1584] 0.78 0.78 0.78 0.49 0.49 0.49 0.68 0.68 0.68 0.79 ...
##  $ RI             : num [1:1584] 1 1 1 1.59 1.59 ...
##  $ SI             : num [1:1584] 1 1 1 0.628 0.628 ...
##  $ SI_sq          : num [1:1584] 1 1 1 0.395 0.395 ...
##  $ GrainYield_Mgha: num [1:1584] 11.4 12.2 12.1 11 10.9 ...
##  $ PI_N_Uptake    : num [1:1584] 158 158 158 78 78 ...
```

```
paper3_data <- tibble::rowid_to_column(paper3_data, "ID") #adds a columns with row number.

paper3_data <- paper3_data %>%
  filter(!ID %in% c(193 , 194 , 721 , 722, 1249 , 1250)) #removes Biggs-18 101 plot bc tractor ran thro

paper3_data$Platform = factor(paper3_data$Platform, levels=c("GreenSeeker_NDVI" , "sUAS_NDVI" , "sUAS_N

boxplot(paper3_data$RI ~ paper3_data$Platform)
```




```
paper3_data$RI <- round(paper3_data$RI , digits = 3)
paper3_data$SI <- round(paper3_data$SI , digits = 3)
paper3_data$SI_sq <- round(paper3_data$SI_sq , digits = 3)
```

##Data for Table 2

```
table_2_data_1 <- paper3_data %>%
  select(site_year , Platform , Index , RI) %>%
  group_by(site_year , Platform) %>%
  summarise(Index_min = min(Index),
            Index_max = max(Index),
            RI_min = min(RI),
            RI_max = max(RI)) %>%
```

```
  ungroup()
```

```
## `summarise()` has grouped output by 'site_year'. You can override using the `.groups` argument.
```

```
table_2_data_1$Index_min <- round(table_2_data_1$Index_min , digits = 2)
table_2_data_1$Index_max <- round(table_2_data_1$Index_max , digits = 2)
table_2_data_1$RI_min <- round(table_2_data_1$RI_min , digits = 2)
table_2_data_1$RI_max <- round(table_2_data_1$RI_max , digits = 2)

table_2_data_2 <- paper3_data %>%
  select(site_year , Platform , PI_N_Uptake) %>%
  group_by(site_year , Platform) %>%
  filter(Platform == "GreenSeeker_NDVI") %>%
  summarise(PI_N_Uptake_min = min(PI_N_Uptake),
            PI_N_Uptake_max = max(PI_N_Uptake)) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'site_year'. You can override using the `.groups` argument.
```

```
table_2_data_2$PI_N_Uptake_min <- round(table_2_data_2$PI_N_Uptake_min , digits = 0)
table_2_data_2$PI_N_Uptake_max <- round(table_2_data_2$PI_N_Uptake_max , digits = 0)
```

#VI MODELS

##GS NDVI

```
gsndvi_df <- paper3_data %>%
  filter(Platform == "GreenSeeker_NDVI") #filter just gs ndvi values


PI_N_Uptake2 <- gsndvi_df$PI_N_Uptake^2
gs_ndvi_qm <- lm(Index ~ PI_N_Uptake + PI_N_Uptake2 , gsndvi_df) #quadratic model
summary(gs_ndvi_qm) #r2 = 0.80
```

```
##
## Call:
## lm(formula = Index ~ PI_N_Uptake + PI_N_Uptake2, data = gsndvi_df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.182258 -0.040094  0.006751  0.038822  0.207385
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.360e-01  1.106e-02   21.34   <2e-16 ***
## PI_N_Uptake   6.169e-03  2.069e-04   29.82   <2e-16 ***
## PI_N_Uptake2 -1.787e-05  8.829e-07  -20.24   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05914 on 523 degrees of freedom
## Multiple R-squared:  0.7974, Adjusted R-squared:  0.7967
## F-statistic:  1029 on 2 and 523 DF,  p-value: < 2.2e-16
```

```
Anova(gs_ndvi_qm , type = 3)
```

```
## Anova Table (Type III tests)
```

```
##
## Response: Index
##              Sum Sq  Df F value    Pr(>F)
## (Intercept)  1.5924   1  455.30 < 2.2e-16 ***
## PI_N_Uptake  3.1102   1  889.26 < 2.2e-16 ***
## PI_N_Uptake2 1.4333   1  409.81 < 2.2e-16 ***
## Residuals    1.8292 523
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
gs_ndvi_fit_qm <- fitted(gs_ndvi_qm) #gets fitted values from quad model
gs_ndvi_qm_df <- data.frame(gsndvi_df$PI_N_Uptake , gs_ndvi_fit_qm) #creates dataframe
gs_ndvi_qm_df <- gs_ndvi_qm_df %>%
  mutate(Platform = "GreenSeeker_NDVI")

gs_ndvi_qm_r2 <- round(summary(gs_ndvi_qm)$adj.r.squared , digits = 2)
gs_ndvi_qm_a <- as.numeric(as.character(coef(gs_ndvi_qm)[3]))
gs_ndvi_qm_b <- as.numeric(as.character(coef(gs_ndvi_qm)[2]))
gs_ndvi_qm_c <- as.numeric(as.character(coef(gs_ndvi_qm)[1]))
gs_ndvi_qm_sym_x <- (-gs_ndvi_qm_b) / (2*gs_ndvi_qm_a)
gs_ndvi_qm_sym_y <- gs_ndvi_qm_a*(gs_ndvi_qm_sym_x^2) + gs_ndvi_qm_b*gs_ndvi_qm_sym_x + gs_ndvi_qm_c

gs_ndvi_qm_eqn <- paste("y == -1.8e-05*x^2 + 0.006* x + 0.24")

gs_ndvi_qm_r2
```

```
## [1] 0.8
```

```
gs_ndvi_qm_sym_y
```

```
## [1] 0.7682878
```

```
gs_ndvi_qm_sym_x
```

```
## [1] 172.5754
```

##sUAS NDVI

```
sUASndvi_df <- paper3_data %>%
  filter(Platform == "sUAS_NDVI") #filter just sUAS ndvi values


PI_N_Uptake2 <- sUASndvi_df$PI_N_Uptake^2
sUAS_ndvi_qm <- lm(Index ~ PI_N_Uptake + PI_N_Uptake2 , sUASndvi_df) #quadratic model
summary(sUAS_ndvi_qm) #r2 = 0.78
```

```
##
## Call:
## lm(formula = Index ~ PI_N_Uptake + PI_N_Uptake2, data = sUASndvi_df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.120191 -0.015915  0.000394  0.022040  0.080498
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.696e-01  5.382e-03  124.41   <2e-16 ***
```

```
## PI_N_Uptake   3.268e-03  1.007e-04   32.46   <2e-16 ***
## PI_N_Uptake2 -9.978e-06  4.297e-07  -23.22   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02878 on 523 degrees of freedom
## Multiple R-squared:  0.8014, Adjusted R-squared:  0.8006
## F-statistic:  1055 on 2 and 523 DF,  p-value: < 2.2e-16
```

```
sUAS_ndvi_fit_qm <- fitted(sUAS_ndvi_qm) #gets fitted values from quad model
sUAS_ndvi_qm_df <- data.frame(sUASndvi_df$PI_N_Uptake , sUAS_ndvi_fit_qm) #creates dataframe

sUAS_ndvi_qm_df <- sUAS_ndvi_qm_df %>%
  mutate(Platform = "sUAS_NDVI")

#getting the equation for the plot
sUAS_ndvi_qm_r2 <- round(summary(sUAS_ndvi_qm)$adj.r.squared , digits = 2)
sUAS_ndvi_qm_a <- as.numeric(as.character(coef(sUAS_ndvi_qm)[3]))
sUAS_ndvi_qm_b <- as.numeric(as.character(coef(sUAS_ndvi_qm)[2]))
sUAS_ndvi_qm_c <- as.numeric(as.character(coef(sUAS_ndvi_qm)[1]))
sUAS_ndvi_qm_sym_x <- (-sUAS_ndvi_qm_b) / (2*sUAS_ndvi_qm_a)
sUAS_ndvi_qm_sym_y <- sUAS_ndvi_qm_a*(sUAS_ndvi_qm_sym_x^2) + sUAS_ndvi_qm_b*sUAS_ndvi_qm_sym_x + sUAS_

sUAS_ndvi_qm_eqn <- paste("y == -1.0e-05*x^2 + 0.0033* x + 0.67")

sUAS_ndvi_qm_r2
```

```
## [1] 0.8
```

```
sUAS_ndvi_qm_sym_y
```

```
## [1] 0.9371271
```

```
sUAS_ndvi_qm_sym_x
```

```
## [1] 163.7569
```

##sUAS NDRE

```
sUASndre_df <- paper3_data %>%
  filter(Platform == "sUAS_NDRE") #filter just sUAS ndre values

PI_N_Uptake2 <- sUASndre_df$PI_N_Uptake^2
sUAS_ndre_qm <- lm(Index ~ PI_N_Uptake + PI_N_Uptake2 , sUASndre_df) #quadratic model
summary(sUAS_ndre_qm) #r2 = 0.82
```

```
##
## Call:
## lm(formula = Index ~ PI_N_Uptake + PI_N_Uptake2, data = sUASndre_df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.141442 -0.034596 -0.001881  0.036173  0.121878
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.738e-01  9.276e-03   29.52   <2e-16 ***
```

```
## PI_N_Uptake   4.442e-03  1.735e-04   25.60   <2e-16 ***
## PI_N_Uptake2 -1.069e-05  7.406e-07  -14.44   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.04961 on 523 degrees of freedom
## Multiple R-squared:  0.8172, Adjusted R-squared:  0.8165
## F-statistic:  1169 on 2 and 523 DF,  p-value: < 2.2e-16
```

```
sUAS_ndre_qm_mse <- mean(residuals(sUAS_ndre_qm)^2)
sUAS_ndre_qm_mse
```

```
## [1] 0.002446664
```

```
sUAS_ndre_qm_rmse <- sqrt(sUAS_ndre_qm_mse)
sUAS_ndre_qm_rmse
```

```
## [1] 0.04946376
```

```
sUAS_ndre_fit_qm <- fitted(sUAS_ndre_qm) #gets fitted values from linear model
sUAS_ndre_qm_df <- data.frame(sUASndre_df$PI_N_Uptake , sUAS_ndre_fit_qm) #creates dataframe

sUAS_ndre_qm_df <- sUAS_ndre_qm_df %>%
  mutate(Platform = "sUAS_NDRE")


#getting the equation for the plot
sUAS_ndre_qm_r2 <- round(summary(sUAS_ndre_qm)$adj.r.squared , digits = 2)
sUAS_ndre_qm_a <- as.numeric(as.character(coef(sUAS_ndre_qm)[3]))
sUAS_ndre_qm_b <- as.numeric(as.character(coef(sUAS_ndre_qm)[2]))
sUAS_ndre_qm_c <- as.numeric(as.character(coef(sUAS_ndre_qm)[1]))
sUAS_ndre_qm_sym_x <- (-sUAS_ndre_qm_b) / (2*sUAS_ndre_qm_a)
sUAS_ndre_qm_sym_y <- sUAS_ndre_qm_a*(sUAS_ndre_qm_sym_x^2) + sUAS_ndre_qm_b*sUAS_ndre_qm_sym_x + sUAS_

sUAS_ndre_qm_eqn <- paste("y == -1.1e-05*x^2 + 0.004* x + 0.27")

sUAS_ndre_qm_r2
```

```
## [1] 0.82
```

```
sUAS_ndre_qm_sym_y
```

```
## [1] 0.7351191
```

```
sUAS_ndre_qm_sym_x
```

```
## [1] 207.7103
```

#RI MODELS

##GS NDVI-RI

```
paper3_data_no_2018 <- paper3_data %>%
  filter(year != "2018",
         !ID %in% c(419, 420 , 431 , 432 , 947 , 948 ,959 , 960 , 1475 , 1476 , 1487 , 1488))
#need to get the dataframe in the right form to make saturation figure for the paper. Removes 2018 year

greenseeker_ndvi_data <- paper3_data_no_2018 %>%
  filter(Platform == "GreenSeeker_NDVI")
```

```
ctrl <- lmeControl(opt = "optim") #changes control to "optimal" settings

gs_ndvi_model <- lme(GrainYield_Mgha ~ SI * TopDress_kgha + I(SI*SI)  ,
             control = ctrl ,
             random = ~ I(TopDress_kgha) | site_year  ,
             data = greenseeker_ndvi_data)

summary(gs_ndvi_model)
```

```
## Linear mixed-effects model fit by REML
##   Data: greenseeker_ndvi_data
##        AIC      BIC    logLik
##   859.9217 894.0313 -420.9608
##
## Random effects:
##  Formula: ~I(TopDress_kgha) | site_year
##  Structure: General positive-definite, Log-Cholesky parametrization
##                   StdDev    Corr
## (Intercept)      1.0746539 (Intr)
## I(TopDress_kgha) 0.0048649 -0.946
## Residual         0.8137735
##
## Fixed effects:  GrainYield_Mgha ~ SI * TopDress_kgha + I(SI * SI)
##                       Value Std.Error  DF   t-value p-value
## (Intercept)      -0.310935 0.9171844 322 -0.339010  0.7348
## SI               16.478265 2.1839827 322  7.545053  0.0000
## TopDress_kgha     0.083219 0.0134688 322  6.178603  0.0000
## I(SI * SI)       -5.517499 1.4660309 322 -3.763563  0.0002
## SI:TopDress_kgha -0.079362 0.0150575 322 -5.270580  0.0000
##  Correlation:
##                  (Intr) SI     TpDrs_ I(SI*S
## SI               -0.842
## TopDress_kgha    -0.311  0.080
## I(SI * SI)        0.781 -0.982  0.055
## SI:TopDress_kgha  0.242 -0.083 -0.973 -0.056
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -3.63978666 -0.56804670  0.03176909  0.63982598  3.32828753
##
## Number of Observations: 332
## Number of Groups: 6
```

```
summary(gs_ndvi_model)$tTable
```

```
##                       Value  Std.Error  DF    t-value      p-value
## (Intercept)      -0.31093457 0.91718440 322 -0.3390099 7.348233e-01
## SI               16.47826480 2.18398272 322  7.5450527 4.668881e-13
## TopDress_kgha     0.08321852 0.01346883 322  6.1786031 1.951965e-09
## I(SI * SI)       -5.51749911 1.46603094 322 -3.7635625 1.990236e-04
## SI:TopDress_kgha -0.07936173 0.01505749 322 -5.2705803 2.496678e-07
```

```
Anova(gs_ndvi_model , type = 3)
```

```
## Analysis of Deviance Table (Type III tests)
```

```
##
## Response: GrainYield_Mgha
##                     Chisq Df Pr(>Chisq)
## (Intercept)        0.1149  1  0.7346023
## SI                56.9278  1  4.521e-14 ***
## TopDress_kgha     38.1751  1  6.467e-10 ***
## I(SI * SI)        14.1644  1  0.0001675 ***
## SI:TopDress_kgha 27.7790  1  1.360e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

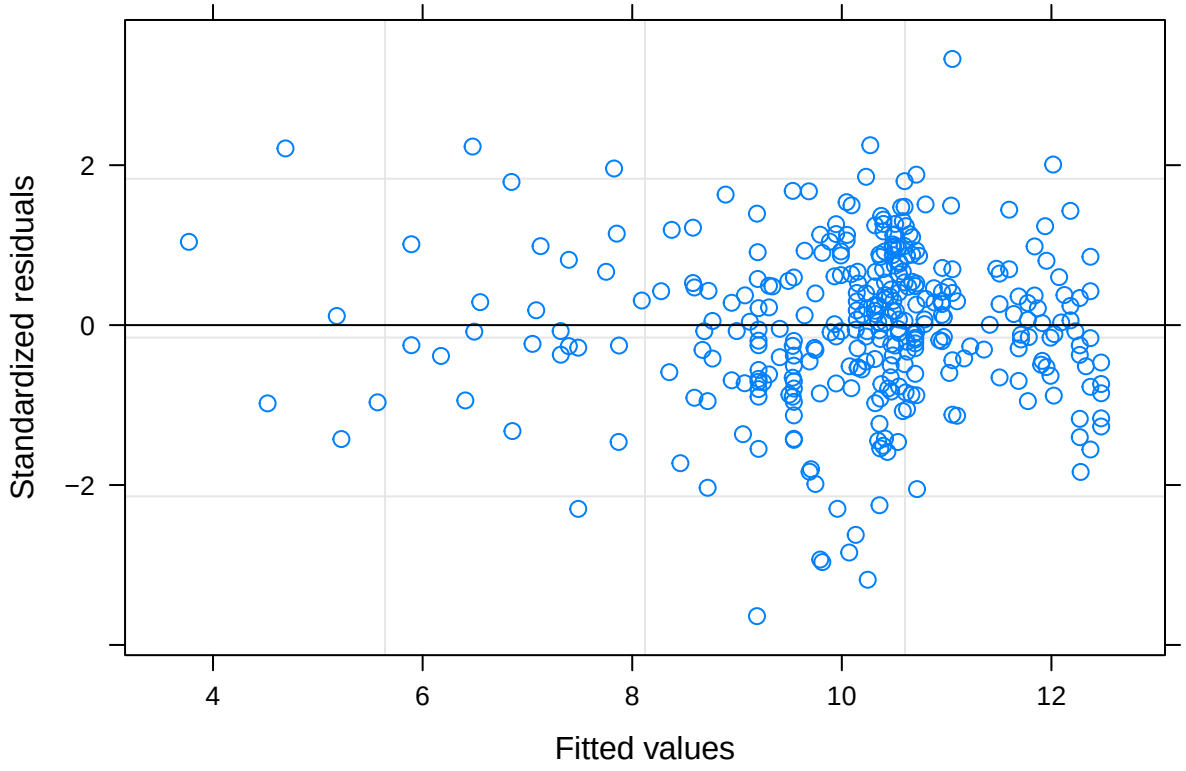
```
r.squaredGLMM(gs_ndvi_model)
```

```
## Warning: 'r.squaredGLMM' now calculates a revised statistic. See the help page.
```

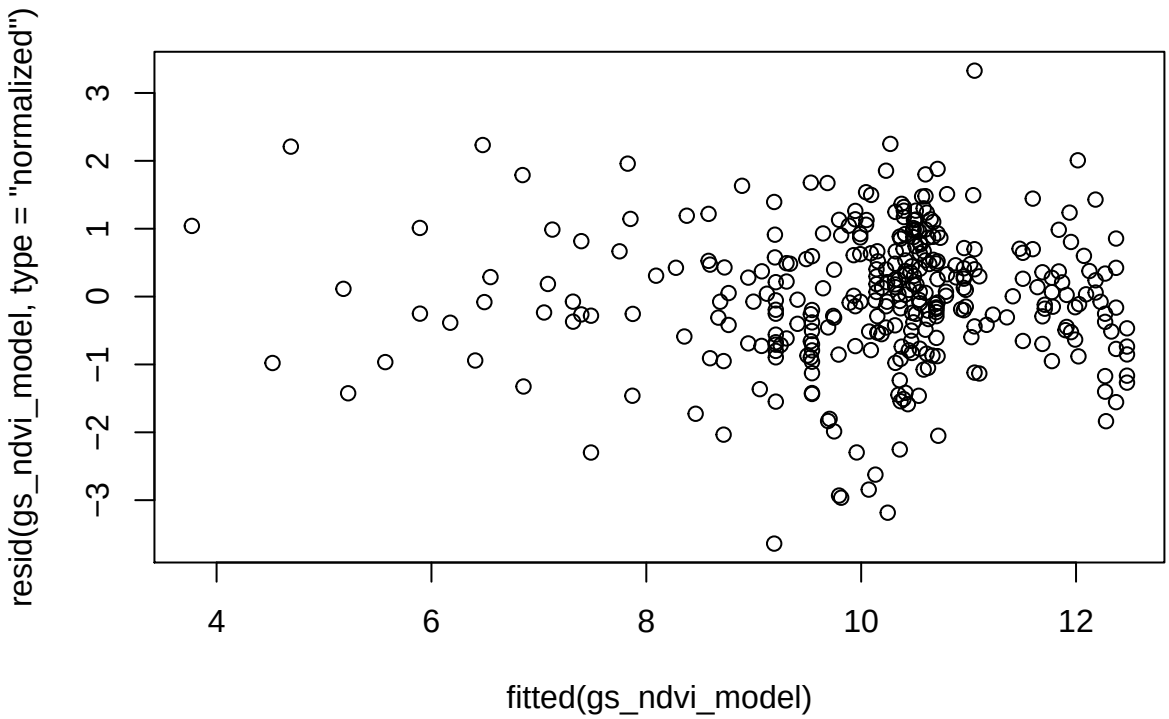
```
##             R2m      R2c
## [1,] 0.4483278 0.776302
```

### model diagnostics

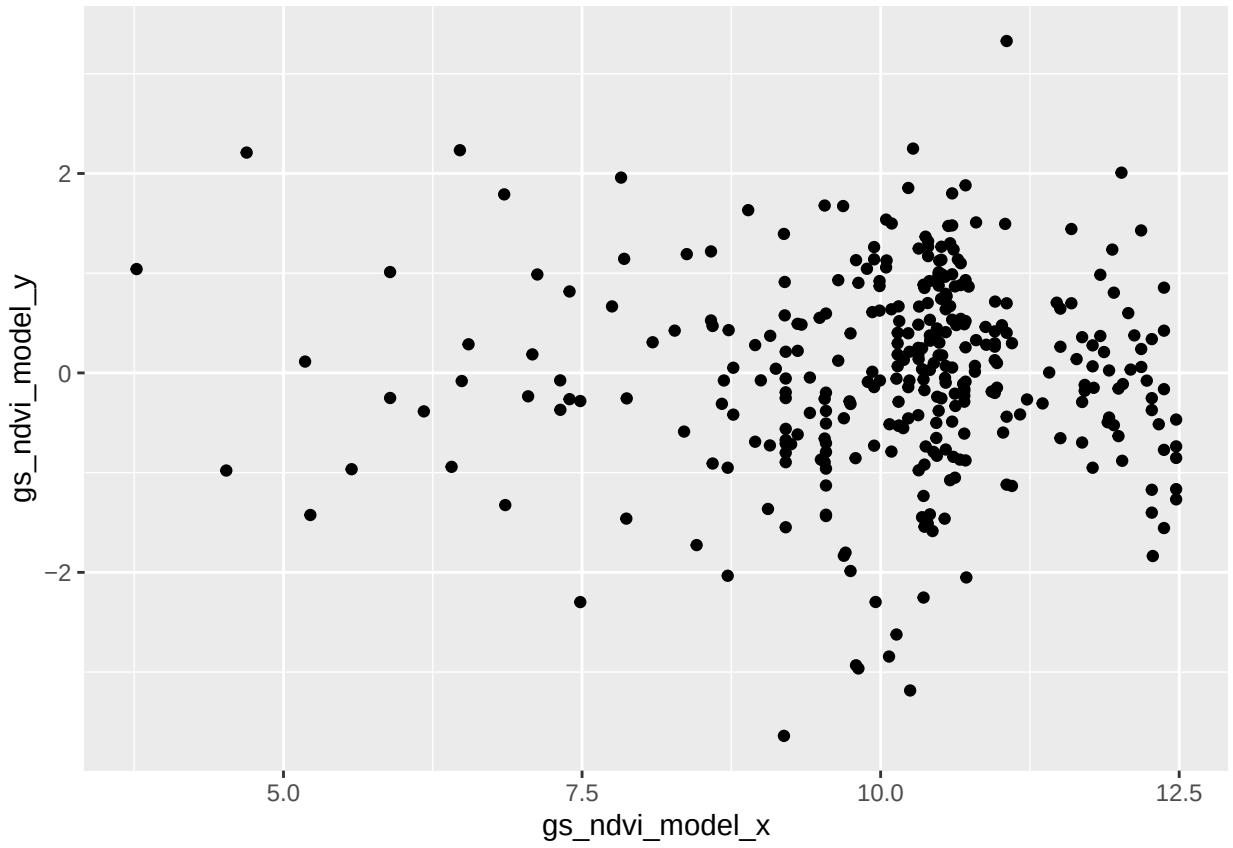
```
plot (gs_ndvi_model)
```



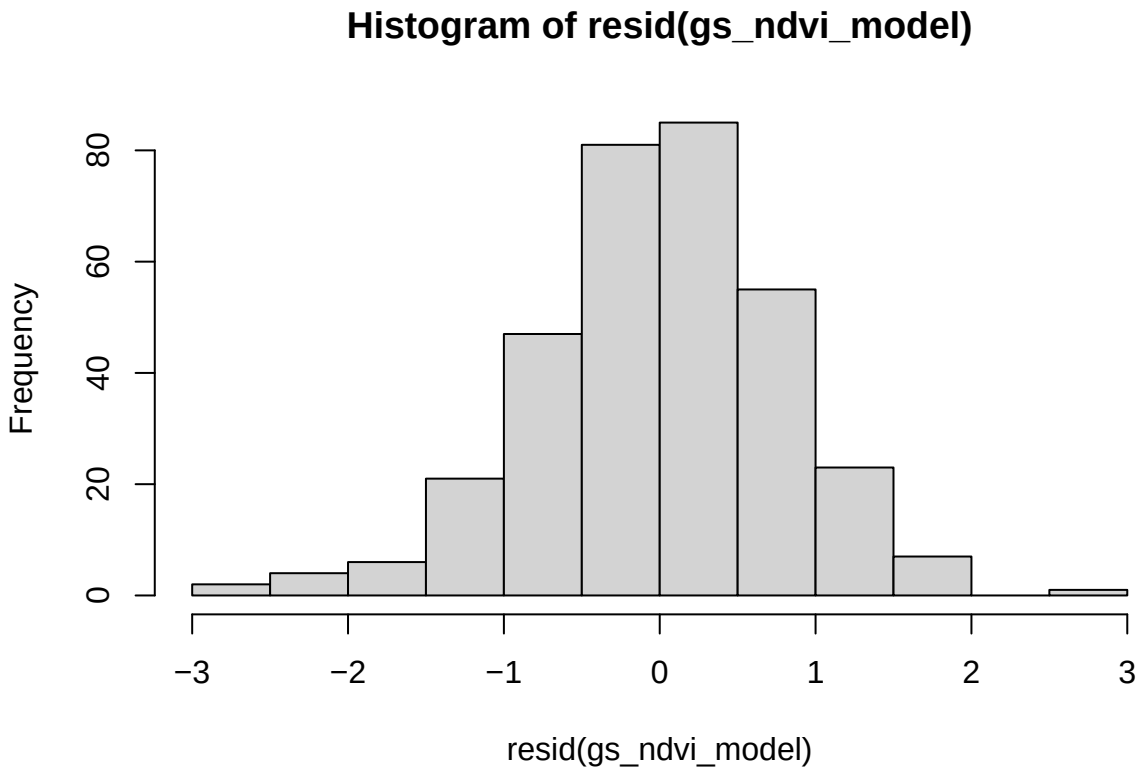

```
plot(resid(gs_ndvi_model, type = "normalized") ~fitted(gs_ndvi_model))
```



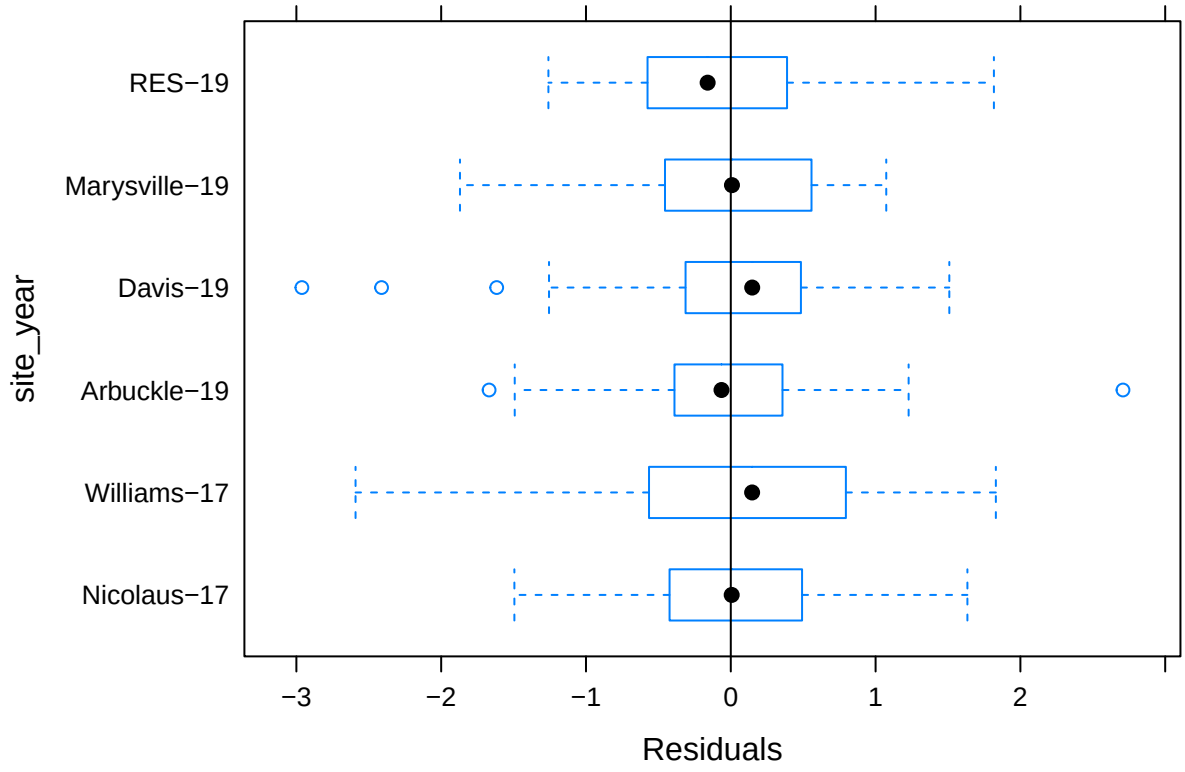

```
gs_ndvi_model_y <- resid(gs_ndvi_model, type = "normalized")
gs_ndvi_model_x <- fitted(gs_ndvi_model)

gs_ndvi_modelresid_data <- data.frame(gs_ndvi_model_x , gs_ndvi_model_y)

ggplot( data = gs_ndvi_modelresid_data , aes( x = gs_ndvi_model_x , y = gs_ndvi_model_y)) +
  geom_point(mapping = aes(gs_ndvi_model_x , gs_ndvi_model_y) , data = gs_ndvi_modelresid_data)
```



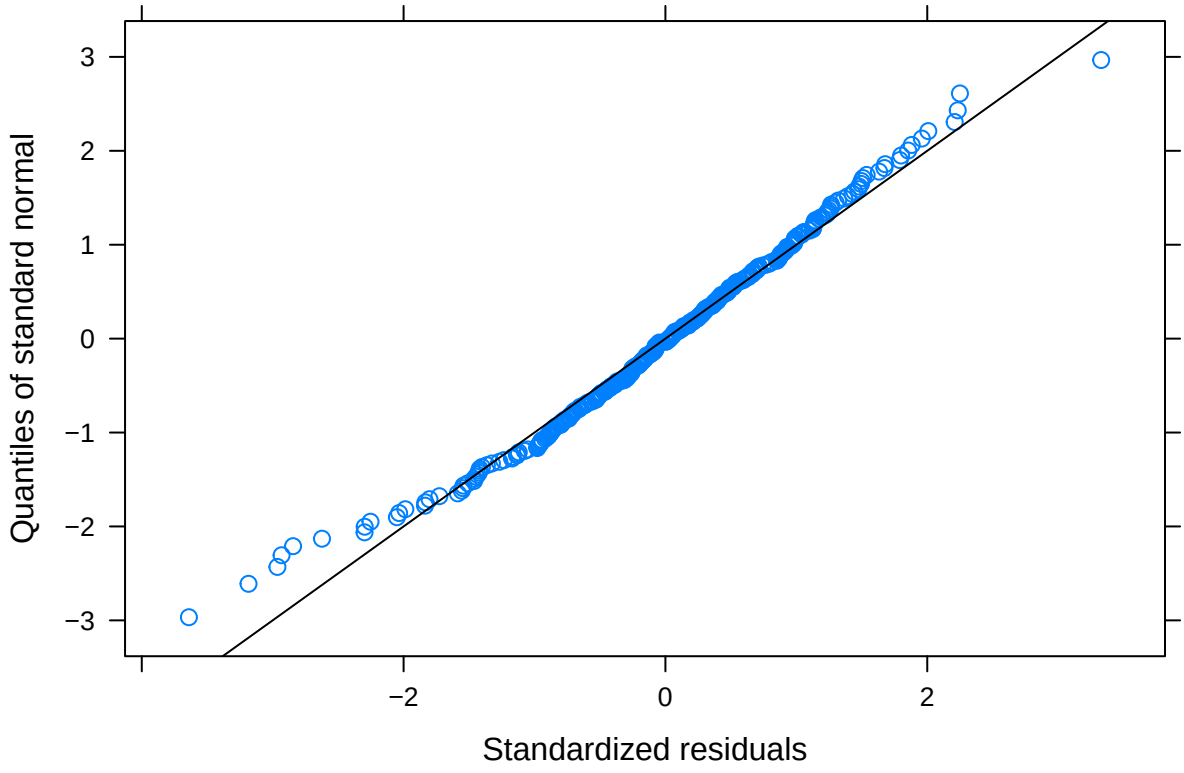

```
hist(resid(gs_ndvi_model))
```



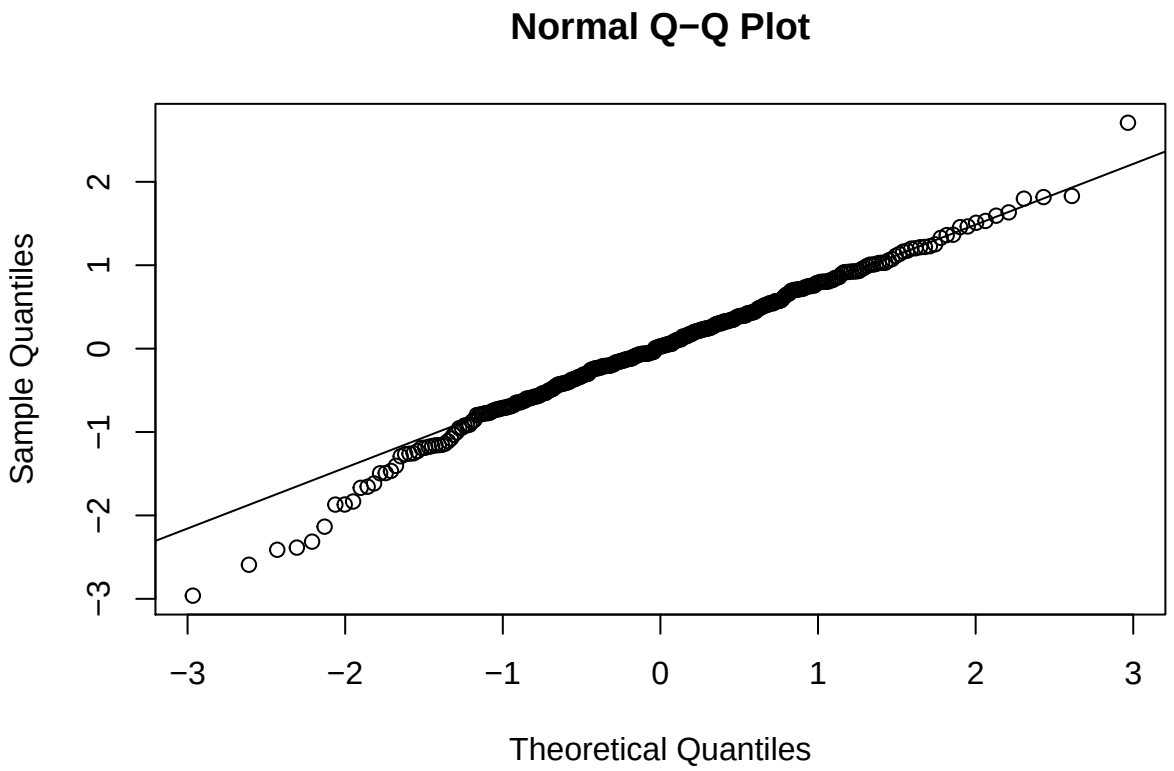
```
plot(gs_ndvi_model, site_year ~ resid(.), abline = 0)
```



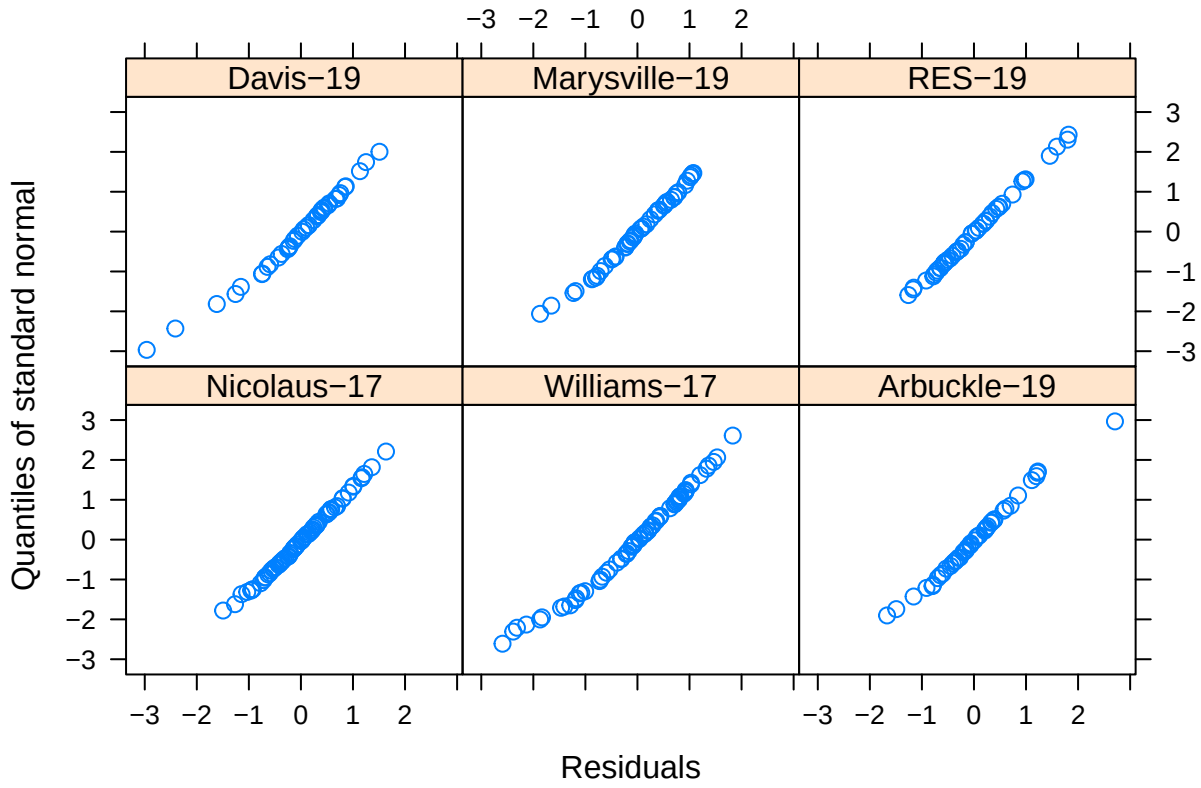

```
qqnorm(gs_ndvi_model, abline = c(0,1) )
```



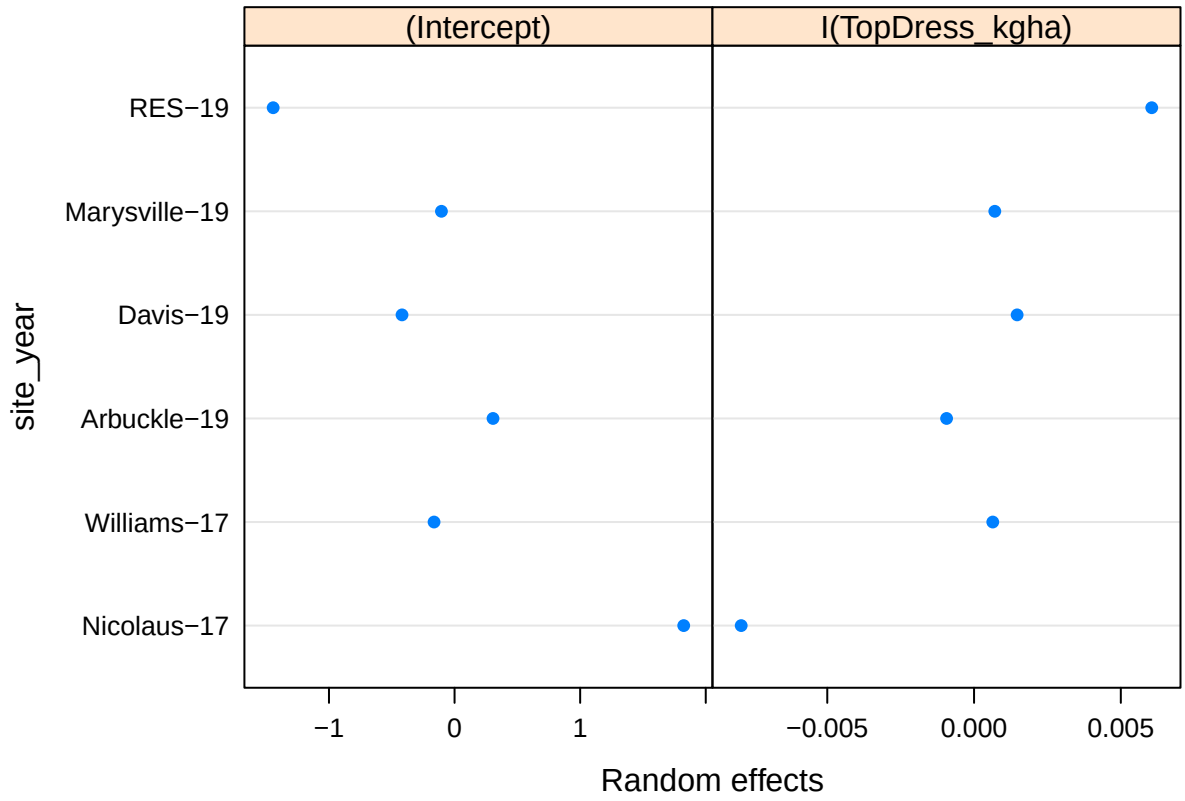
```
qqnorm(resid(gs_ndvi_model))
qqline(resid(gs_ndvi_model))
```



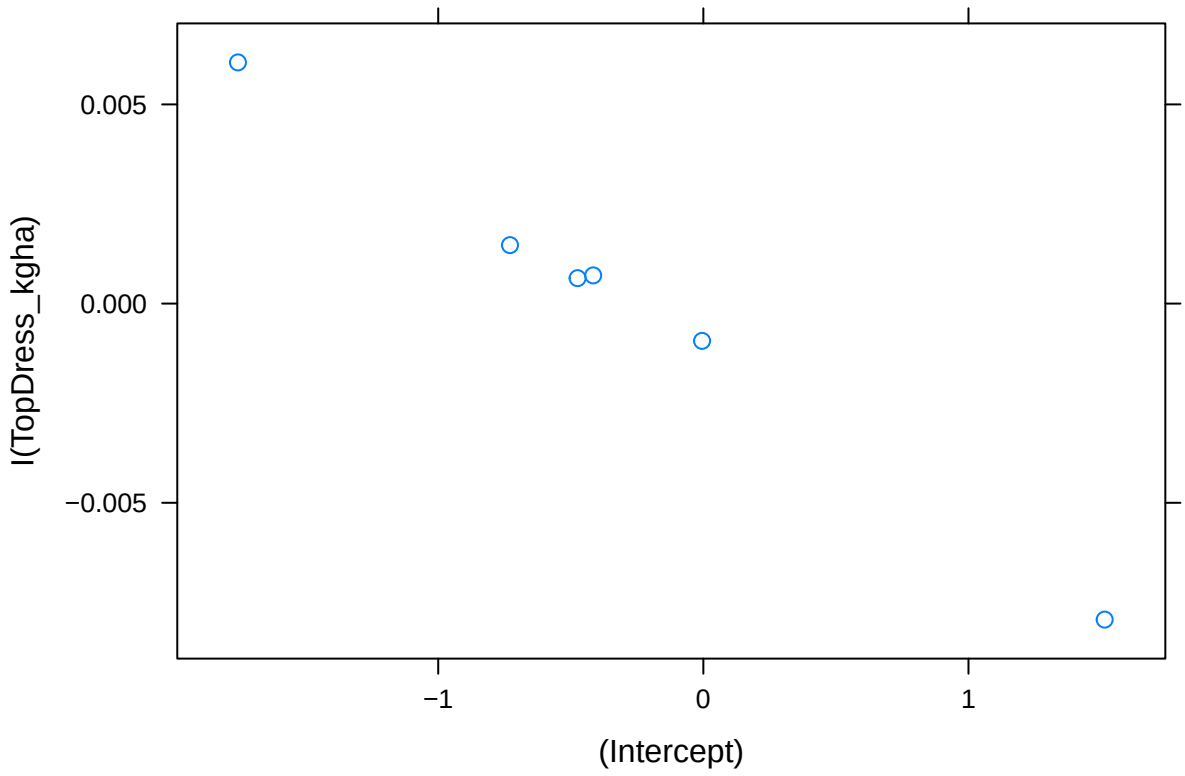

```
qqnorm(gs_ndvi_model , ~resid(.) | site_year)
```



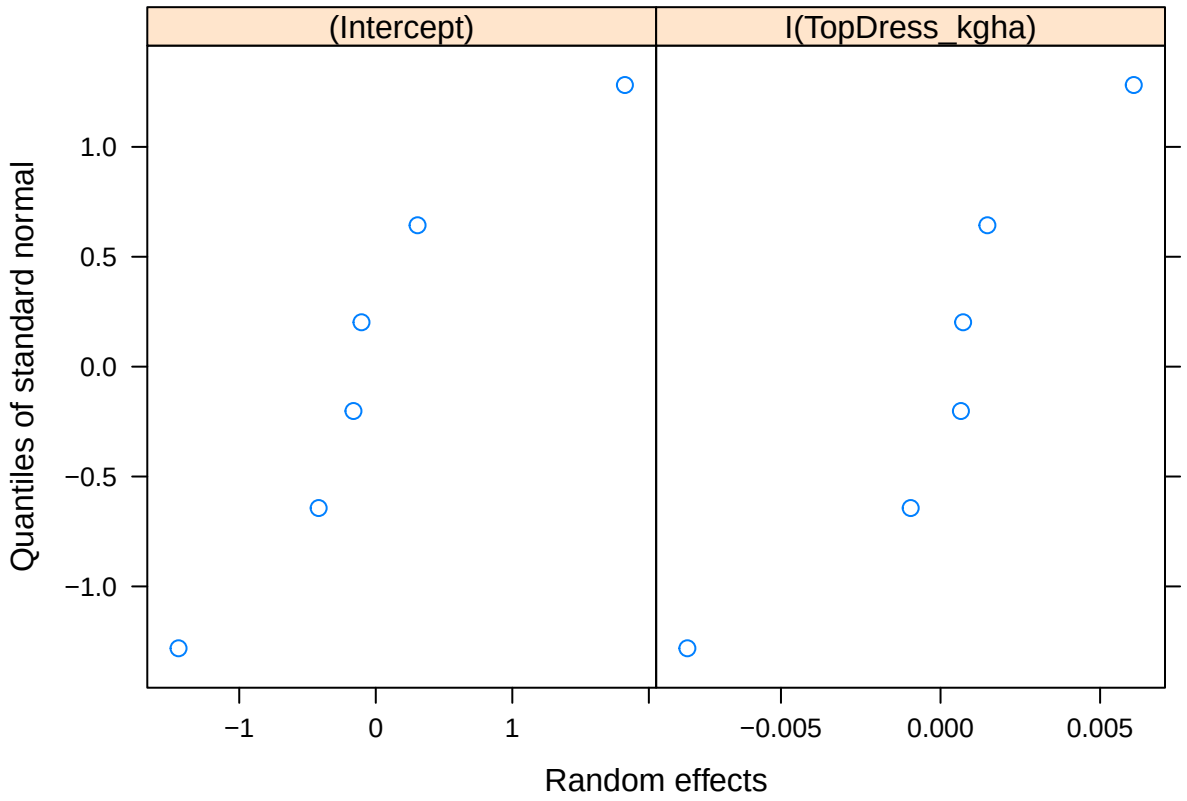
```
plot(ranef(gs_ndvi_model))
```



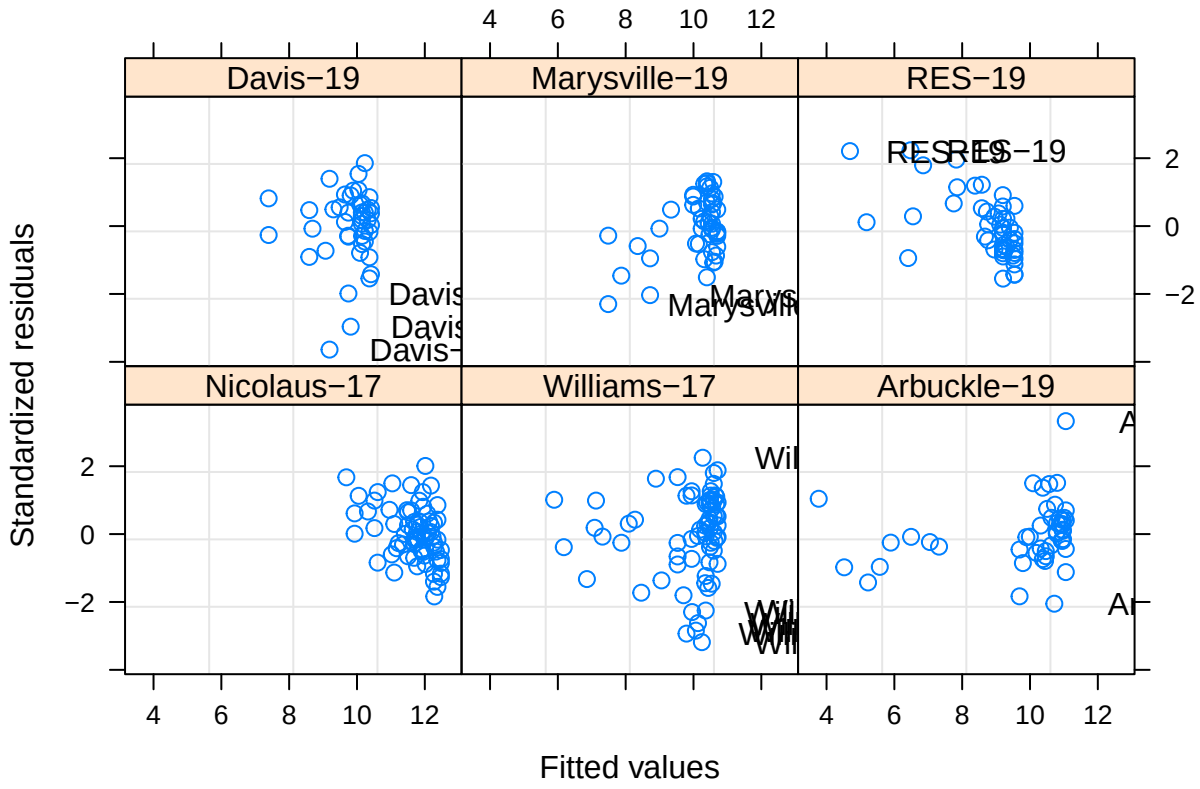

```
pairs(gs_ndvi_model , id = 0.1)
```



```
qqnorm(gs_ndvi_model , ~ranef(.))
```




```
plot( gs_ndvi_model, resid(., type = "p") ~ fitted(.) | site_year,
      id = 0.05, adj = -0.3 )
```




###outlier removal

```
greenseeker_ndvi_data_OR <- greenseeker_ndvi_data[-(c(which(abs(residuals(gs_ndvi_model,type="normalize
```

###rerun model sans outliers

```
ctrl <- lmeControl(opt = "optim") #changes control to "optimal" settings

gs_ndvi_model_OR <- lme(GrainYield_Mgha ~ SI * TopDress_kgha + I(SI*SI)  ,
             control = ctrl ,
             random = ~ I(TopDress_kgha) | site_year  ,
             data = greenseeker_ndvi_data_OR)

summary(gs_ndvi_model_OR)
```

```
## Linear mixed-effects model fit by REML
##   Data: greenseeker_ndvi_data_OR
##        AIC      BIC    logLik
##   818.3621 852.3888 -400.1811
##
## Random effects:
##  Formula: ~I(TopDress_kgha) | site_year
##  Structure: General positive-definite, Log-Cholesky parametrization
##                   StdDev      Corr
## (Intercept)      1.061801398 (Intr)
## I(TopDress_kgha) 0.004744882 -0.942
## Residual         0.771366042
##
## Fixed effects:  GrainYield_Mgha ~ SI * TopDress_kgha + I(SI * SI)
##                       Value Std.Error  DF   t-value p-value
```

```
## (Intercept)      -0.476590 0.8784971 319 -0.542506  0.5878
## SI               17.147163 2.0736291 319  8.269156  0.0000
## TopDress_kgha     0.082491 0.0127849 319  6.452271  0.0000
## I(SI * SI)       -6.015697 1.3924269 319 -4.320296  0.0000
## SI:TopDress_kgha -0.079565 0.0142834 319 -5.570456  0.0000
##  Correlation:
##                  (Intr) SI     TpDrs_ I(SI*S
## SI               -0.834
## TopDress_kgha    -0.311  0.078
## I(SI * SI)        0.774 -0.982  0.056
## SI:TopDress_kgha  0.239 -0.082 -0.972 -0.056
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -3.21293731 -0.58644957  0.03185367  0.66072053  2.35677933
##
## Number of Observations: 329
## Number of Groups: 6
```

```
summary(gs_ndvi_model_OR)$tTable
```

```
##                       Value  Std.Error  DF    t-value      p-value
## (Intercept)      -0.47658960 0.87849707 319 -0.5425056 5.878490e-01
## SI               17.14716306 2.07362910 319  8.2691563 3.690901e-15
## TopDress_kgha     0.08249138 0.01278486 319  6.4522706 4.088965e-10
## I(SI * SI)       -6.01569684 1.39242688 319 -4.3202964 2.083895e-05
## SI:TopDress_kgha -0.07956498 0.01428339 319 -5.5704563 5.404405e-08
```

```
Anova(gs_ndvi_model_OR , type = 3)
```

```
## Analysis of Deviance Table (Type III tests)
##
## Response: GrainYield_Mgha
##                    Chisq Df Pr(>Chisq)
## (Intercept)       0.2943  1     0.5875
## SI               68.3789  1  < 2.2e-16 ***
## TopDress_kgha    41.6318  1  1.102e-10 ***
## I(SI * SI)       18.6650  1  1.558e-05 ***
## SI:TopDress_kgha 31.0300  1  2.541e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
r.squaredGLMM(gs_ndvi_model_OR)
```

```
##            R2m       R2c
## [1,] 0.4590344 0.7917617
```

```
gs_ndvi_r_sq <- r.squaredGLMM(gs_ndvi_model_OR)

gs_ndvi_r_sq_fixed <- round(gs_ndvi_r_sq[1] , digits = 2)
gs_ndvi_r_sq_fixed
```

```
## [1] 0.46
```

```
gs_ndvi_r_sq_total <- round(gs_ndvi_r_sq[2] , digits = 2)
gs_ndvi_r_sq_total
```

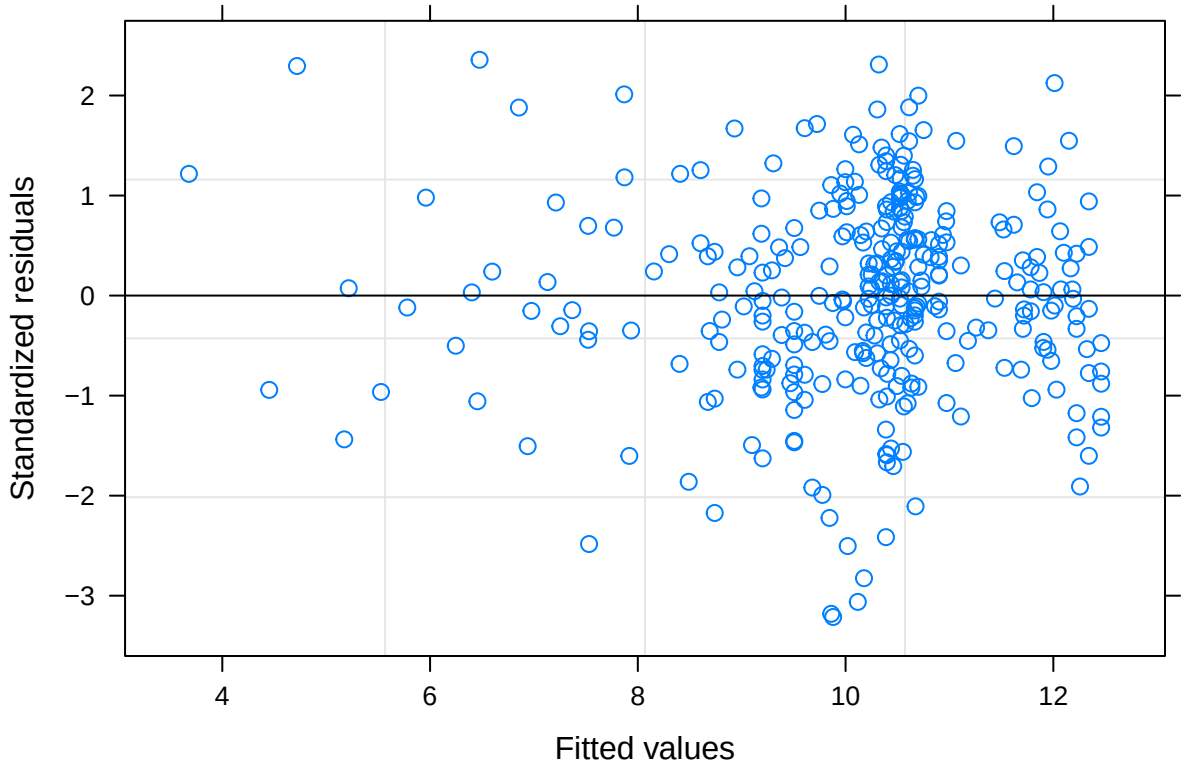
```
## [1] 0.79
```

```
gs_ndvi_r_sq_random <- gs_ndvi_r_sq_total - gs_ndvi_r_sq_fixed
gs_ndvi_r_sq_random
```

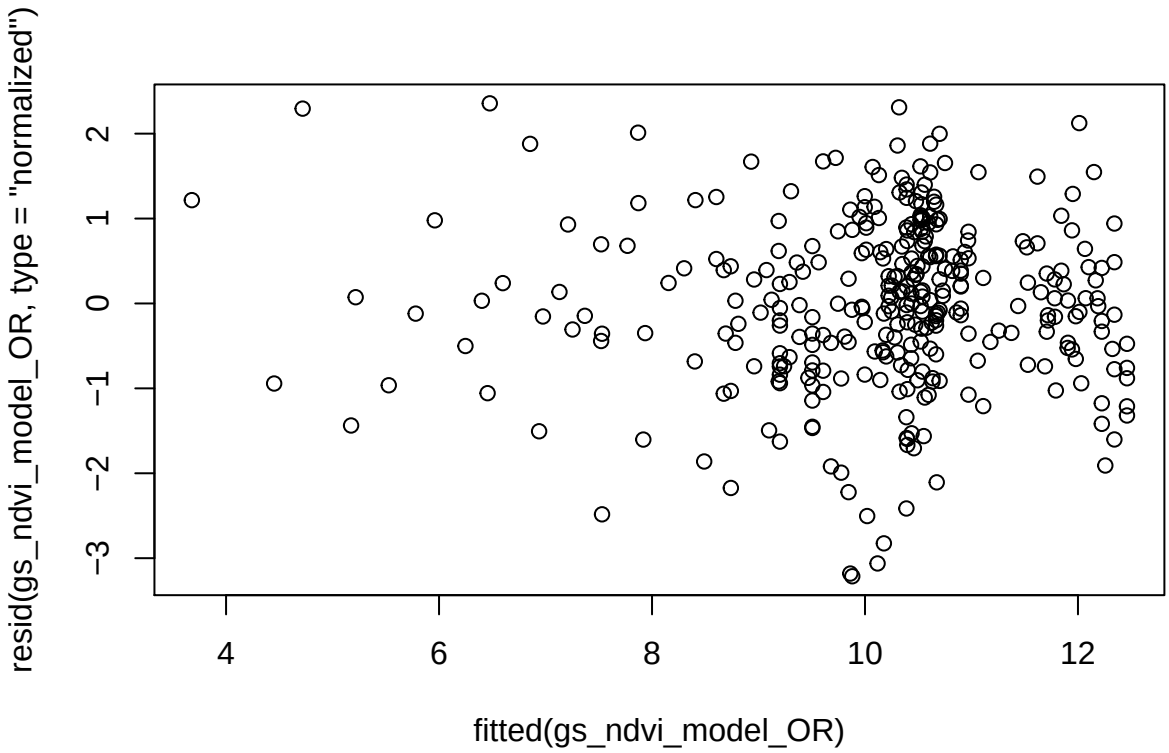
```
## [1] 0.33
```

### model diagnostics

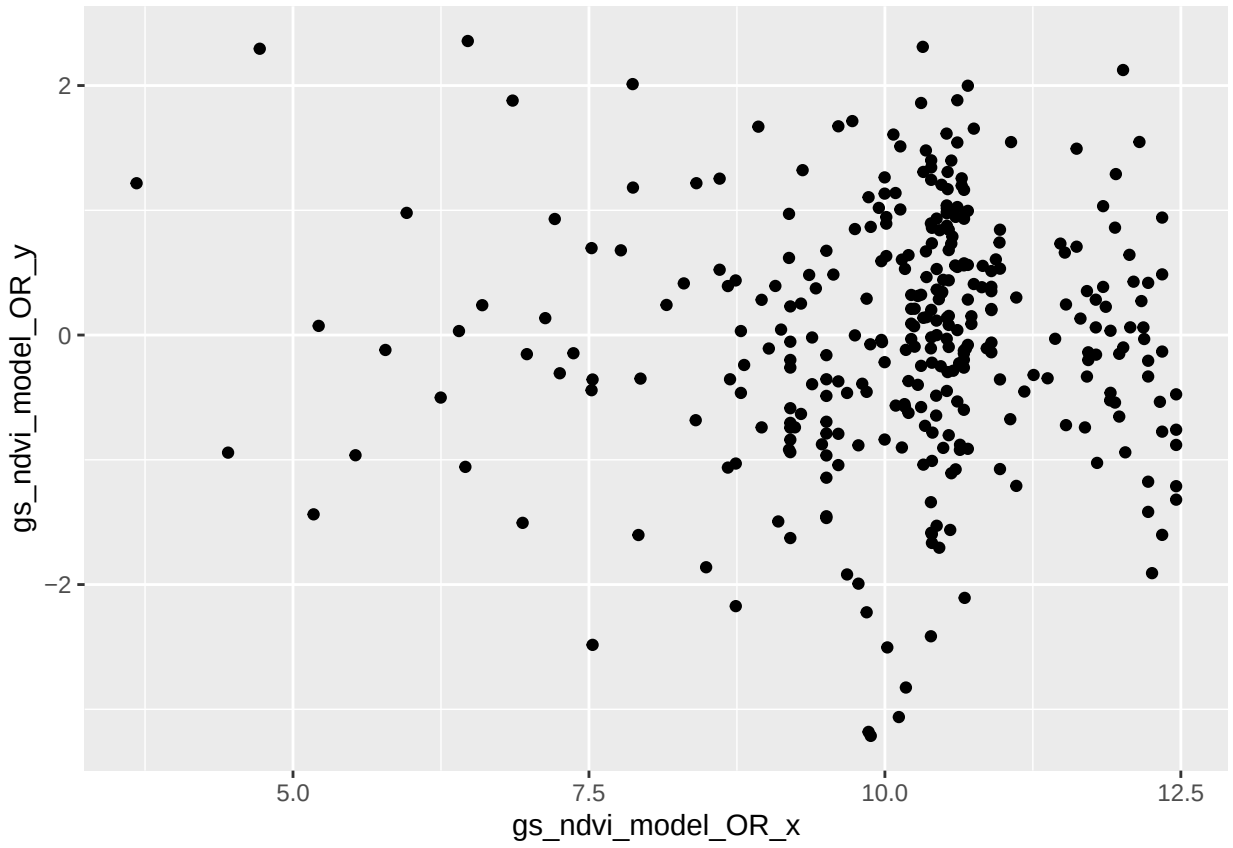
```
plot (gs_ndvi_model_OR)
```



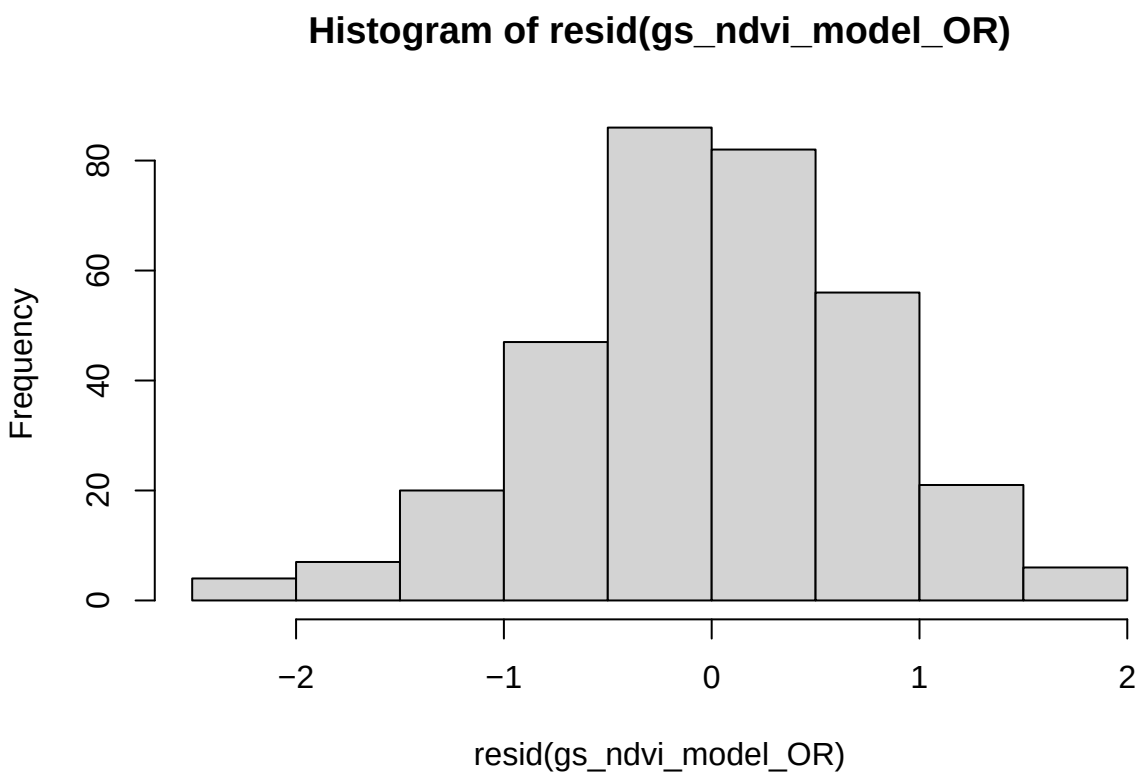

```
plot(resid(gs_ndvi_model_OR, type = "normalized") ~fitted(gs_ndvi_model_OR))
```



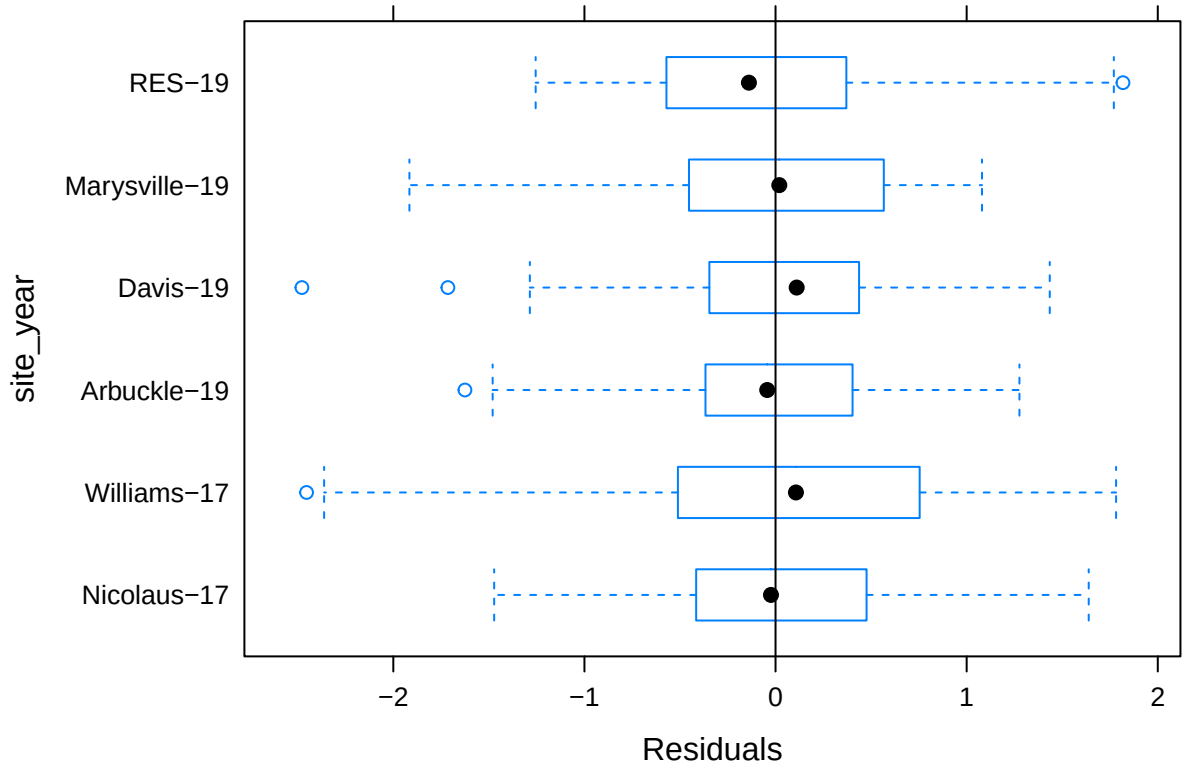

```
gs_ndvi_model_OR_y <- resid(gs_ndvi_model_OR, type = "normalized")
gs_ndvi_model_OR_x <- fitted(gs_ndvi_model_OR)

gs_ndvi_model_ORresid_data <- data.frame(gs_ndvi_model_OR_x , gs_ndvi_model_OR_y)

ggplot( data = gs_ndvi_model_ORresid_data , aes( x = gs_ndvi_model_OR_x , y = gs_ndvi_model_OR_y)) +
  geom_point(mapping = aes(gs_ndvi_model_OR_x , gs_ndvi_model_OR_y) , data = gs_ndvi_model_ORresid_data
```



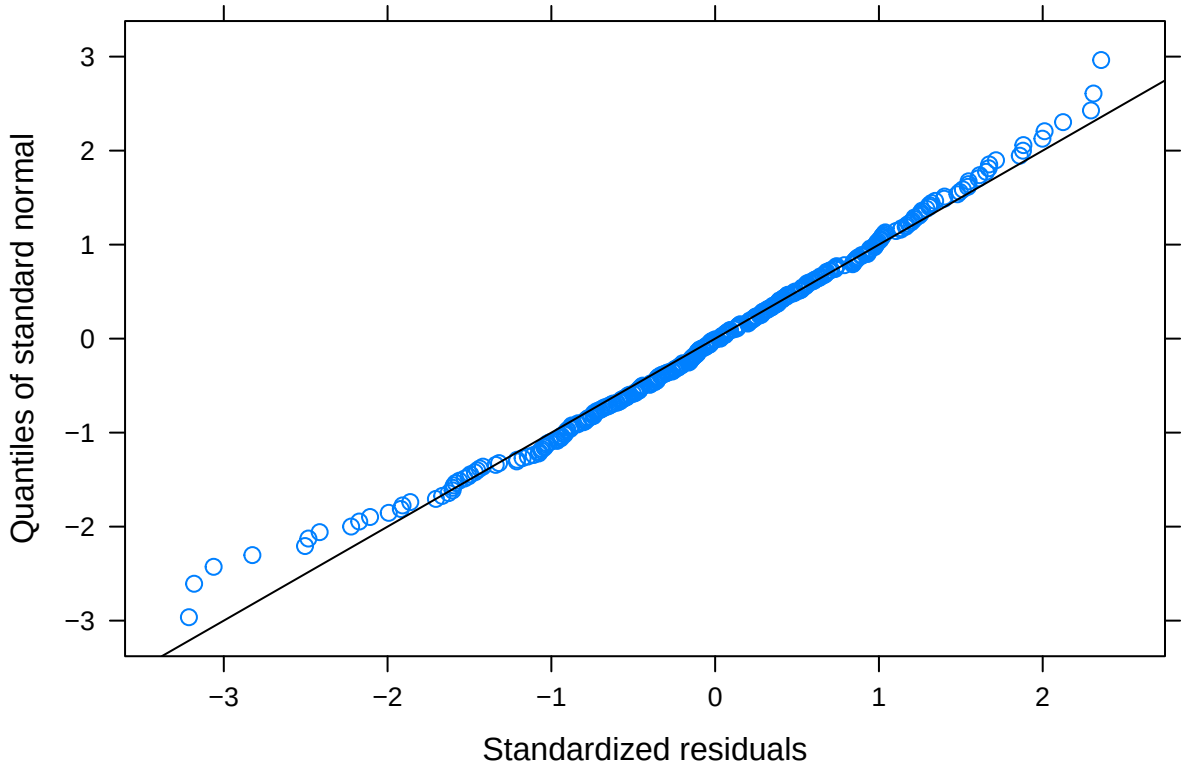

```
hist(resid(gs_ndvi_model_OR))
```



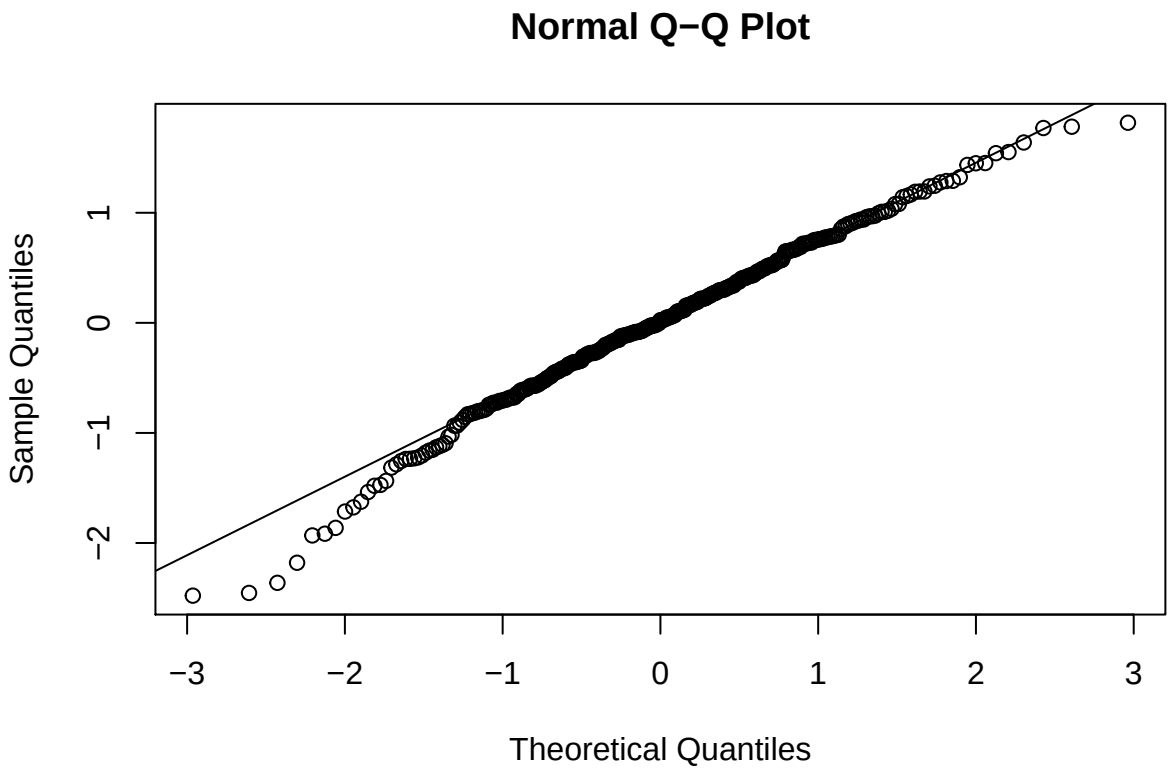
```
plot(gs_ndvi_model_OR, site_year ~ resid(.), abline = 0)
```



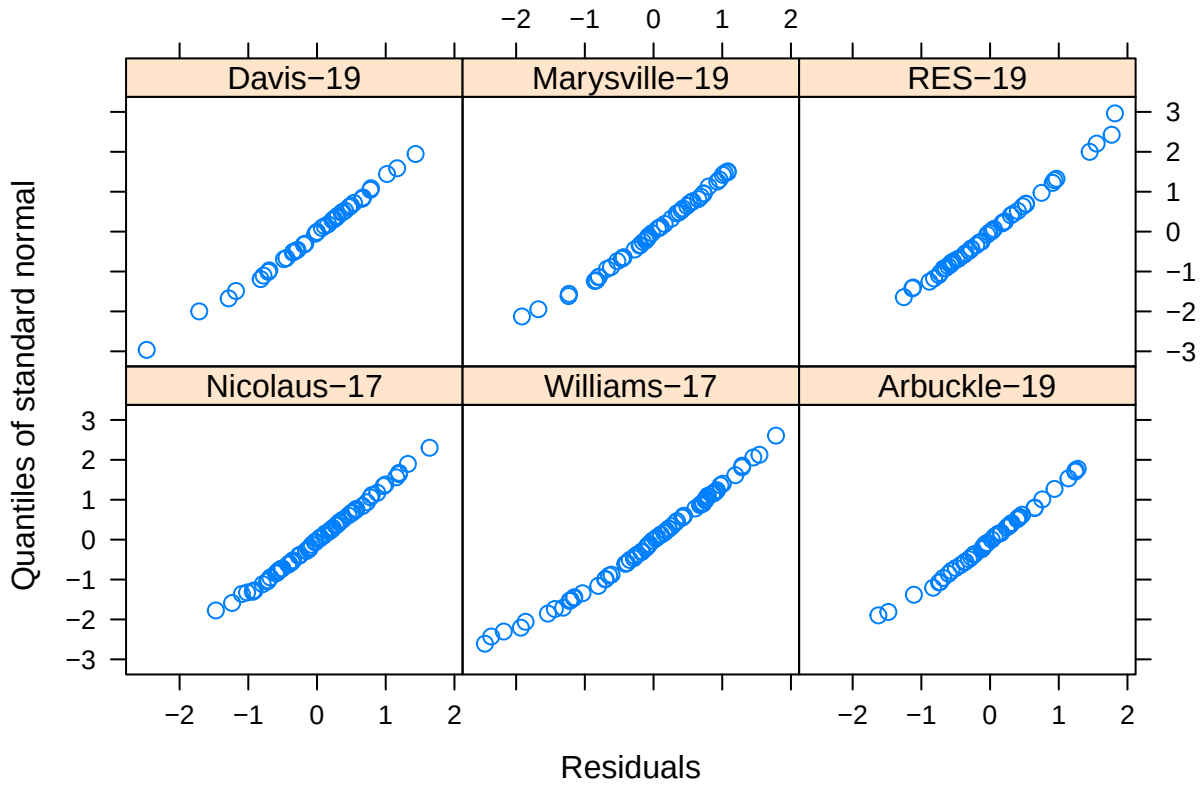

```
qqnorm(gs_ndvi_model_OR, abline = c(0,1) )
```



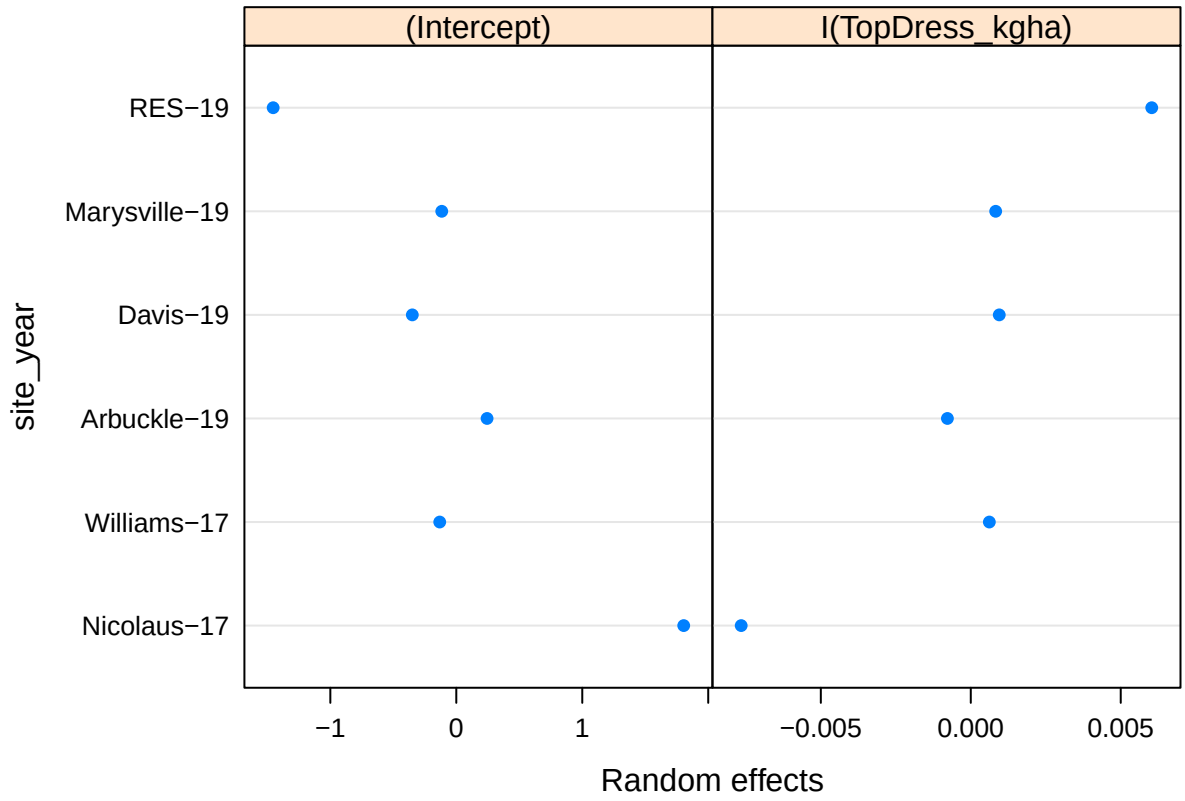
```
qqnorm(resid(gs_ndvi_model_OR))
qqline(resid(gs_ndvi_model_OR))
```



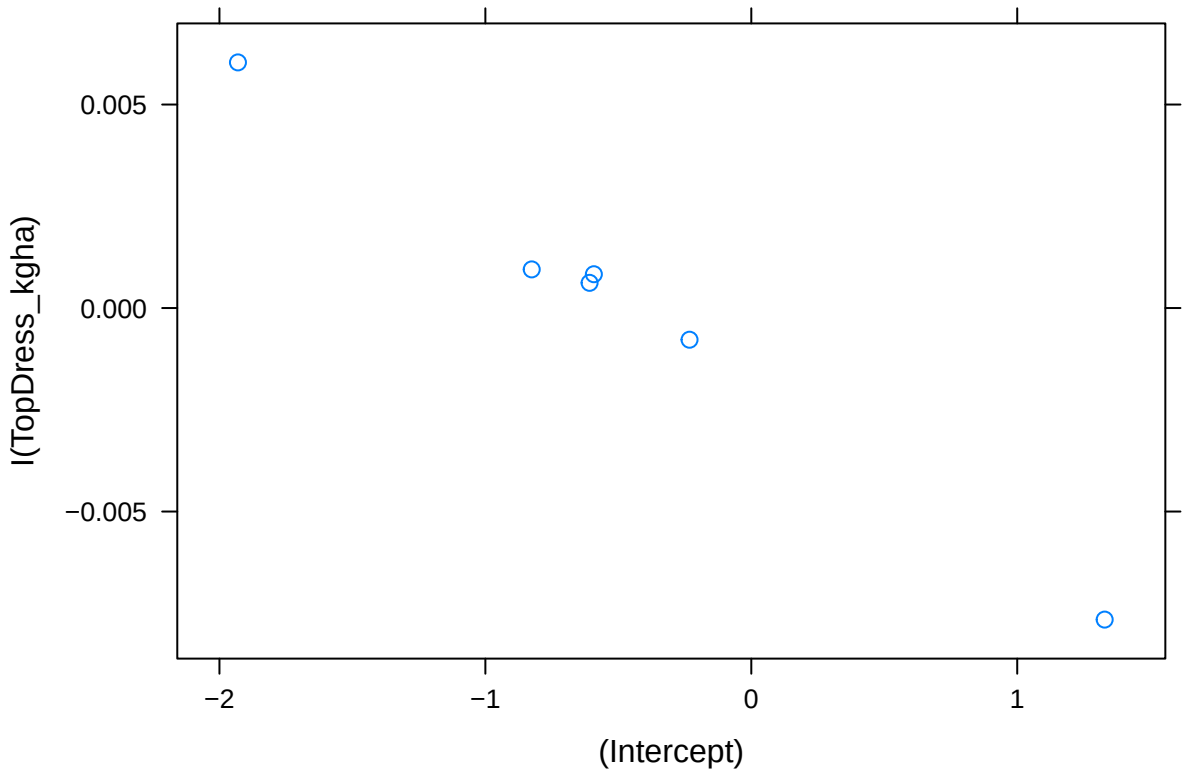

```
qqnorm(gs_ndvi_model_OR , ~resid(.) | site_year)
```



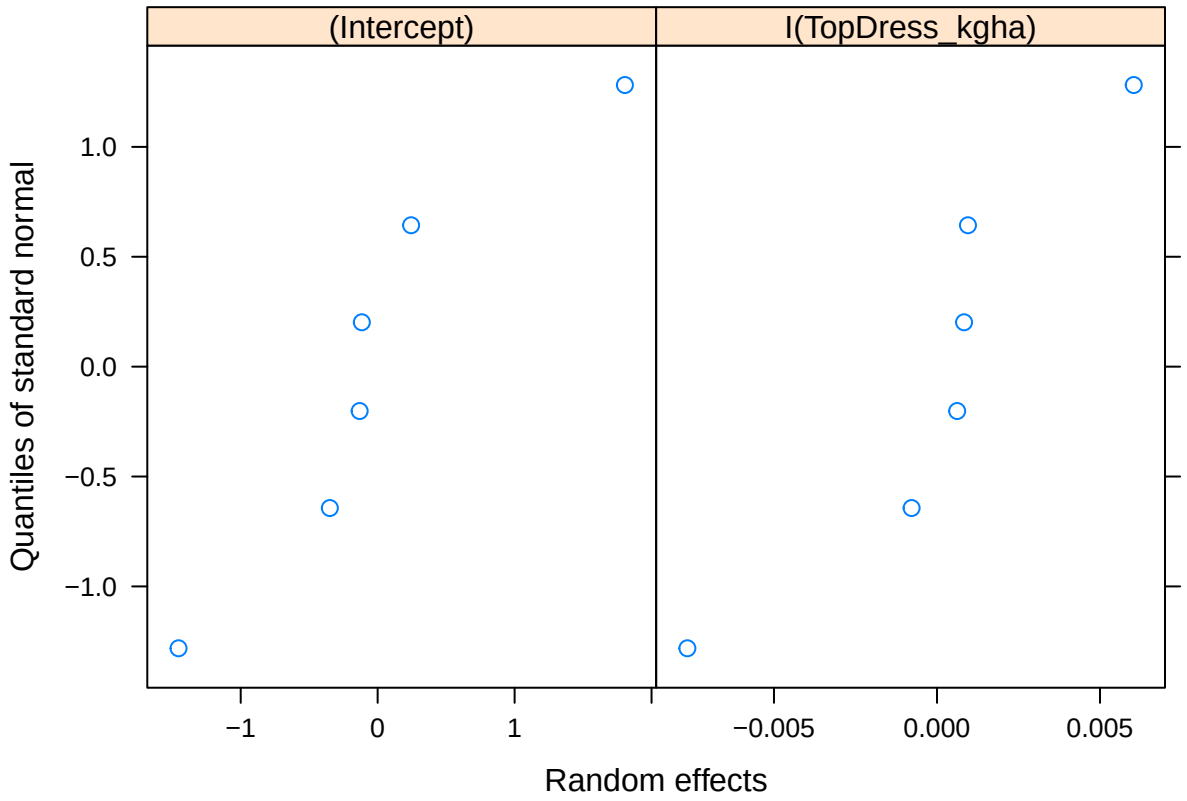
```
plot(ranef(gs_ndvi_model_OR))
```



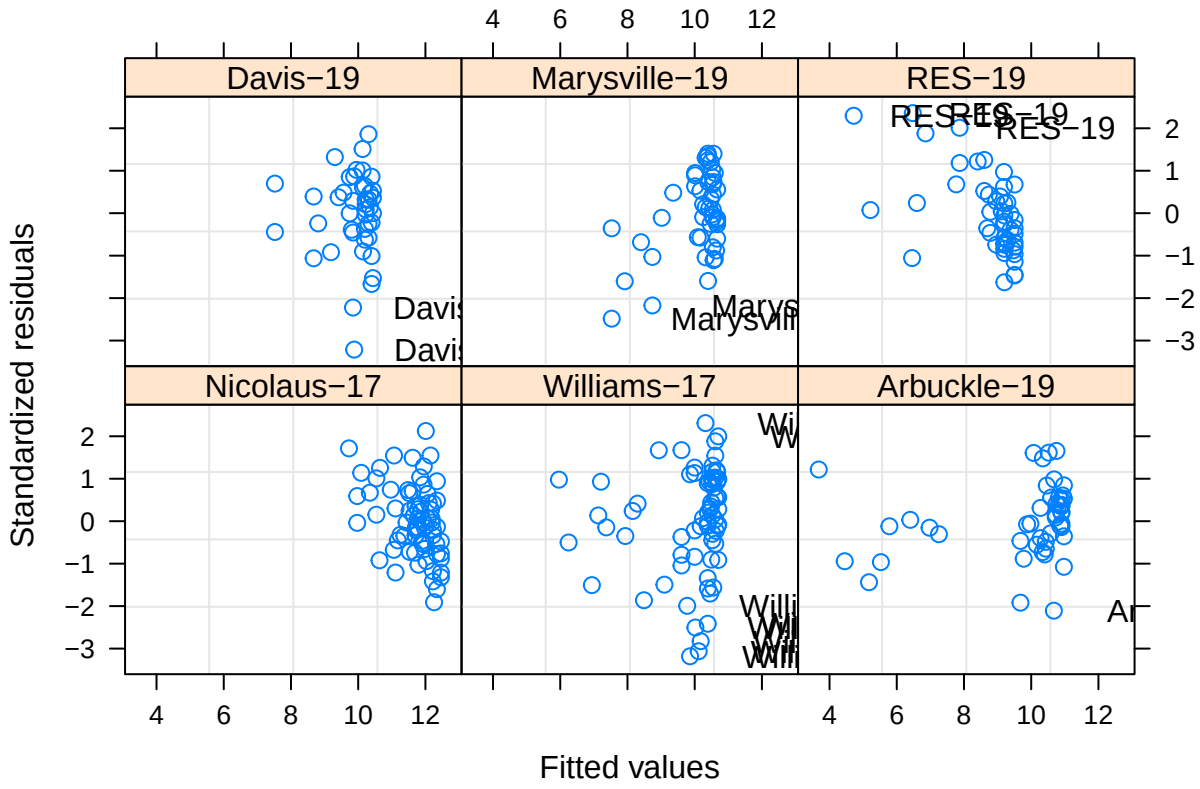

```
pairs(gs_ndvi_model_OR , id = 0.1)
```



```
qqnorm(gs_ndvi_model_OR , ~ranef(.))
```




```
plot( gs_ndvi_model_OR, resid(., type = "p") ~ fitted(.) | site_year,
      id = 0.05, adj = -0.3 )
```




###emmeans

```
mylist <- list(SI=seq(0.25 , 1 , by = .0005),TopDress_kgha =c( 34 , 0))

gs_ndvi_emmeans <- emmeans(gs_ndvi_model_OR , ~TopDress_kgha * SI , at = mylist )

gs_ndvi_emmeans_contrast <- as.data.frame(summary(contrast(gs_ndvi_emmeans , "pairwise" , side = ">" , l

gs_ndvi_emmeans_contrast <- gs_ndvi_emmeans_contrast %>%
  mutate(GS_NDVI_Response_Index = 1 / SI )

gs_ndvi_emmeans_contrast <- gs_ndvi_emmeans_contrast %>%
  mutate(prob_postive_resp = (1 - p.value) * 100)

gs_ndvi_emmeans_contrast <- gs_ndvi_emmeans_contrast %>%
  mutate(t_score = abs((estimate - 0.26)) / SE)

gs_ndvi_emmeans_contrast <- gs_ndvi_emmeans_contrast %>%
  mutate(prob = if_else(estimate < 0.26 , pt(q = t_score , df = df , lower.tail = F) , pt(q = t_score ,

gs_ndvi_emmeans_contrast$prob_percent <- (round(gs_ndvi_emmeans_contrast$prob , digits = 3) * 100)

gs_ndvi_emmeans_contrast$GS_NDVI_Response_Index_r <- round(gs_ndvi_emmeans_contrast$GS_NDVI_Response_Inc

gs_ndvi_run1 <- 125 - 100
gs_ndvi_rise1 <- 0.6404977 - 0.09953884
gs_ndvi_slope1 <- round((gs_ndvi_rise1 / gs_ndvi_run1) * 5 , digits = 2)
gs_ndvi_slope1
```

```
## [1] 0.11
```

```
gs_ndvi_run2 <- 250 - 125
gs_ndvi_rise2 <- 1.722415 - 0.6404977
gs_ndvi_slope2 <- round((gs_ndvi_rise2 / gs_ndvi_run2) * 5 , digits = 3)
gs_ndvi_slope2
```

```
## [1] 0.043
```

```
gs_ndvi_run3 <- 400 - 250
gs_ndvi_rise3 <- 2.128135 - 1.722415
gs_ndvi_slope3 <- round((gs_ndvi_rise3 / gs_ndvi_run3) * 5 , digits = 3)
gs_ndvi_slope3
```

```
## [1] 0.014
```

```
gs_ndvi_mean_se <- gs_ndvi_emmeans_contrast %>%
  select(SE) %>%
  summarise(mean_se = mean(SE))

gs_ndvi_mean_se$mean_se <- round(gs_ndvi_mean_se$mean_se , digits = 2)

gs_ndvi_mean_se$mean_se
```

```
## [1] 0.18
```

###confidence interval

```
#at every 0.10 interval, what is the 90% confidence interval of the yield response, and what RI value d

str(gs_ndvi_emmeans_contrast , give.attr = F)
```

```
## 'data.frame':    1501 obs. of  13 variables:
##  $ contrast               : Factor w/ 1 level "34 - 0": 1 1 1 1 1 1 1 1 1 1 ...
##  $ SI                     : num  0.25 0.251 0.251 0.252 0.252 ...
##  $ estimate               : num  2.13 2.13 2.13 2.12 2.12 ...
##  $ SE                     : num  0.318 0.318 0.317 0.317 0.317 ...
##  $ df                     : num  319 319 319 319 319 319 319 319 319 319 ...
##  $ t.ratio                : num  6.69 6.69 6.7 6.7 6.7 ...
##  $ p.value                : num  4.88e-11 4.87e-11 4.85e-11 4.83e-11 4.82e-11 ...
##  $ GS_NDVI_Response_Index : num  4 3.99 3.98 3.98 3.97 ...
##  $ prob_postive_resp      : num  100 100 100 100 100 ...
##  $ t_score                : num  5.88 5.88 5.88 5.88 5.88 ...
##  $ prob                   : num  1 1 1 1 1 ...
##  $ prob_percent           : num  100 100 100 100 100 100 100 100 100 100 ...
##  $ GS_NDVI_Response_Index_r: num  4 3.99 3.98 3.98 3.97 ...
```

```
gs_ndvi_emmeans_contrast <- gs_ndvi_emmeans_contrast %>%
  mutate(response_mgha_r = round(estimate , digits = 2),
         se_mgha_r = round(SE , digits = 3))

z_score <- 1.645

#estimate = 0.10 , SE = 0.120

print(round(0.10 - z_score*0.120 , digits = 2))
```

```
## [1] -0.1
```

```
print(round(0.10 + z_score*0.120 , digits = 2))
```

```
## [1] 0.3
```

```
#estimate = 0.20 , SE = 0.112

print(round(0.20 - z_score*0.112 , digits = 2))
```

```
## [1] 0.02
```

```
print(round(0.20 + z_score*0.112 , digits = 2))
```

```
## [1] 0.38
```

```
#estimate = 0.30 , SE = 0.106

print(round(0.30 - z_score*0.106 , digits = 2))
```

```
## [1] 0.13
```

```
print(round(0.30 + z_score*0.106 , digits = 2))
```

```
## [1] 0.47
```

```
#estimate = 0.40 , SE = 0.102

print(round(0.40 - z_score*0.102 , digits = 2))
```

```
## [1] 0.23
```

```
print(round(0.40 + z_score*0.102 , digits = 2))
```

```
## [1] 0.57
```

```
#estimate = 0.50 , SE = 0.102

print(round(0.50 - z_score*0.102 , digits = 2))
```

```
## [1] 0.33
```

```
print(round(0.50 + z_score*0.102 , digits = 2))
```

```
## [1] 0.67
```

##sUAS NDVI-RI

```
sUAS_ndvi_data <- paper3_data_no_2018 %>%
  filter(Platform == "sUAS_NDVI")

ctrl <- lmeControl(opt = "optim") #changes control to "optimal" settings

sUAS_ndvi_model <- lme(GrainYield_Mgha ~ SI * TopDress_kgha + I(SI*SI)  ,
              control = ctrl ,
              random = ~ I(TopDress_kgha) | site_year  ,
              data = sUAS_ndvi_data)

summary(sUAS_ndvi_model)
```

```
## Linear mixed-effects model fit by REML
##   Data: sUAS_ndvi_data
##       AIC      BIC    logLik
```

```
##   821.447 855.5566 -401.7235
##
## Random effects:
##  Formula: ~I(TopDress_kgha) | site_year
##  Structure: General positive-definite, Log-Cholesky parametrization
##                  StdDev      Corr
## (Intercept)      1.071047781 (Intr)
## I(TopDress_kgha) 0.004528693 -0.935
## Residual         0.776505570
##
## Fixed effects:  GrainYield_Mgha ~ SI * TopDress_kgha + I(SI * SI)
##                       Value Std.Error  DF   t-value p-value
## (Intercept)      -24.39218  7.976495 322 -3.058007  0.0024
## SI                47.63446 17.706195 322  2.690271  0.0075
## TopDress_kgha      0.20859  0.042869 322  4.865671  0.0000
## I(SI * SI)       -12.53880  9.815347 322 -1.277468  0.2024
## SI:TopDress_kgha  -0.20263  0.044502 322 -4.553251  0.0000
##  Correlation:
##                  (Intr) SI     TpDrs_ I(SI*S
## SI               -0.996
## TopDress_kgha    -0.025 -0.031
## I(SI * SI)        0.989 -0.998  0.078
## SI:TopDress_kgha  0.023  0.031 -0.998 -0.078
##
## Standardized Within-Group Residuals:
##          Min           Q1          Med           Q3          Max
## -2.676205354 -0.636885362 -0.005202893  0.586056673  3.342143075
##
## Number of Observations: 332
## Number of Groups: 6
```

```
summary(sUAS_ndvi_model)$tTable
```

```
##                       Value   Std.Error  DF   t-value      p-value
## (Intercept)      -24.3921810  7.97649532 322 -3.058007 2.414970e-03
## SI                47.6344646 17.70619487 322  2.690271 7.511739e-03
## TopDress_kgha      0.2085855  0.04286881 322  4.865671 1.788676e-06
## I(SI * SI)       -12.5387959  9.81534736 322 -1.277468 2.023572e-01
## SI:TopDress_kgha  -0.2026276  0.04450175 322 -4.553251 7.501114e-06
```

```
Anova(sUAS_ndvi_model , type = 3)
```

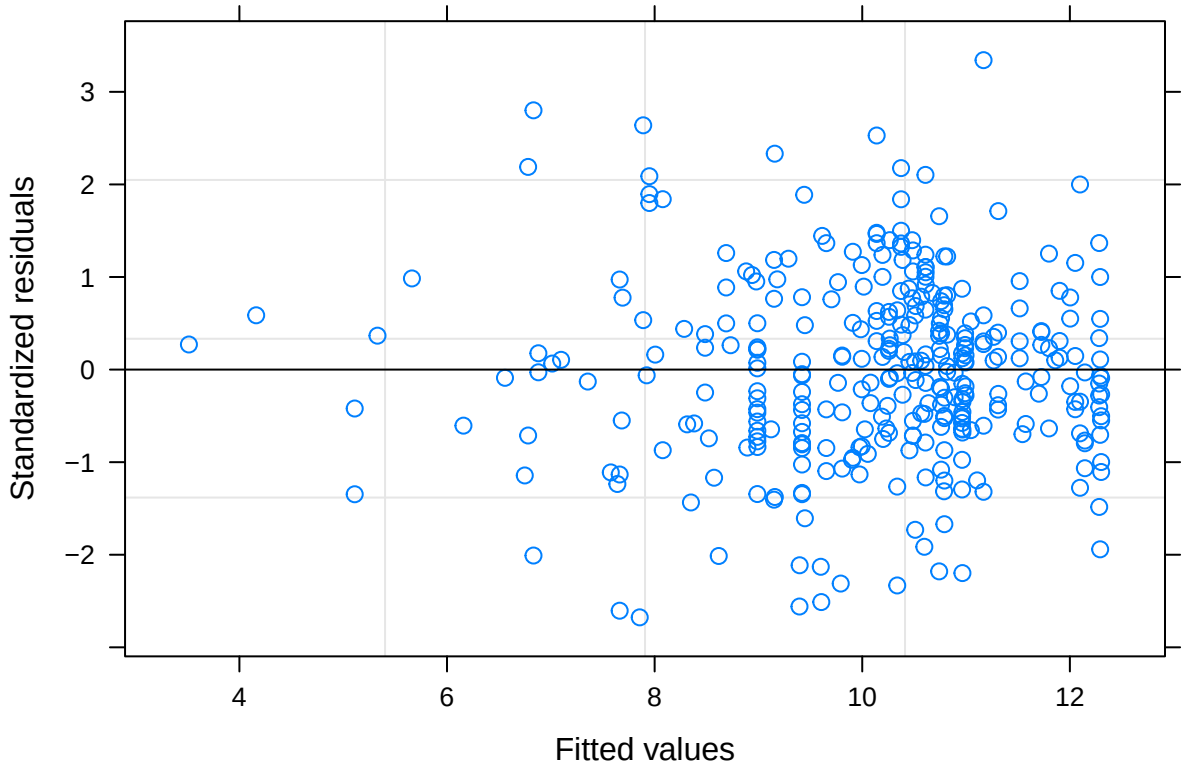
```
## Analysis of Deviance Table (Type III tests)
##
## Response: GrainYield_Mgha
##                    Chisq Df Pr(>Chisq)
## (Intercept)       9.3514  1   0.002228 **
## SI                7.2376  1   0.007139 **
## TopDress_kgha    23.6748  1  1.141e-06 ***
## I(SI * SI)        1.6319  1   0.201437
## SI:TopDress_kgha 20.7321  1  5.282e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
r.squaredGLMM(sUAS_ndvi_model)
```

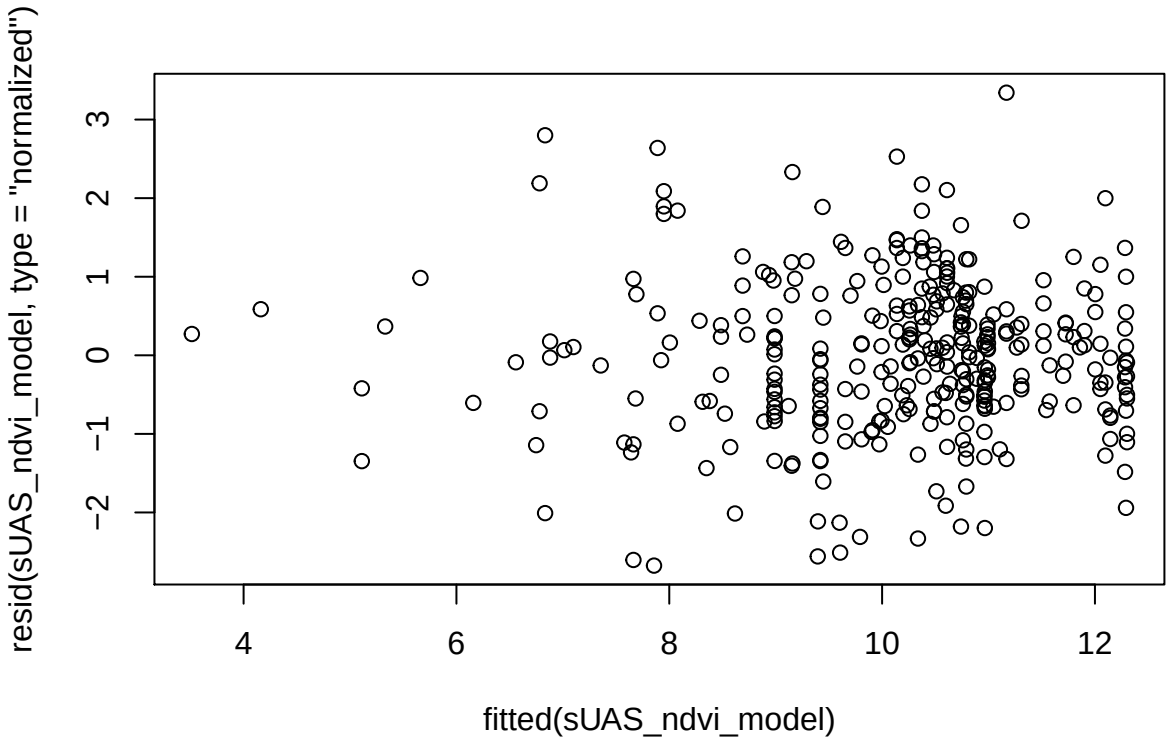
```
##           R2m       R2c
## [1,] 0.471899 0.7985167
```

**model diagnostics**

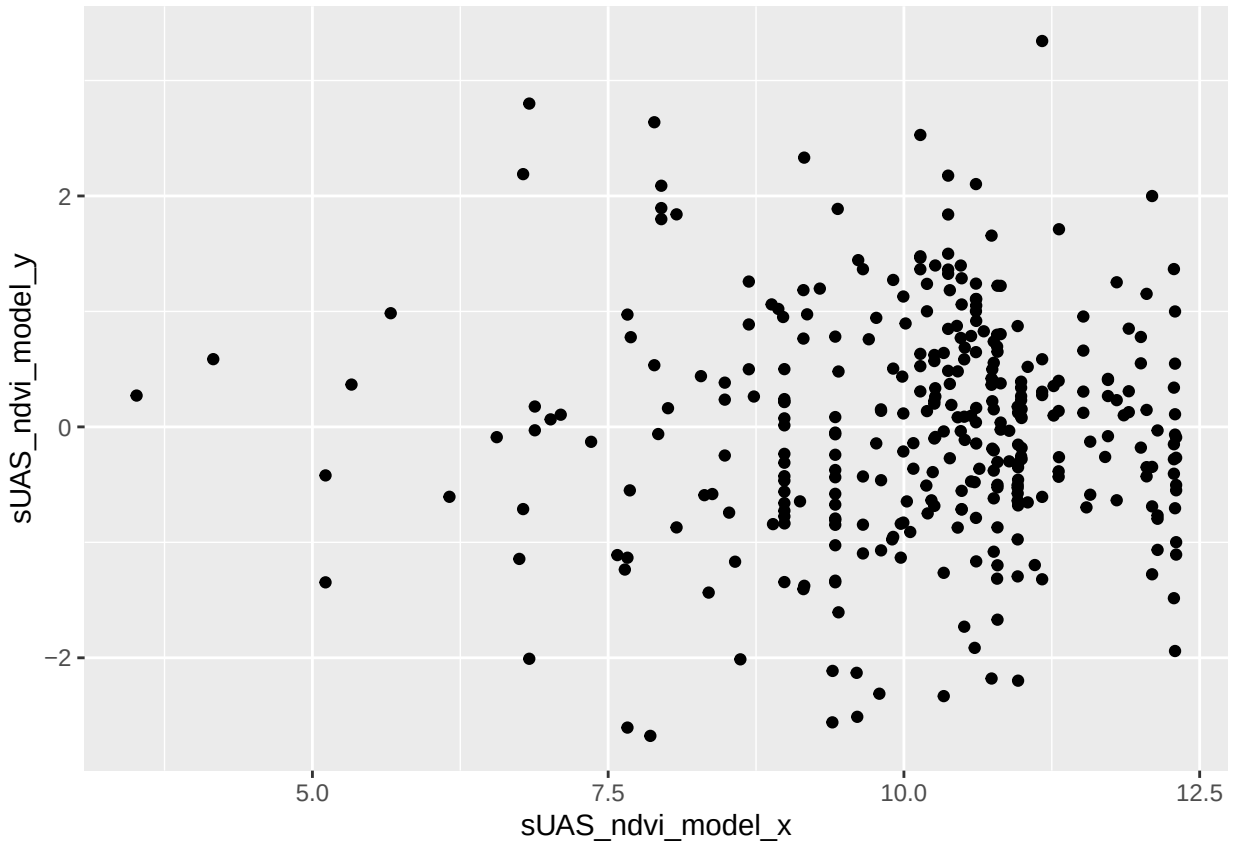
```
plot (sUAS_ndvi_model)
```



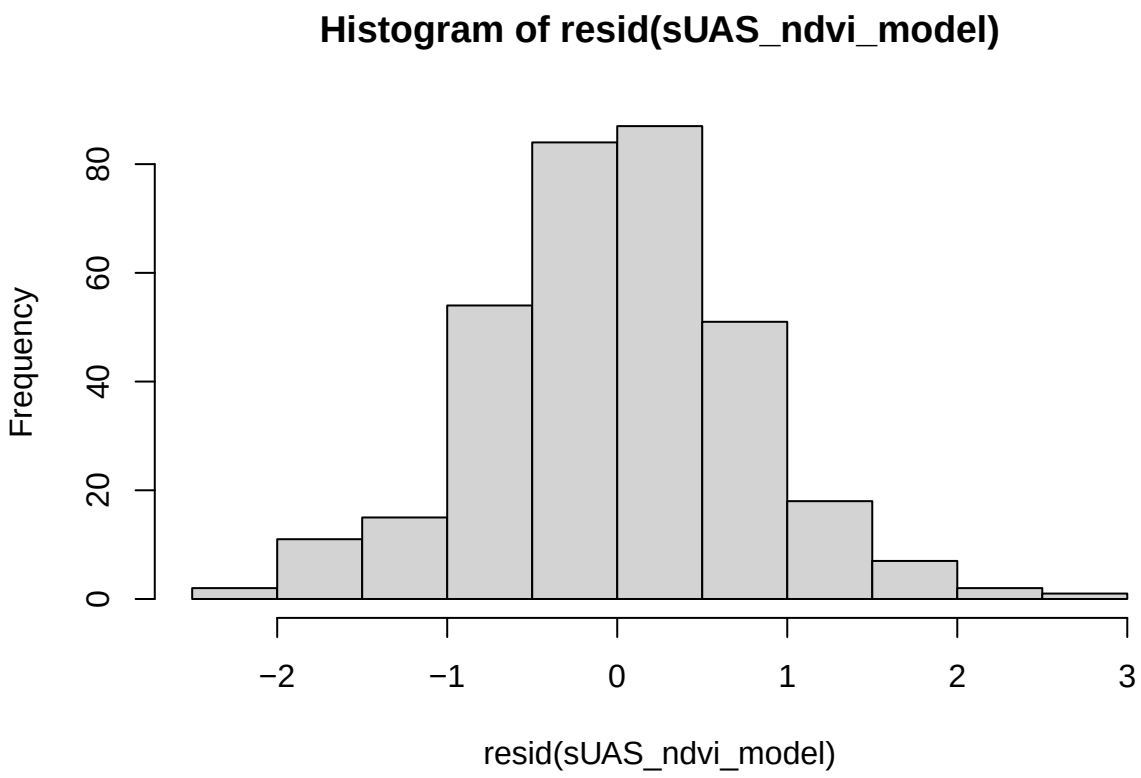

```
plot(resid(sUAS_ndvi_model, type = "normalized") ~fitted(sUAS_ndvi_model))
```



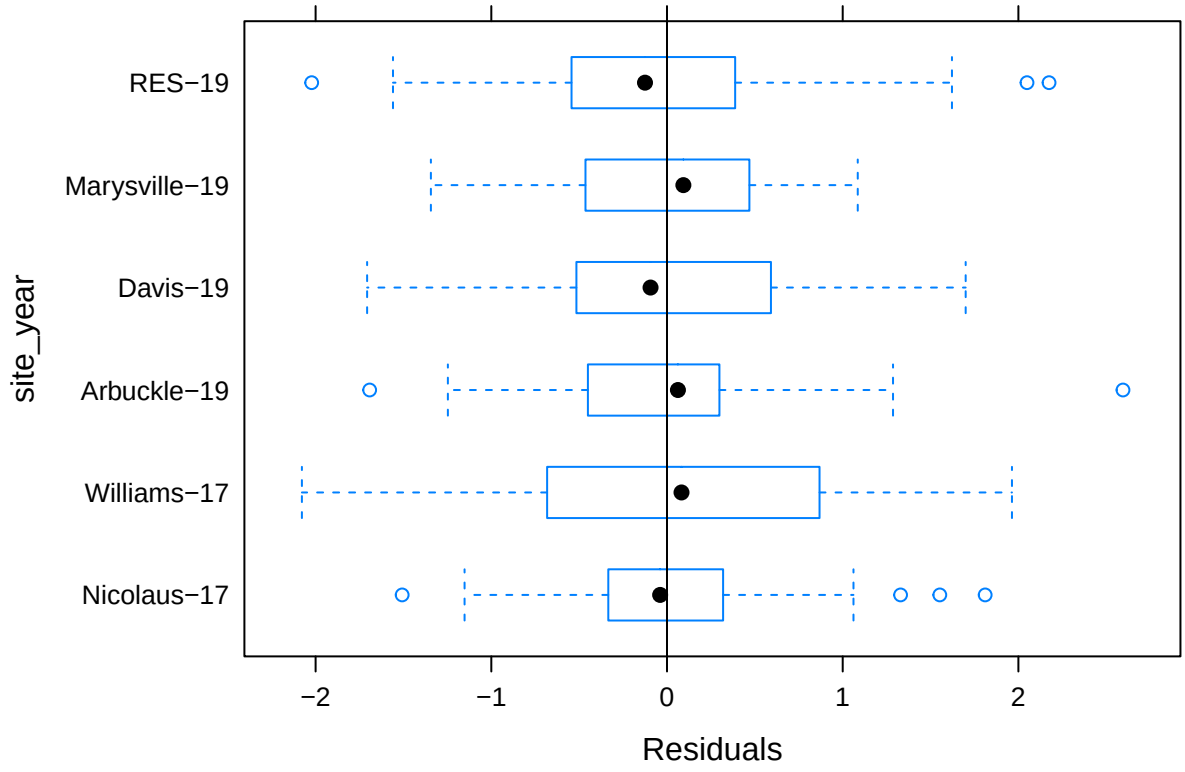

```
sUAS_ndvi_model_y <- resid(sUAS_ndvi_model, type = "normalized")
sUAS_ndvi_model_x <- fitted(sUAS_ndvi_model)

sUAS_ndvi_modelresid_data <- data.frame(sUAS_ndvi_model_x , sUAS_ndvi_model_y)

ggplot( data = sUAS_ndvi_modelresid_data , aes( x = sUAS_ndvi_model_x , y = sUAS_ndvi_model_y)) +
  geom_point(mapping = aes(sUAS_ndvi_model_x , sUAS_ndvi_model_y) , data = sUAS_ndvi_modelresid_data)
```



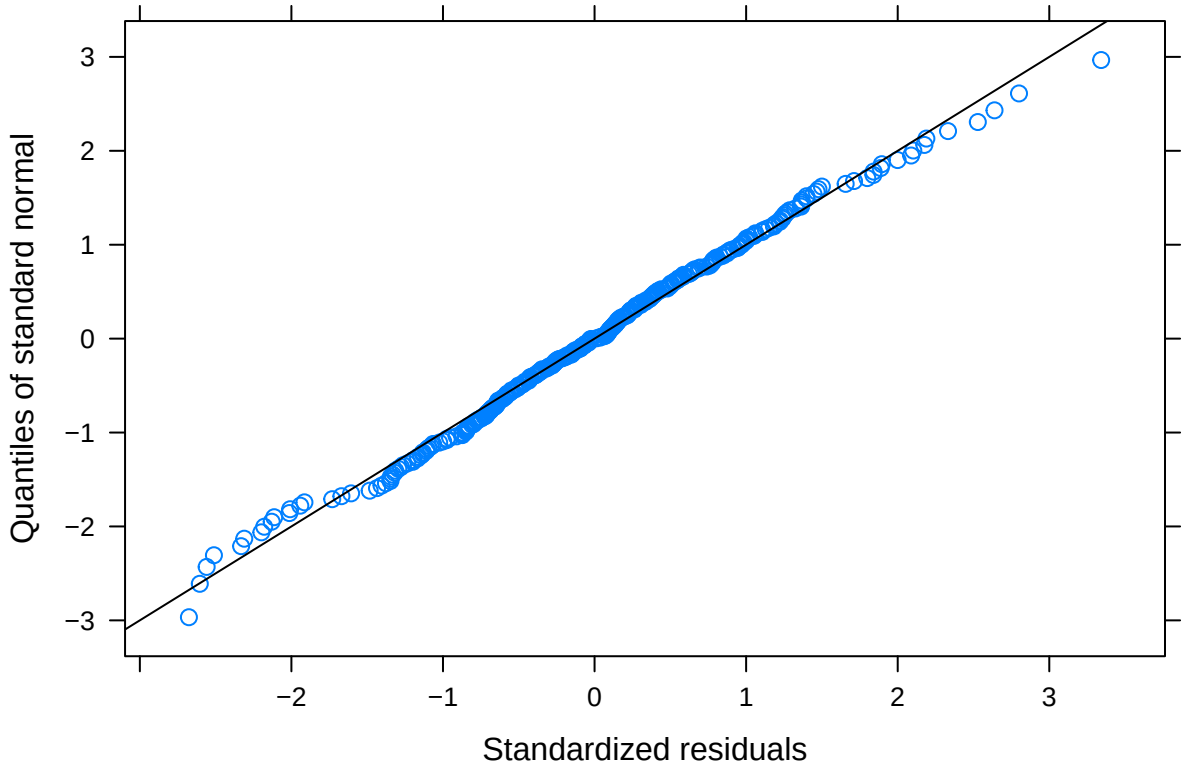

```
hist(resid(sUAS_ndvi_model))
```



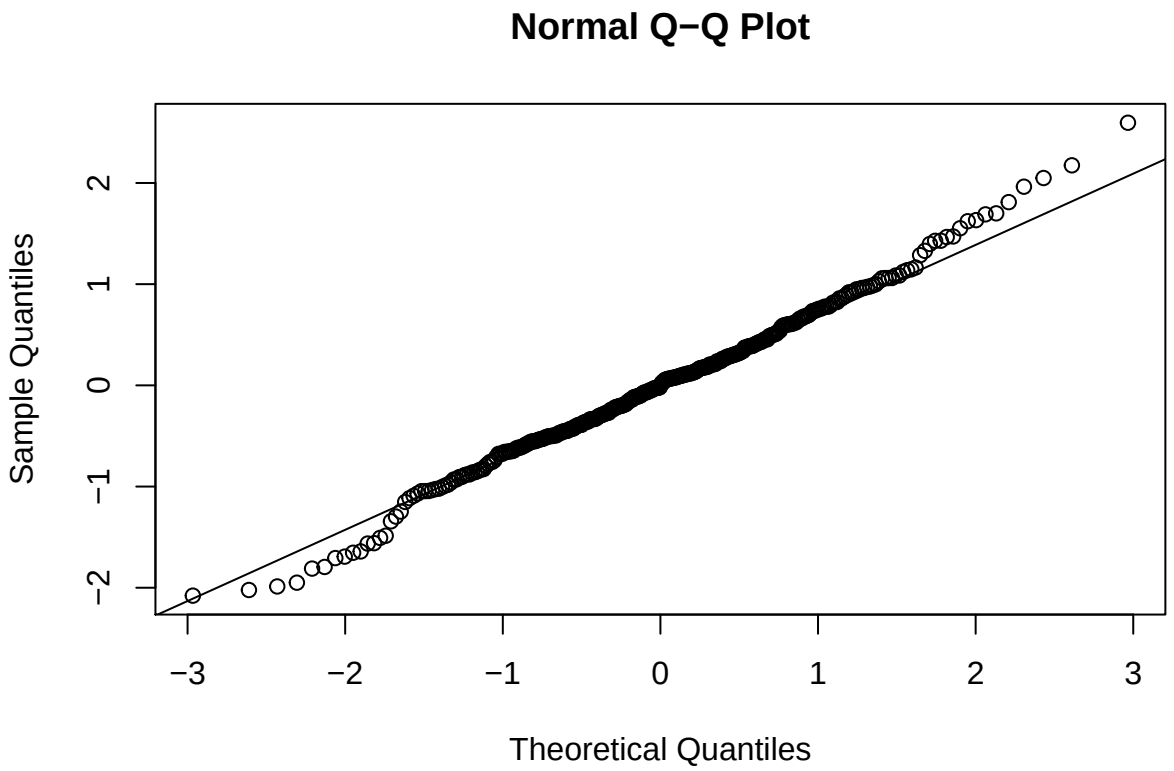
```
plot(sUAS_ndvi_model, site_year ~ resid(.), abline = 0)
```



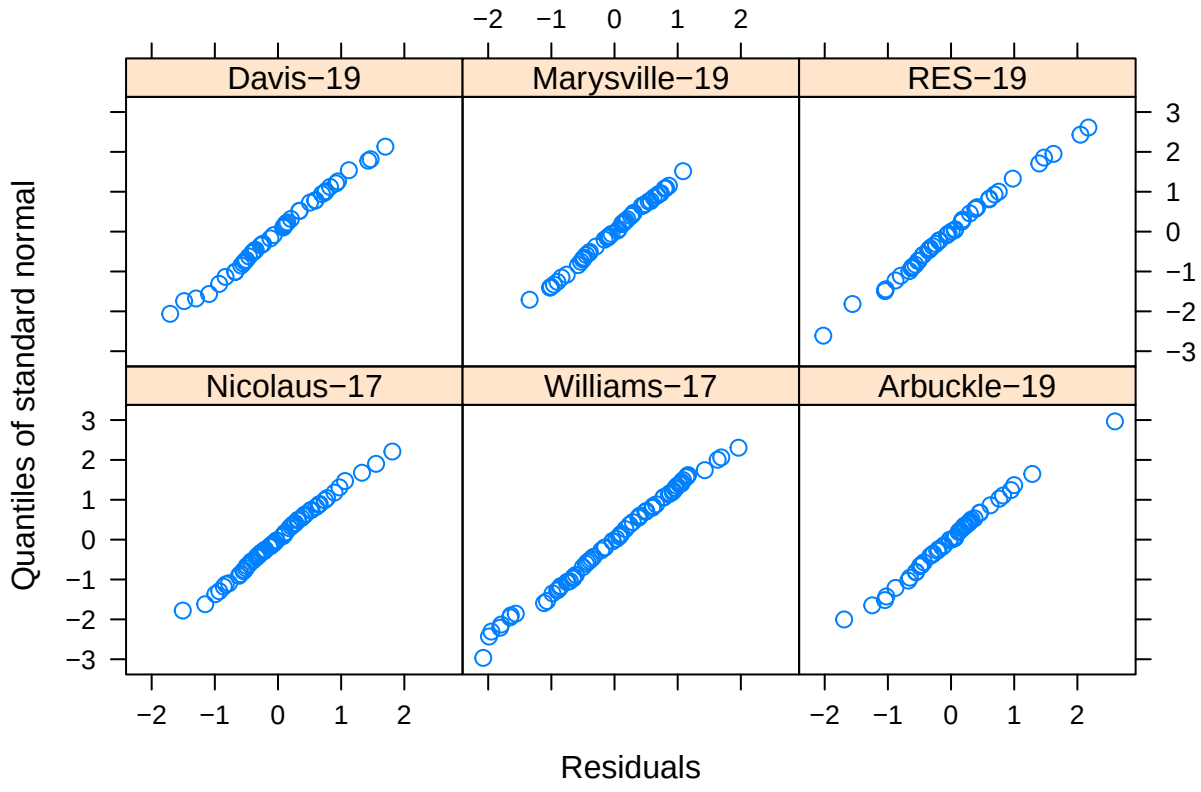

```
qqnorm(sUAS_ndvi_model, abline = c(0,1) )
```



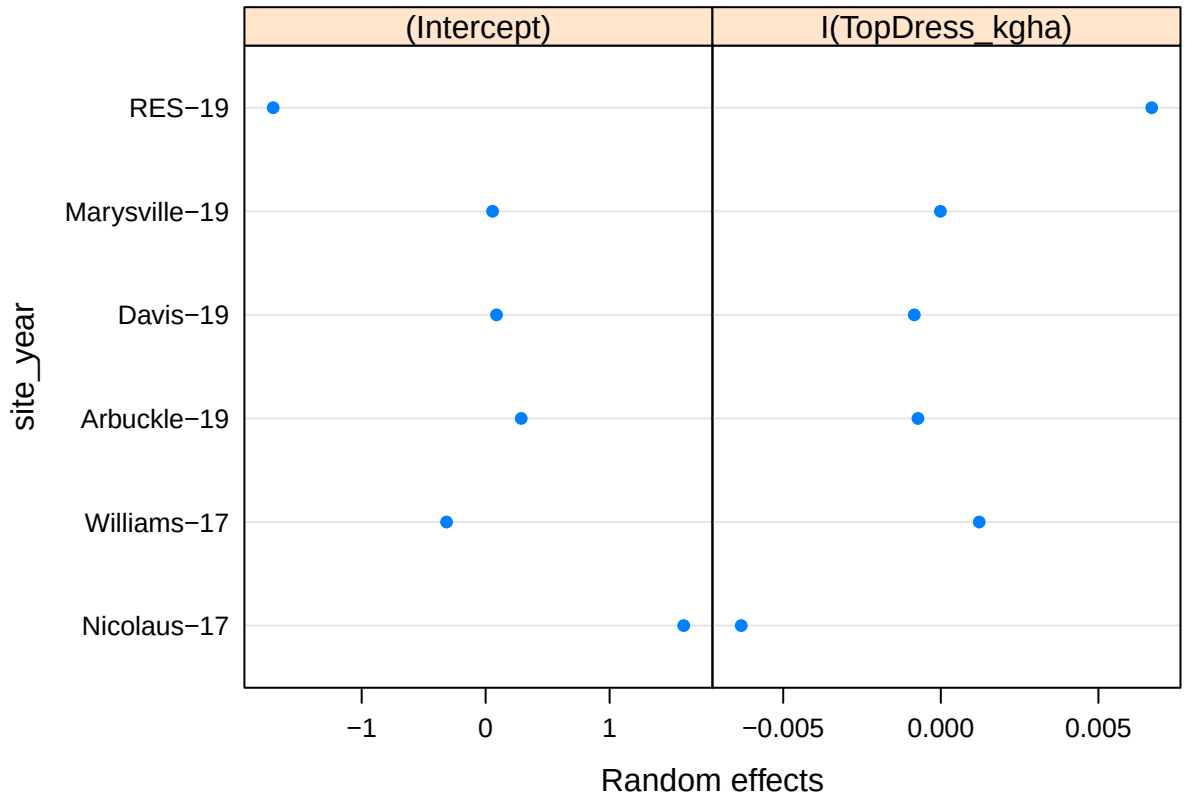
```
qqnorm(resid(sUAS_ndvi_model))
qqline(resid(sUAS_ndvi_model))
```



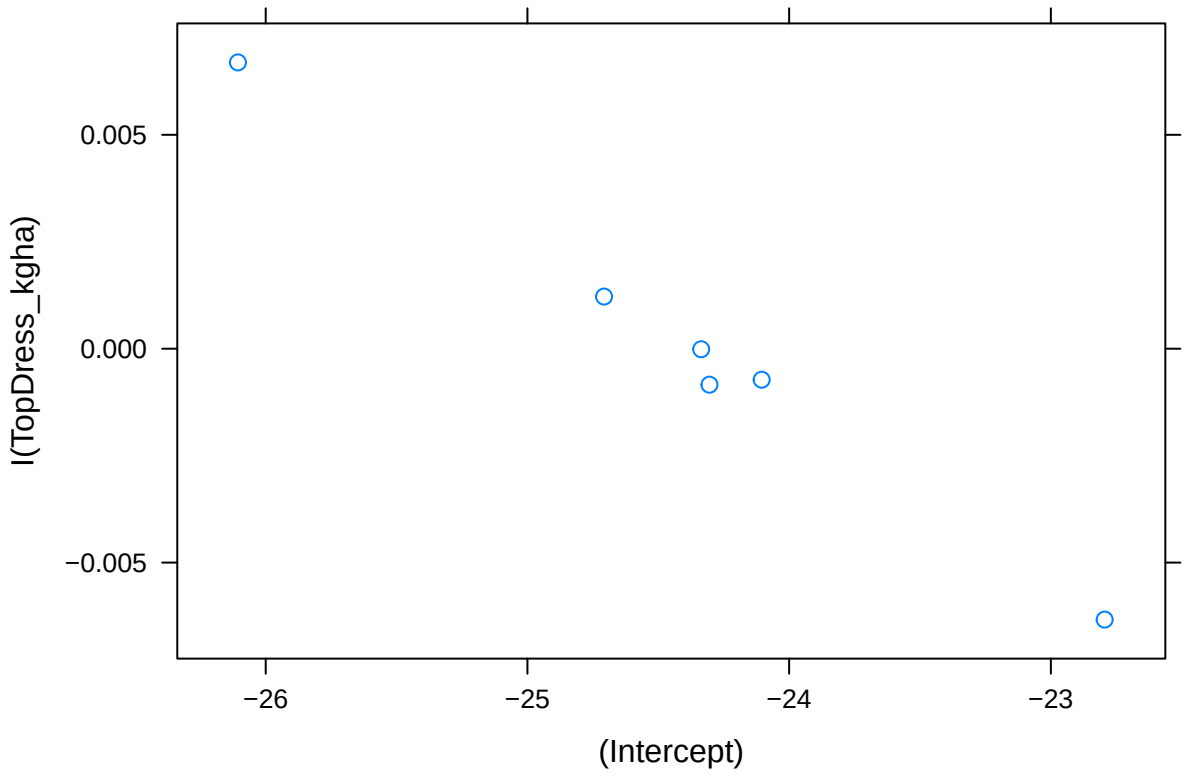

```
qqnorm(sUAS_ndvi_model , ~resid(.) | site_year)
```



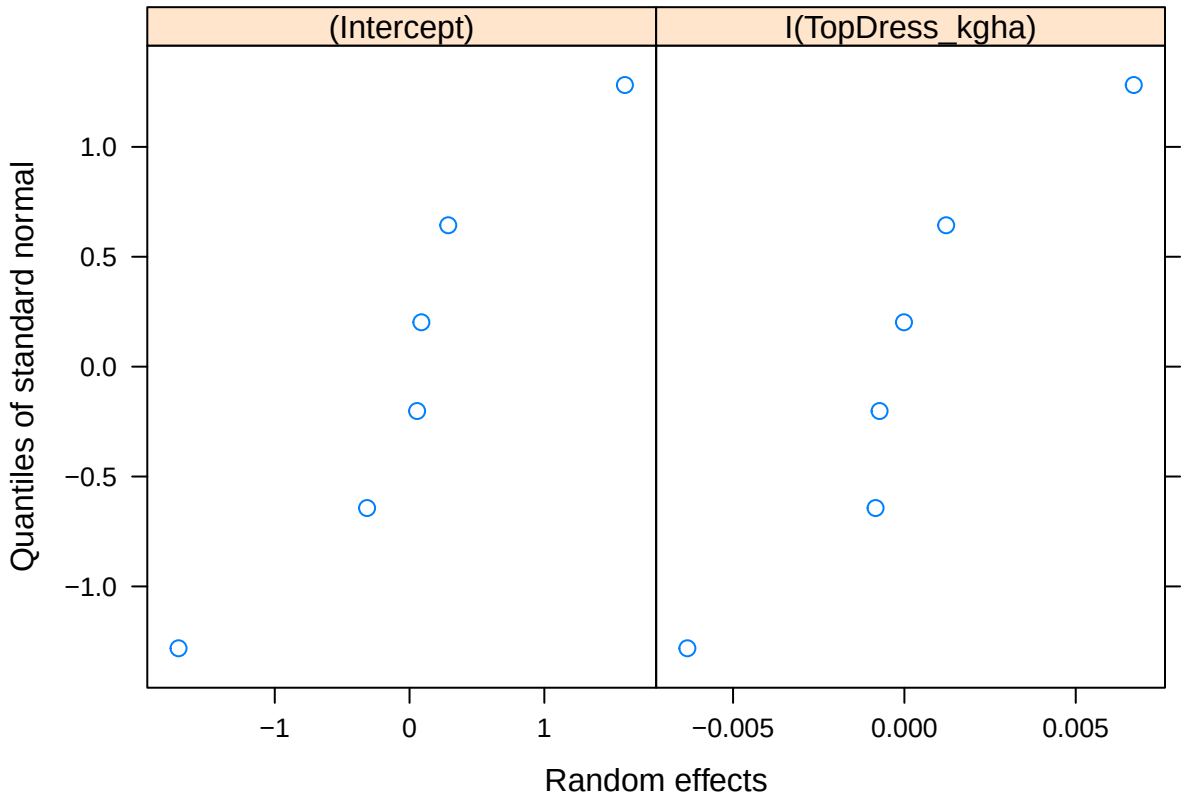
```
plot(ranef(sUAS_ndvi_model))
```



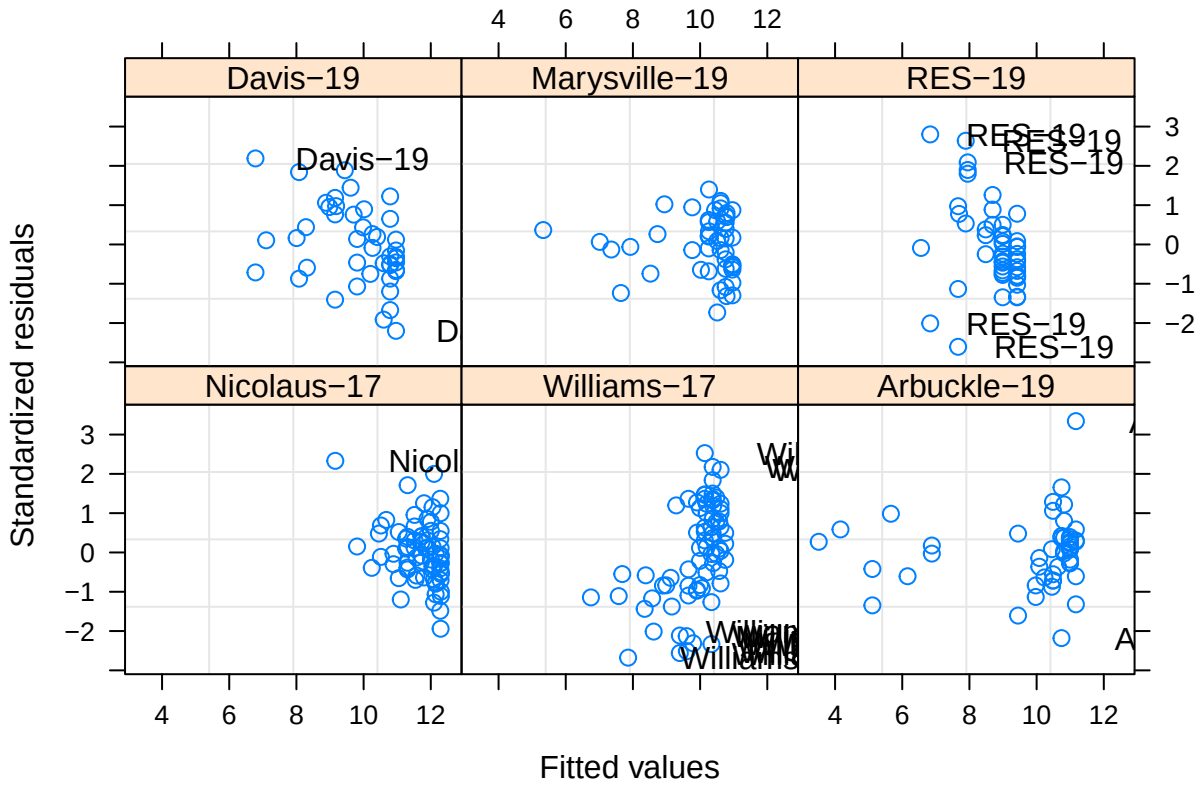

```
pairs(sUAS_ndvi_model , id = 0.1)
```



```
qqnorm(sUAS_ndvi_model , ~ranef(.))
```




```
plot( sUAS_ndvi_model, resid(., type = "p") ~ fitted(.) | site_year,
      id = 0.05, adj = -0.3 )
```




###outlier removal

```
sUAS_ndvi_data_OR <- sUAS_ndvi_data[-(c(which(abs(residuals(sUAS_ndvi_model,type="normalized"))>qnorm(0
```

###rerun model sans outliers

```
ctrl <- lmeControl(opt = "optim") #changes control to "optimal" settings

sUAS_ndvi_model_OR <- lme(GrainYield_Mgha ~ SI * TopDress_kgha + I(SI*SI)  ,
             control = ctrl ,
             random = ~ I(TopDress_kgha) | site_year  ,
             data = sUAS_ndvi_data)

summary(sUAS_ndvi_model_OR)
```

```
## Linear mixed-effects model fit by REML
##   Data: sUAS_ndvi_data
##       AIC      BIC    logLik
##   821.447 855.5566 -401.7235
##
## Random effects:
##  Formula: ~I(TopDress_kgha) | site_year
##  Structure: General positive-definite, Log-Cholesky parametrization
##                   StdDev      Corr
## (Intercept)      1.071047781 (Intr)
## I(TopDress_kgha) 0.004528693 -0.935
## Residual         0.776505570
##
## Fixed effects:  GrainYield_Mgha ~ SI * TopDress_kgha + I(SI * SI)
##                       Value Std.Error  DF   t-value p-value
```

```
## (Intercept)      -24.39218  7.976495 322 -3.058007  0.0024
## SI                47.63446 17.706195 322  2.690271  0.0075
## TopDress_kgha      0.20859  0.042869 322  4.865671  0.0000
## I(SI * SI)       -12.53880  9.815347 322 -1.277468  0.2024
## SI:TopDress_kgha  -0.20263  0.044502 322 -4.553251  0.0000
##  Correlation:
##                  (Intr) SI     TpDrs_ I(SI*S
## SI               -0.996
## TopDress_kgha    -0.025 -0.031
## I(SI * SI)        0.989 -0.998  0.078
## SI:TopDress_kgha  0.023  0.031 -0.998 -0.078
##
## Standardized Within-Group Residuals:
##          Min           Q1          Med           Q3          Max
## -2.676205354 -0.636885362 -0.005202893  0.586056673  3.342143075
##
## Number of Observations: 332
## Number of Groups: 6
```

```
summary(sUAS_ndvi_model_OR)$tTable
```

```
##                       Value   Std.Error  DF   t-value      p-value
## (Intercept)      -24.3921810  7.97649532 322 -3.058007 2.414970e-03
## SI                47.6344646 17.70619487 322  2.690271 7.511739e-03
## TopDress_kgha      0.2085855  0.04286881 322  4.865671 1.788676e-06
## I(SI * SI)       -12.5387959  9.81534736 322 -1.277468 2.023572e-01
## SI:TopDress_kgha  -0.2026276  0.04450175 322 -4.553251 7.501114e-06
```

```
Anova(sUAS_ndvi_model_OR , type = 3)
```

```
## Analysis of Deviance Table (Type III tests)
##
## Response: GrainYield_Mgha
##                     Chisq Df Pr(>Chisq)
## (Intercept)        9.3514  1   0.002228 **
## SI                 7.2376  1   0.007139 **
## TopDress_kgha     23.6748  1  1.141e-06 ***
## I(SI * SI)         1.6319  1   0.201437
## SI:TopDress_kgha 20.7321  1  5.282e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
r.squaredGLMM(sUAS_ndvi_model_OR)
```

```
##           R2m       R2c
## [1,] 0.471899 0.7985167
```

```
ndvi_r_sq <- r.squaredGLMM(sUAS_ndvi_model_OR)

ndvi_r_sq_fixed <- round(ndvi_r_sq[1] , digits = 2)
ndvi_r_sq_fixed
```

```
## [1] 0.47
```

```
ndvi_r_sq_total <- round(ndvi_r_sq[2] , digits = 2)
ndvi_r_sq_total
```

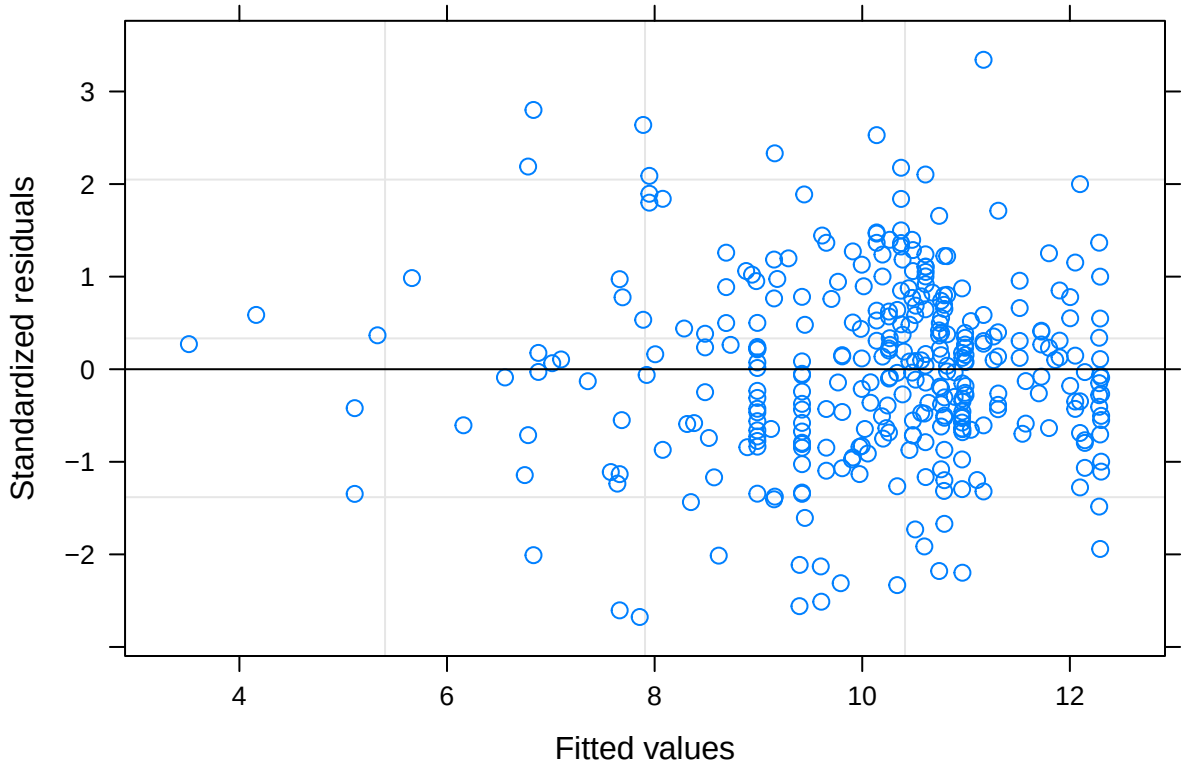
```
## [1] 0.8
```

```
ndvi_r_sq_random <- ndvi_r_sq_total - ndvi_r_sq_fixed
ndvi_r_sq_random
```

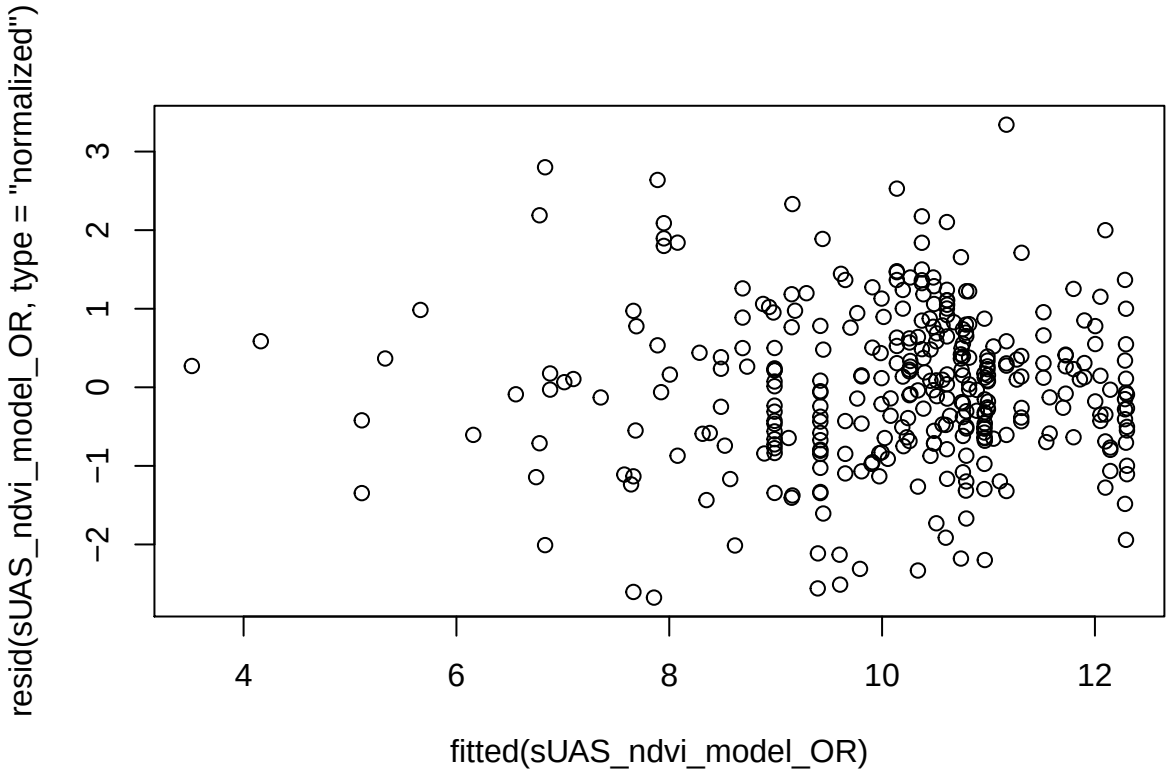
```
## [1] 0.33
```

### model diagnostics

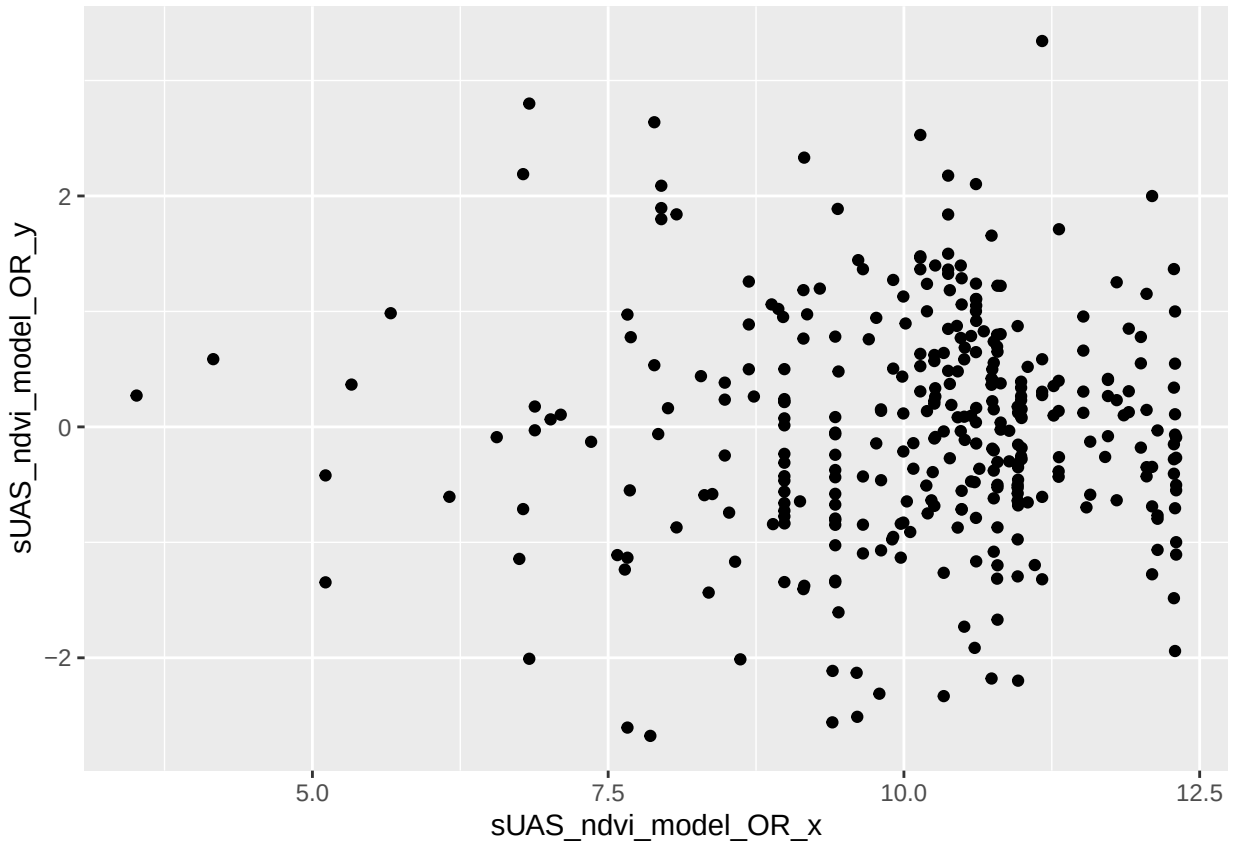
```
plot (sUAS_ndvi_model_OR)
```



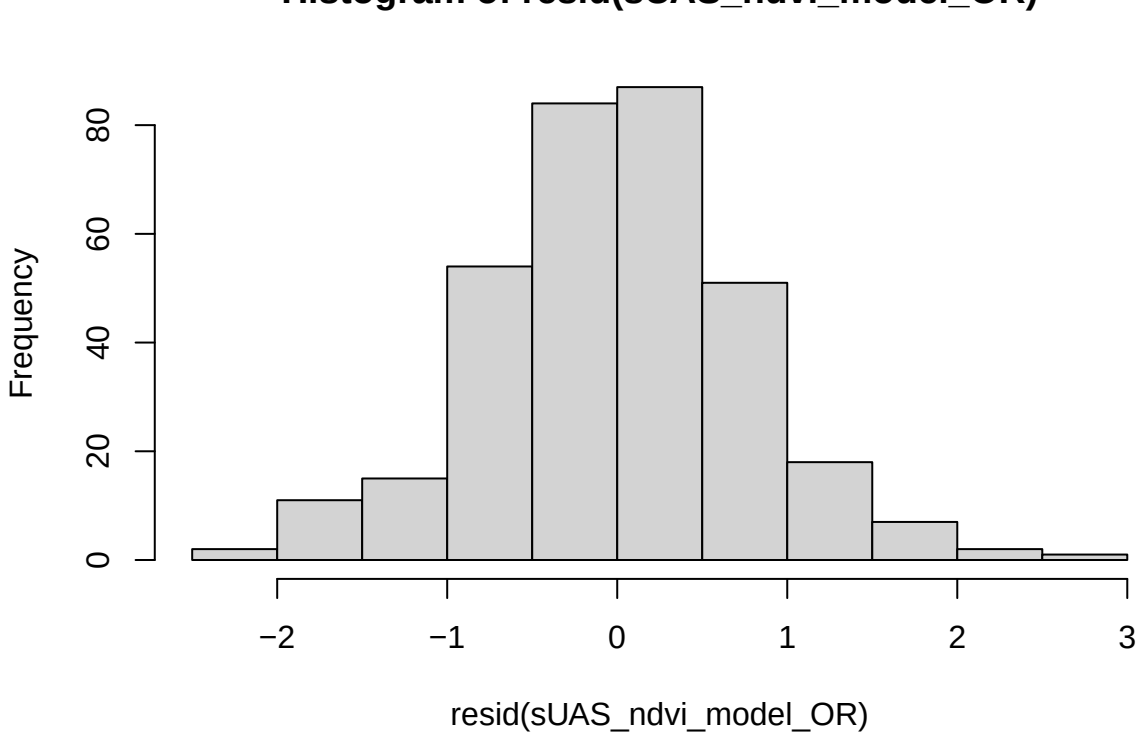

```
plot(resid(sUAS_ndvi_model_OR, type = "normalized") ~fitted(sUAS_ndvi_model_OR))
```



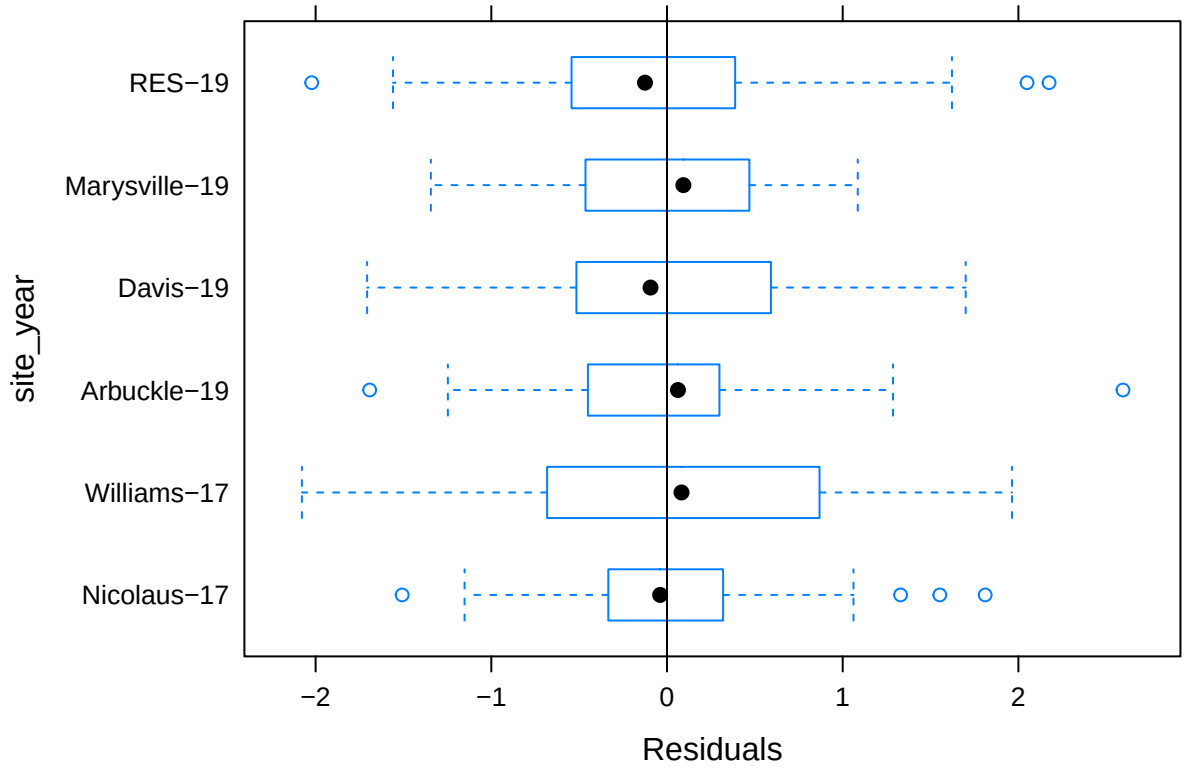

```
sUAS_ndvi_model_OR_y <- resid(sUAS_ndvi_model_OR, type = "normalized")
sUAS_ndvi_model_OR_x <- fitted(sUAS_ndvi_model_OR)

sUAS_ndvi_model_ORresid_data <- data.frame(sUAS_ndvi_model_OR_x , sUAS_ndvi_model_OR_y)

ggplot( data = sUAS_ndvi_model_ORresid_data , aes( x = sUAS_ndvi_model_OR_x , y = sUAS_ndvi_model_OR_y)
  geom_point(mapping = aes(sUAS_ndvi_model_OR_x , sUAS_ndvi_model_OR_y) , data = sUAS_ndvi_model_ORresi
```



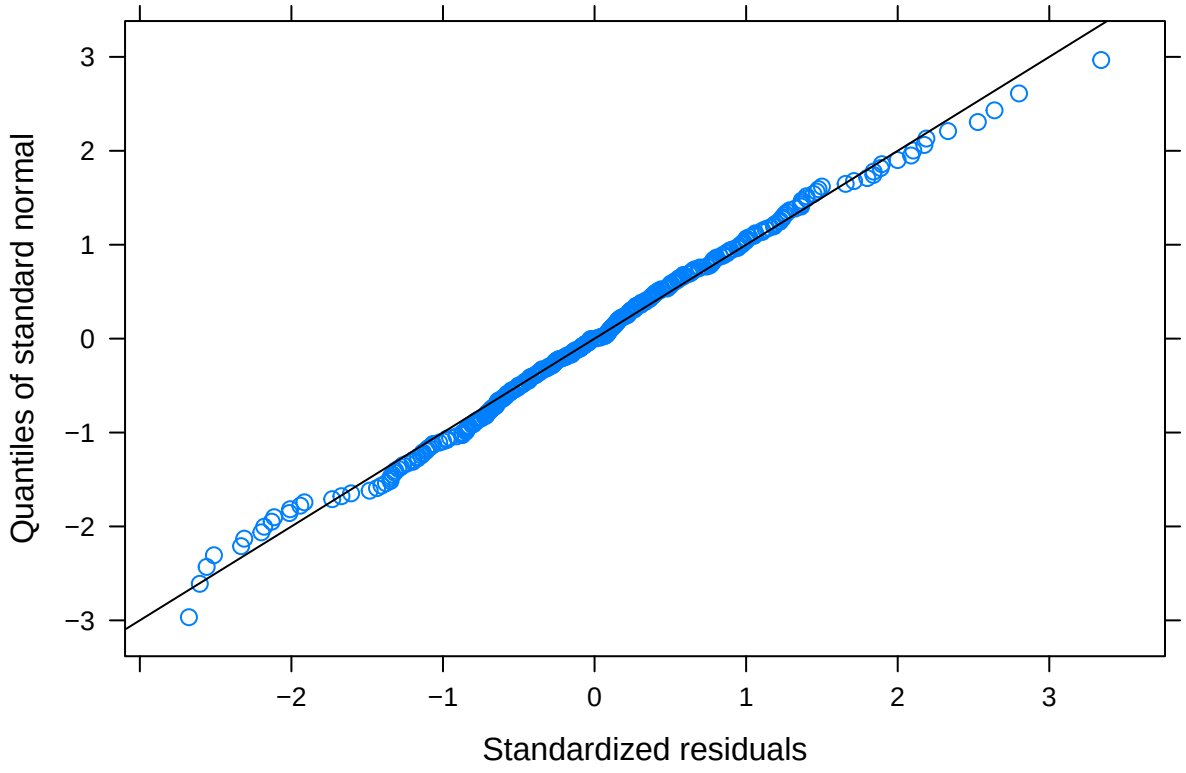

```
hist(resid(sUAS_ndvi_model_OR))
```



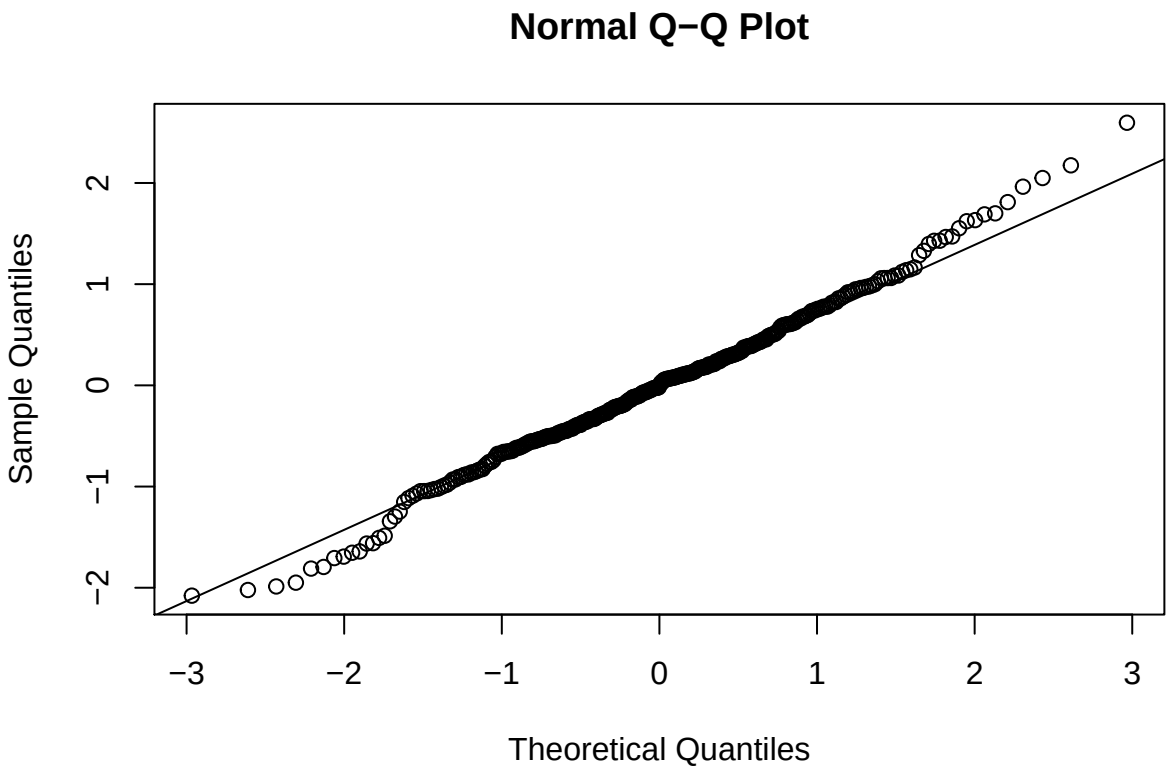
```
plot(sUAS_ndvi_model_OR, site_year ~ resid(.), abline = 0)
```



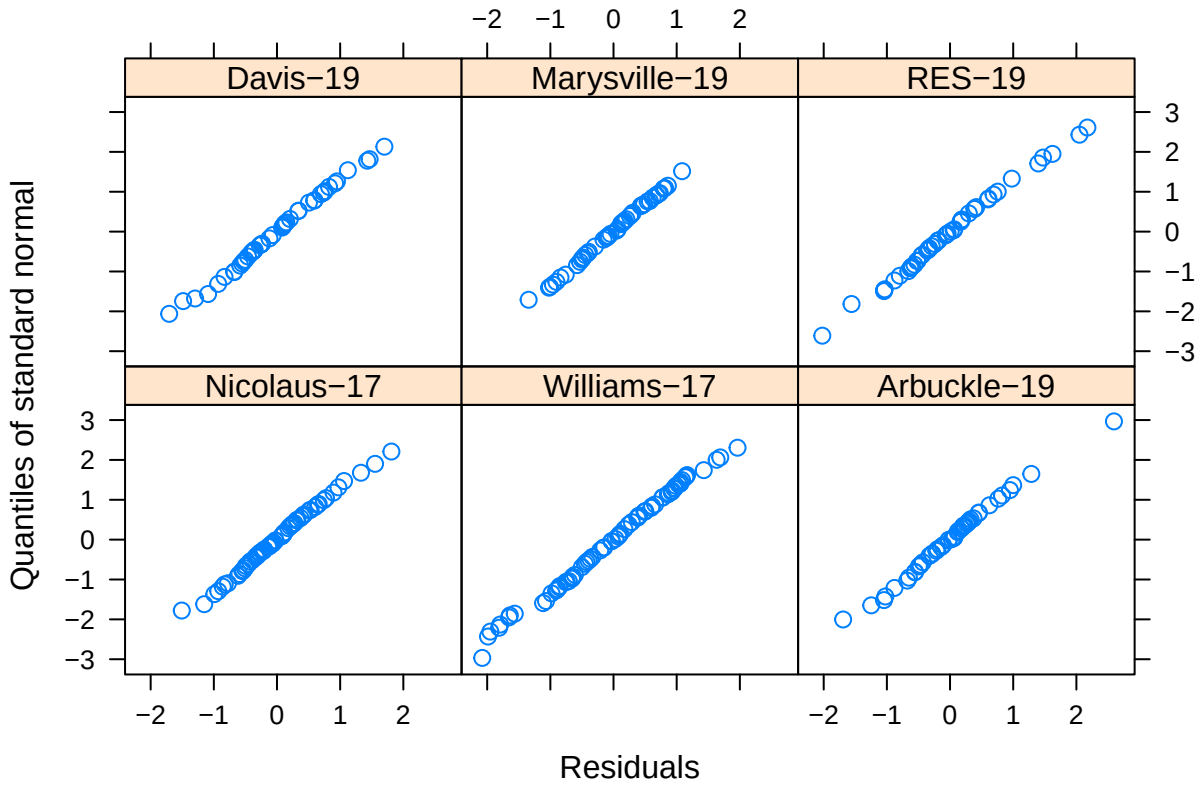

```
qqnorm(sUAS_ndvi_model_OR, abline = c(0,1) )
```



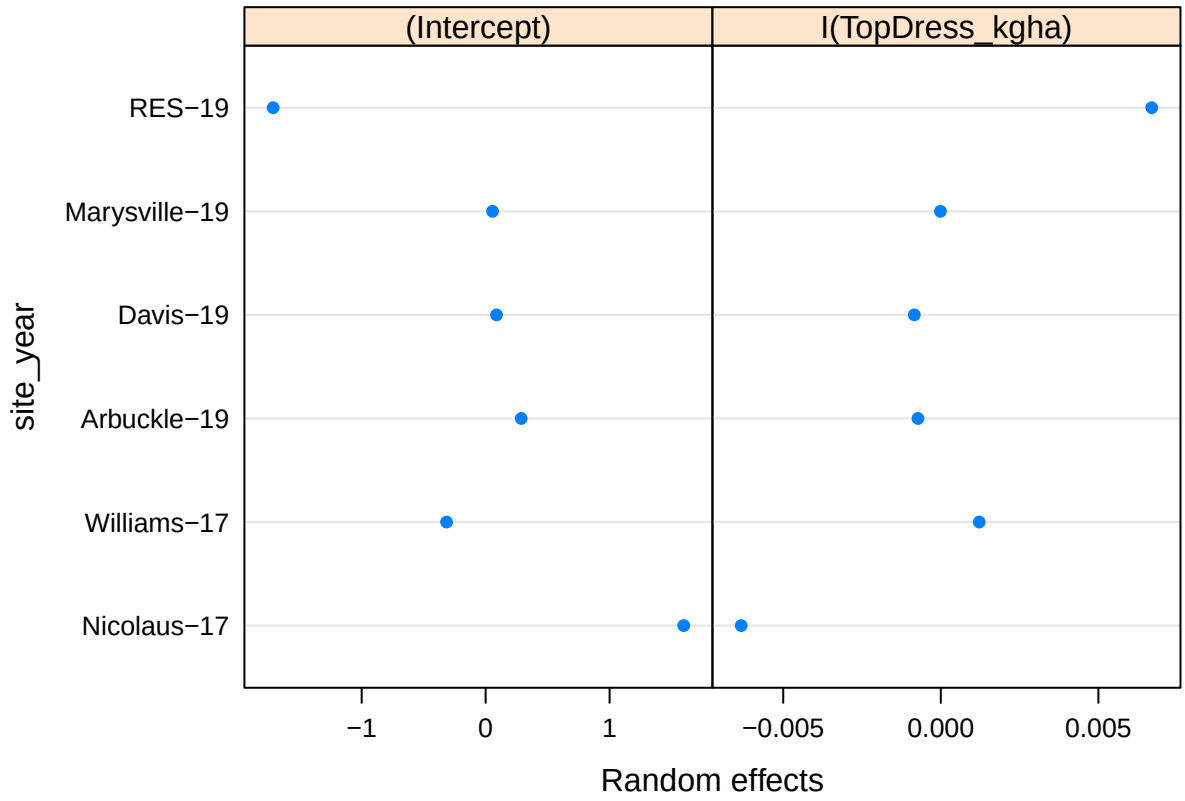
```
qqnorm(resid(sUAS_ndvi_model_OR))
qqline(resid(sUAS_ndvi_model_OR))
```



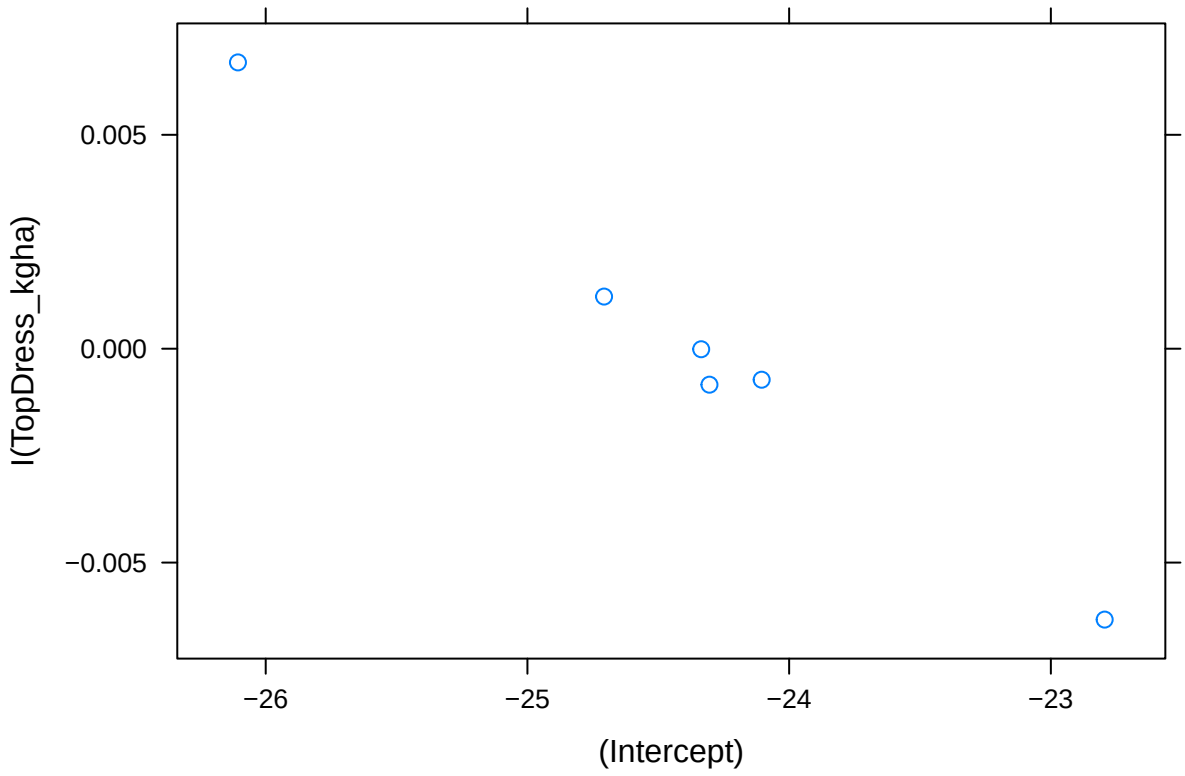

```
qqnorm(sUAS_ndvi_model_OR , ~resid(.) | site_year)
```



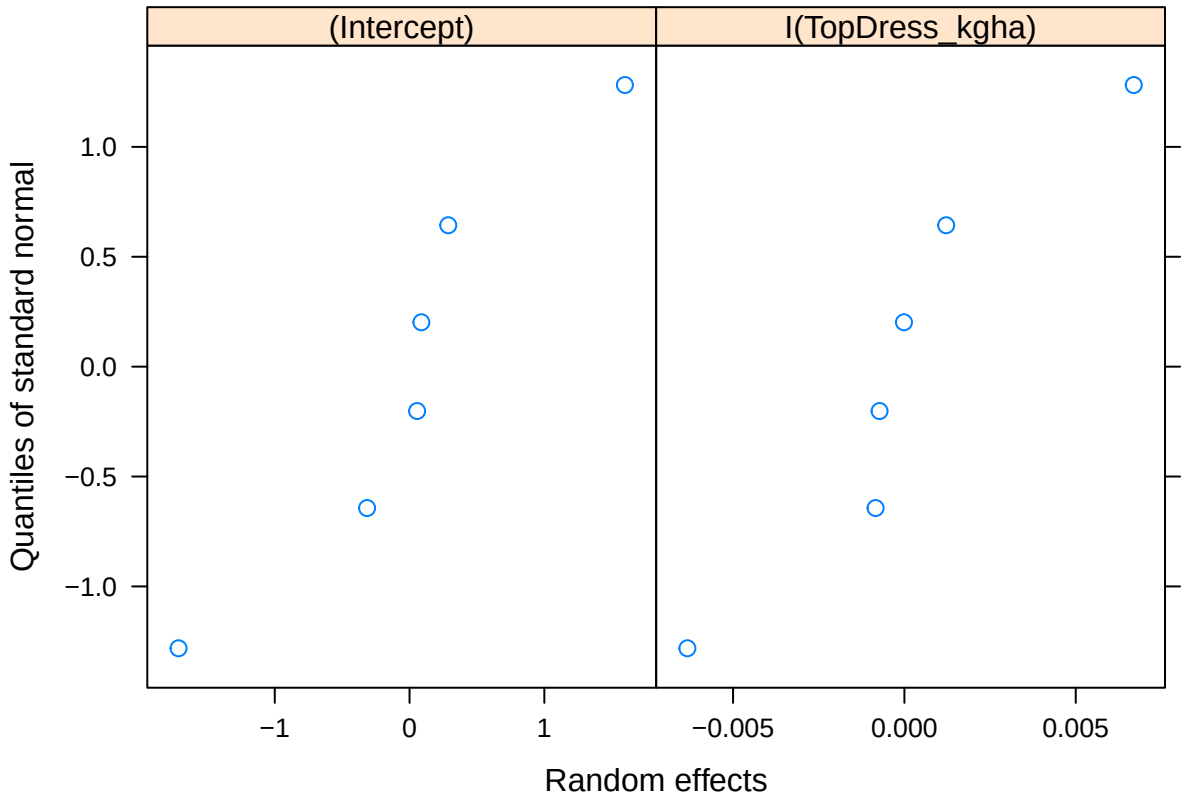
```
plot(ranef(sUAS_ndvi_model_OR))
```



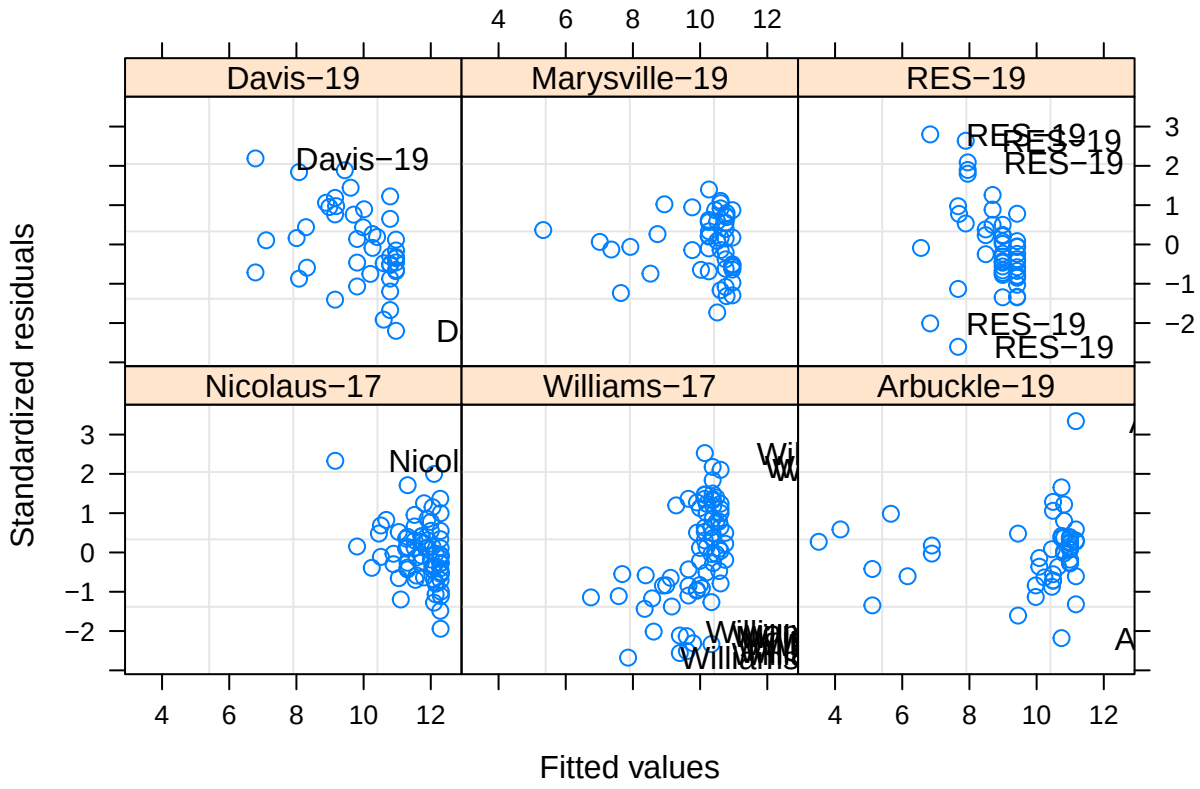

```
pairs(sUAS_ndvi_model_OR , id = 0.1)
```



```
qqnorm(sUAS_ndvi_model_OR , ~ranef(.))
```




```
plot( sUAS_ndvi_model_OR, resid(., type = "p") ~ fitted(.) | site_year,
      id = 0.05, adj = -0.3 )
```




###emmeans

```
mylist <- list(SI=seq(0.708 , 1 , by = .0001),TopDress_kgha =c( 34 , 0))

sUAS_ndvi_emmeans <- emmeans(sUAS_ndvi_model_OR , ~TopDress_kgha * SI , at = mylist )

sUAS_ndvi_emmeans_contrast <- as.data.frame(summary(contrast(sUAS_ndvi_emmeans , "pairwise" , side = ">

sUAS_ndvi_emmeans_contrast <- sUAS_ndvi_emmeans_contrast %>%
  mutate(sUAS_ndvi_Response_Index = 1 / SI )

sUAS_ndvi_emmeans_contrast <- sUAS_ndvi_emmeans_contrast %>%
  mutate(t_score = abs((estimate - 0.26)) / SE)

sUAS_ndvi_emmeans_contrast <- sUAS_ndvi_emmeans_contrast %>%
  mutate(prob_greater_than_26 = if_else(estimate < 0.26 , pt(q = t_score , df = df , lower.tail = F) , p

sUAS_ndvi_emmeans_contrast <- sUAS_ndvi_emmeans_contrast %>%
  mutate(prob_diff_from_26 = 1 - (pt(q = t_score , df = df , lower.tail = F) + pt(q = -t_score , df = df

sUAS_ndvi_emmeans_contrast$sUAS_ndvi_Response_Index_r <- (round(sUAS_ndvi_emmeans_contrast$sUAS_ndvi_Res

sUAS_ndvi_run1 <- 125 - 100
sUAS_ndvi_rise1 <- 1.596946 - 0.2038182
sUAS_ndvi_slope1 <- round((sUAS_ndvi_rise1 / sUAS_ndvi_run1) * 5 , digits = 2)
sUAS_ndvi_slope1
```

```
## [1] 0.28
```

```
sUAS_ndvi_run2 <- 142 - 125
sUAS_ndvi_rise2 <- 2.237784 - 1.596946
sUAS_ndvi_slope2 <- round((sUAS_ndvi_rise2 / sUAS_ndvi_run2) * 5 , digits = 3)
sUAS_ndvi_slope2
```

```
## [1] 0.188
```

```
sUAS_ndvi_mean_se <- sUAS_ndvi_emmeans_contrast %>%
  select(SE) %>%
  summarise(mean_se = mean(SE))

sUAS_ndvi_mean_se$mean_se <- round(sUAS_ndvi_mean_se$mean_se , digits = 2)

sUAS_ndvi_mean_se$mean_se
```

```
## [1] 0.21
```

###confidence interval

```
#at every 0.10 interval, what is the 90% confidence interval of the yield response, and what RI value d

str(sUAS_ndvi_emmeans_contrast , give.attr = F)
```

```
## 'data.frame':    2921 obs. of  12 variables:
##  $ contrast                 : Factor w/ 1 level "34 - 0": 1 1 1 1 1 1 1 1 1 1 ...
##  $ SI                       : num  0.708 0.708 0.708 0.708 0.708 ...
##  $ estimate                 : num  2.21 2.21 2.21 2.21 2.21 ...
##  $ SE                       : num  0.396 0.396 0.395 0.395 0.395 ...
##  $ df                       : num  322 322 322 322 322 322 322 322 322 322 ...
##  $ t.ratio                  : num  5.59 5.6 5.6 5.6 5.6 ...
##  $ p.value                  : num  2.36e-08 2.36e-08 2.35e-08 2.35e-08 2.35e-08 ...
##  $ sUAS_ndvi_Response_Index  : num  1.41 1.41 1.41 1.41 1.41 ...
##  $ t_score                  : num  4.94 4.94 4.94 4.94 4.94 ...
##  $ prob_greater_than_26     : num  1 1 1 1 1 ...
##  $ prob_diff_from_26        : num  1 1 1 1 1 ...
##  $ sUAS_ndvi_Response_Index_r: num  1.41 1.41 1.41 1.41 1.41 ...
```

```
sUAS_ndvi_emmeans_contrast <- sUAS_ndvi_emmeans_contrast %>%
  mutate(response_mgha_r = round(estimate , digits = 2),
         se_mgha_r = round(SE , digits = 3))

#estimate = 0.20 , SE = 0.114

print(round(0.20 - z_score*0.114 , digits = 2))
```

```
## [1] 0.01
```

```
print(round(0.20 + z_score*0.114 , digits = 2))
```

```
## [1] 0.39
```

```
#estimate = 0.30 , SE = 0.104

print(round(0.30 - z_score*0.106 , digits = 2))
```

```
## [1] 0.13
```

```r
print(round(0.30 + z_score*0.106 , digits = 2))
```

```
## [1] 0.47
```

```r
#estimate = 0.40 , SE = 0.100

print(round(0.40 - z_score*0.100 , digits = 2))
```

```
## [1] 0.24
```

```r
print(round(0.40 + z_score*0.100 , digits = 2))
```

```
## [1] 0.56
```

```r
#estimate = 0.50 , SE = 0.099

print(round(0.50 - z_score*0.099 , digits = 2))
```

```
## [1] 0.34
```

```r
print(round(0.50 + z_score*0.099 , digits = 2))
```

```
## [1] 0.66
```

##sUAS NDRE-RI

```r
sUAS_ndre_data <- paper3_data_no_2018 %>%
  filter(Platform == "sUAS_NDRE")

ctrl <- lmeControl(opt = "optim") #changes control to "optimal" settings

sUAS_ndre_model <- lme(GrainYield_Mgha ~ SI * TopDress_kgha + I(SI*SI)  ,
             control = ctrl ,
             random = ~ I(TopDress_kgha) | site_year  ,
             data = sUAS_ndre_data)

summary(sUAS_ndre_model)
```

```
## Linear mixed-effects model fit by REML
##   Data: sUAS_ndre_data
##        AIC      BIC    logLik
##   784.3945 818.5041 -383.1972
##
## Random effects:
##  Formula: ~I(TopDress_kgha) | site_year
##  Structure: General positive-definite, Log-Cholesky parametrization
##                  StdDev      Corr
## (Intercept)      1.025351819 (Intr)
## I(TopDress_kgha) 0.003962301 -0.919
## Residual         0.726916252
##
## Fixed effects:  GrainYield_Mgha ~ SI * TopDress_kgha + I(SI * SI)
##                       Value Std.Error  DF   t-value p-value
## (Intercept)      -11.45611  1.717014 322 -6.672109       0
## SI                39.19829  4.123213 322  9.506734       0
## TopDress_kgha      0.10343  0.016790 322  6.160088       0
## I(SI * SI)       -16.99240  2.527712 322 -6.722443       0
## SI:TopDress_kgha  -0.10031  0.018537 322 -5.411511       0
```

```
##  Correlation:
##                   (Intr) SI     TpDrs_ I(SI*S
## SI                -0.958
## TopDress_kgha     -0.138  0.011
## I(SI * SI)         0.934 -0.993  0.075
## SI:TopDress_kgha  0.115 -0.011 -0.987 -0.076
##
## Standardized Within-Group Residuals:
##          Min           Q1          Med           Q3          Max
## -2.607665461 -0.648795155 -0.006631185  0.600700113  3.701360082
##
## Number of Observations: 332
## Number of Groups: 6
```

```
summary(sUAS_ndre_model)$tTable
```

```
##                       Value  Std.Error  DF   t-value      p-value
## (Intercept)      -11.4561084 1.71701447 322 -6.672109 1.101242e-10
## SI                39.1982873 4.12321292 322  9.506734 4.730027e-19
## TopDress_kgha      0.1034275 0.01678994 322  6.160088 2.167408e-09
## I(SI * SI)       -16.9924007 2.52771225 322 -6.722443 8.140477e-11
## SI:TopDress_kgha  -0.1003116 0.01853671 322 -5.411511 1.222445e-07
```

```
Anova(sUAS_ndre_model , type = 3)
```

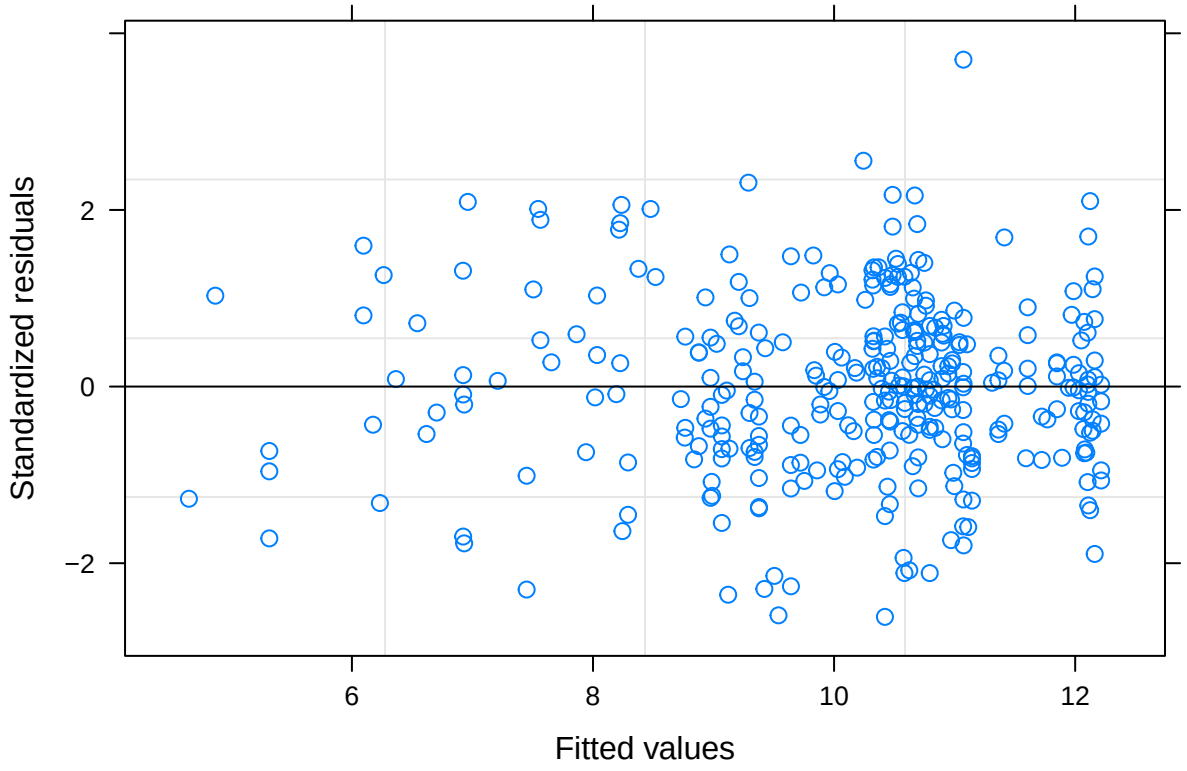
```
## Analysis of Deviance Table (Type III tests)
##
## Response: GrainYield_Mgha
##                    Chisq Df Pr(>Chisq)
## (Intercept)      44.517  1  2.522e-11 ***
## SI               90.378  1  < 2.2e-16 ***
## TopDress_kgha    37.947  1  7.270e-10 ***
## I(SI * SI)       45.191  1  1.787e-11 ***
## SI:TopDress_kgha 29.285  1  6.250e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
r.squaredGLMM(sUAS_ndre_model)
```

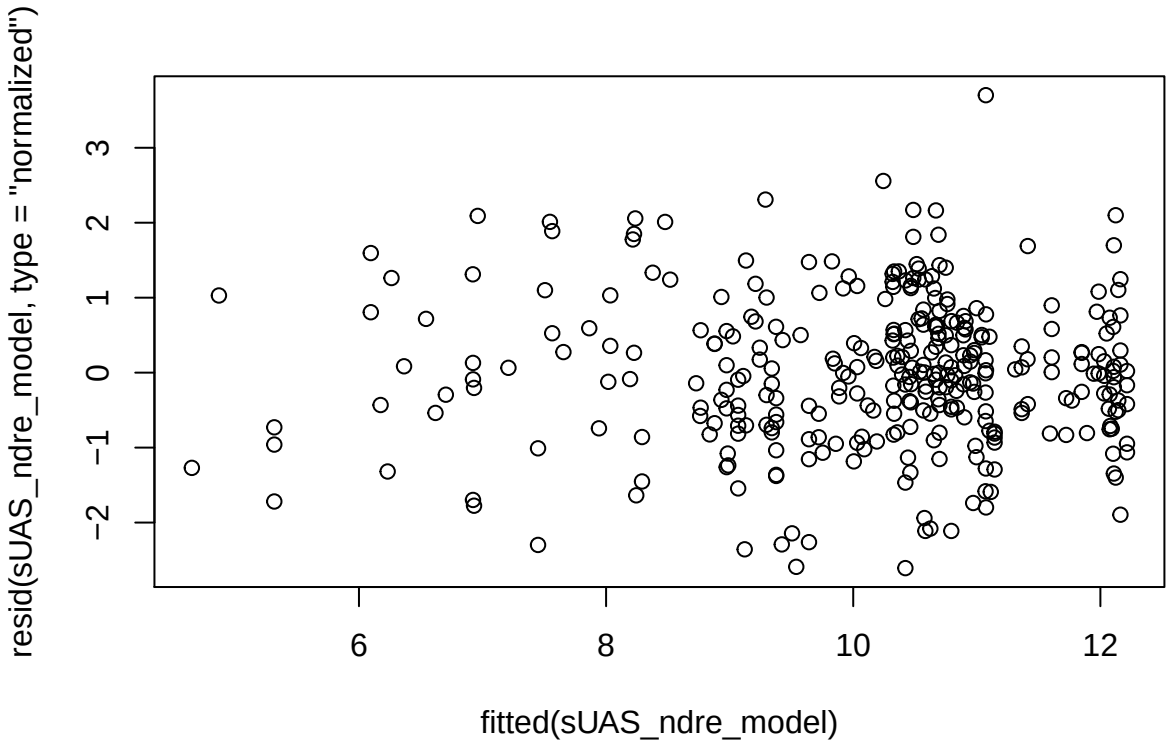
```
##            R2m       R2c
## [1,] 0.5079755 0.8193337
```

### model diagnostics

```
plot (sUAS_ndre_model)
```

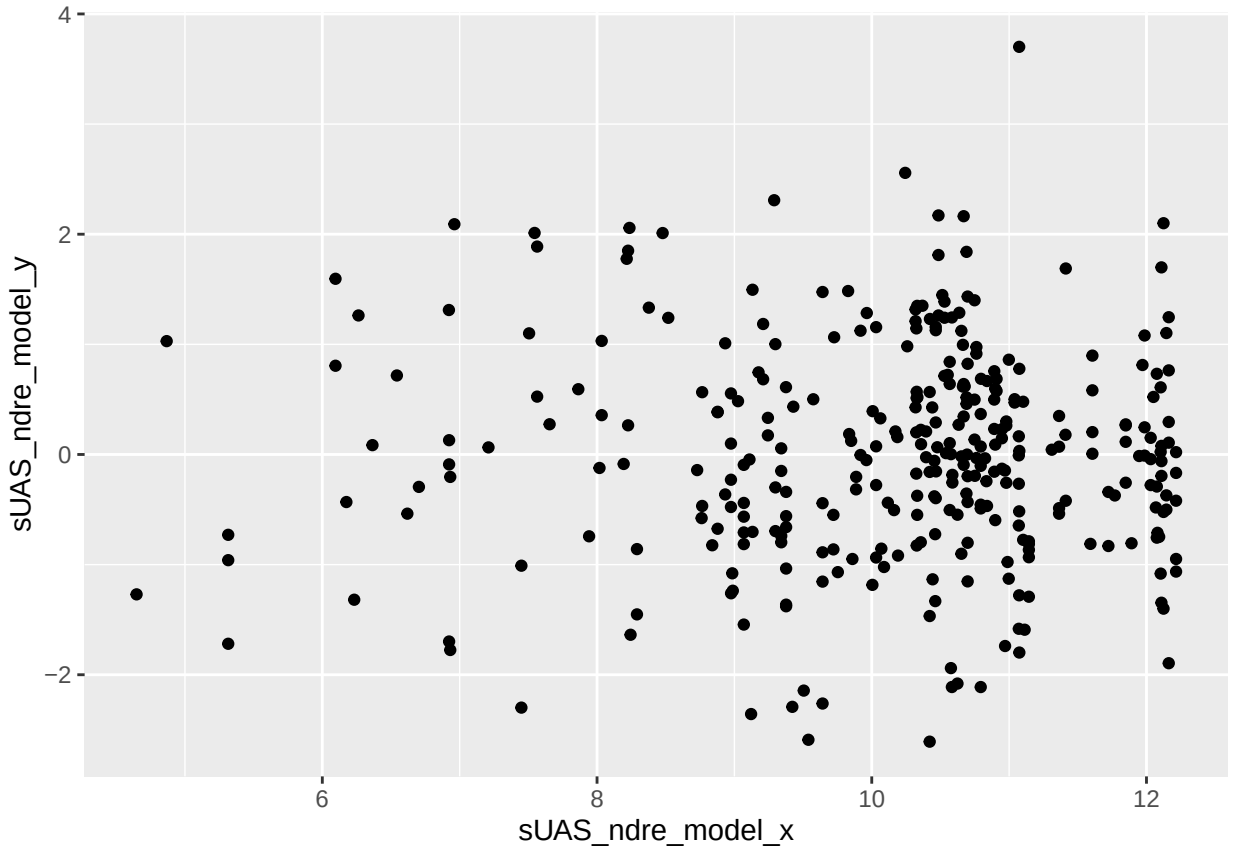



```
plot(resid(sUAS_ndre_model, type = "normalized") ~fitted(sUAS_ndre_model))
```

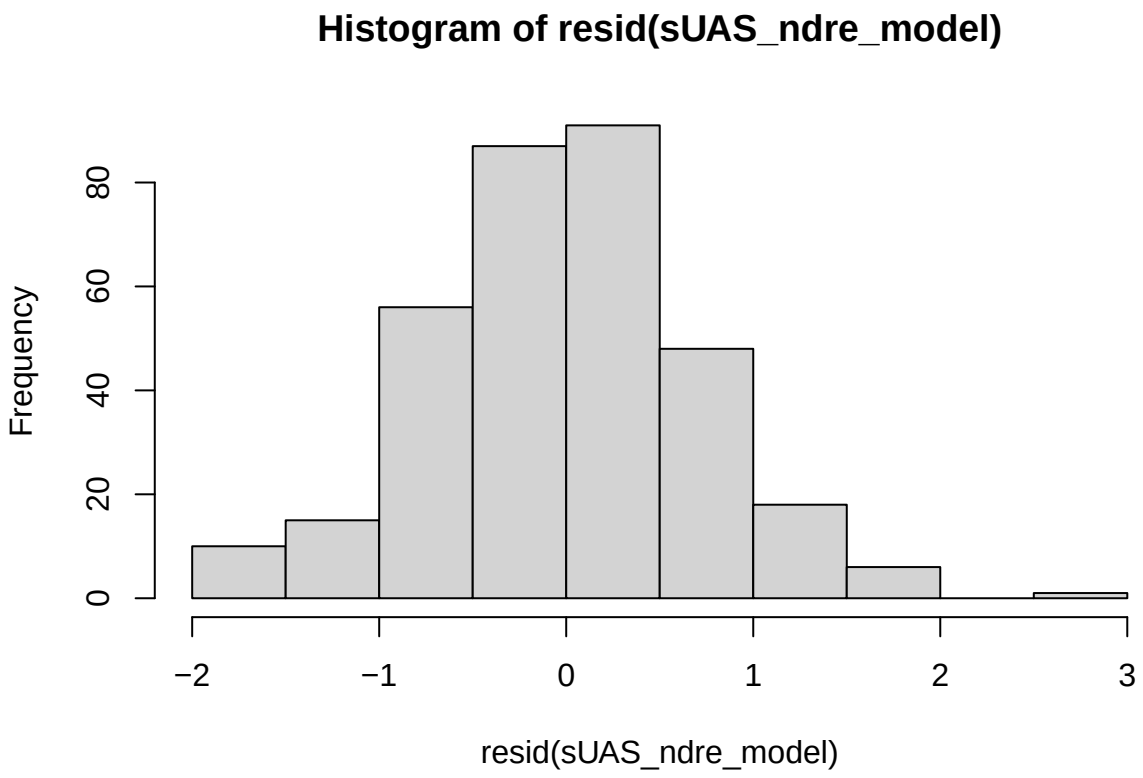



```
sUAS_ndre_model_y <- resid(sUAS_ndre_model, type = "normalized")
sUAS_ndre_model_x <- fitted(sUAS_ndre_model)

sUAS_ndre_modelresid_data <- data.frame(sUAS_ndre_model_x , sUAS_ndre_model_y)
```

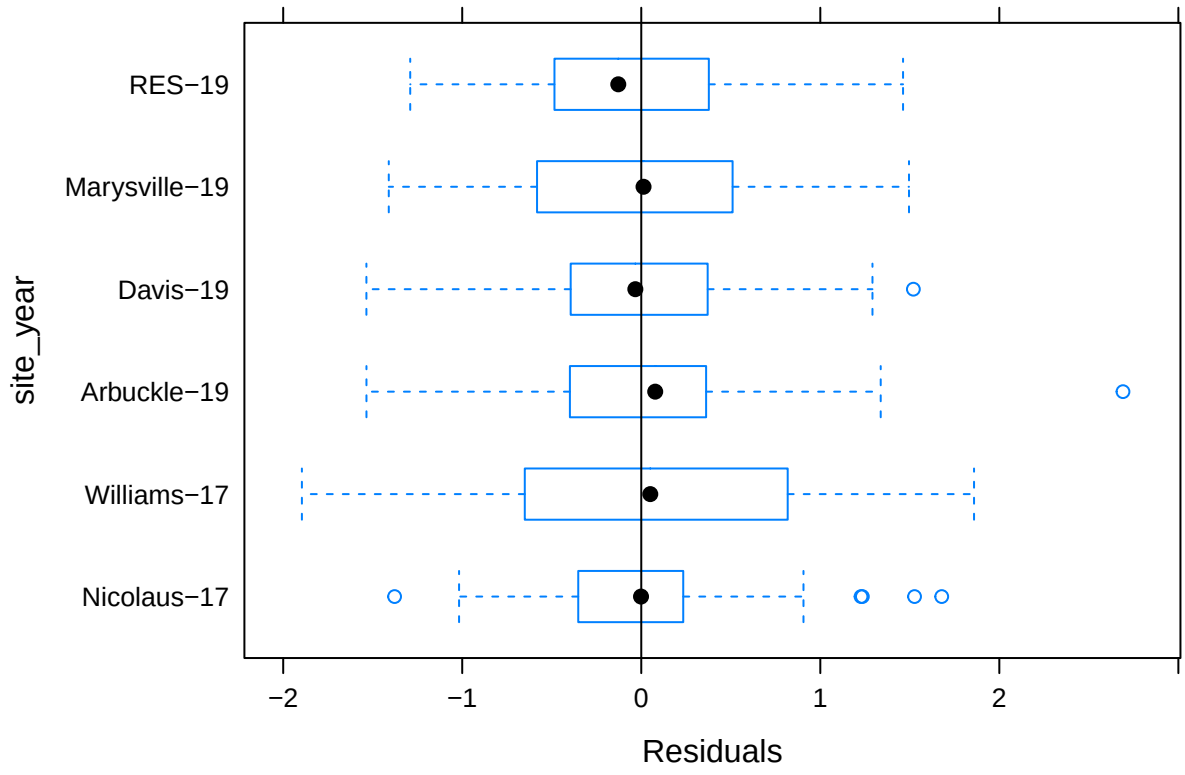
```
ggplot( data = sUAS_ndre_modelresid_data , aes( x = sUAS_ndre_model_x , y = sUAS_ndre_model_y)) +
  geom_point(mapping = aes(sUAS_ndre_model_x , sUAS_ndre_model_y) , data = sUAS_ndre_modelresid_data)
```



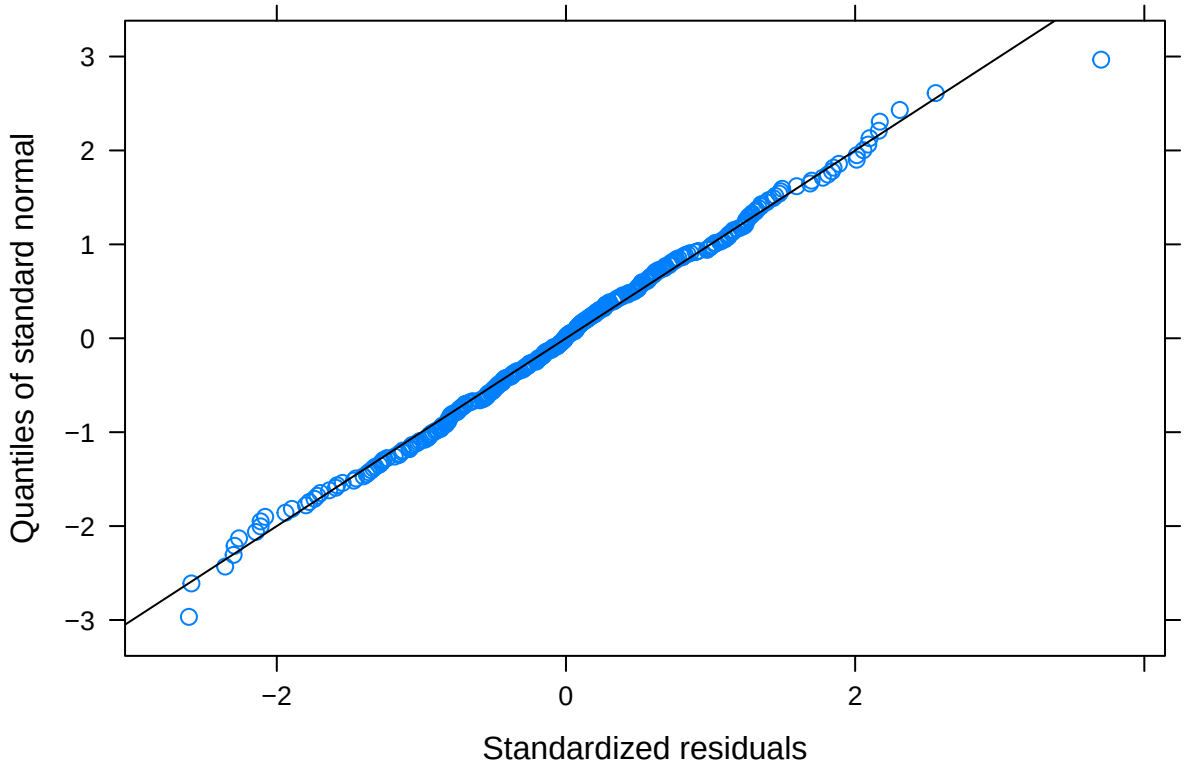

```
hist(resid(sUAS_ndre_model))
```



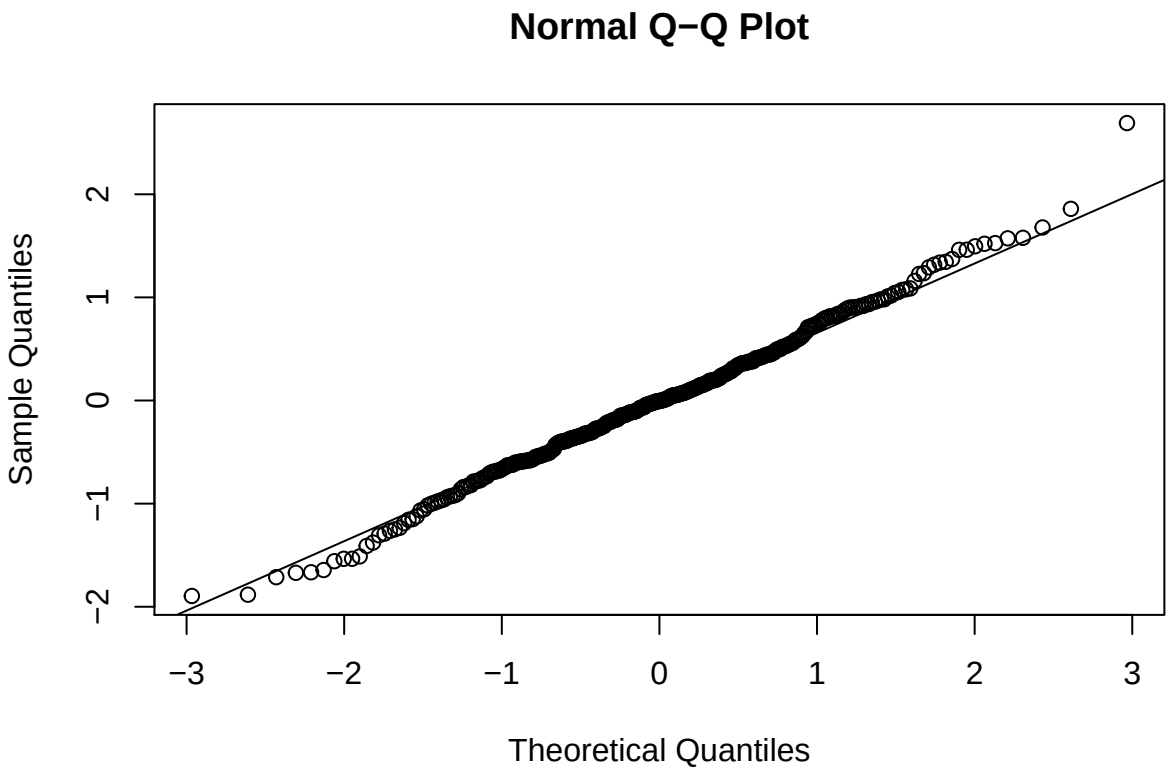

```
plot(sUAS_ndre_model, site_year ~ resid(.), abline = 0)
```



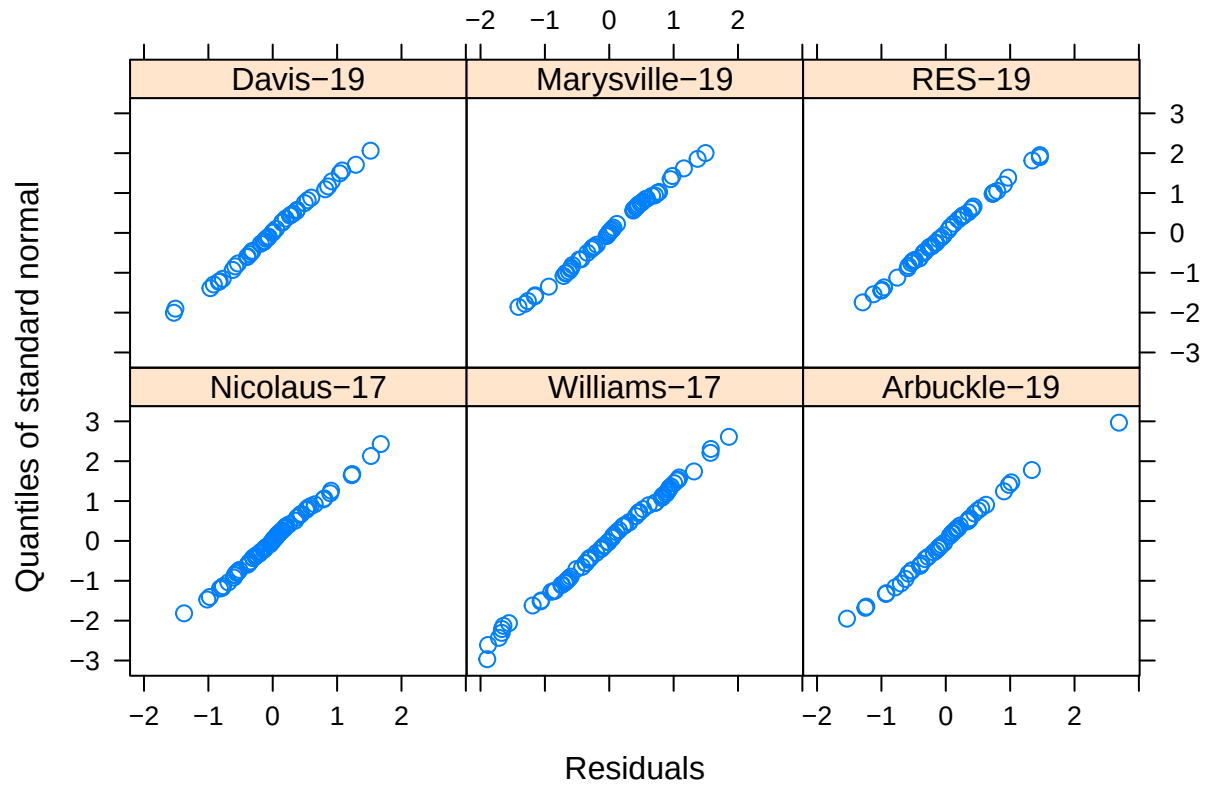

```
qqnorm(sUAS_ndre_model, abline = c(0,1) )
```



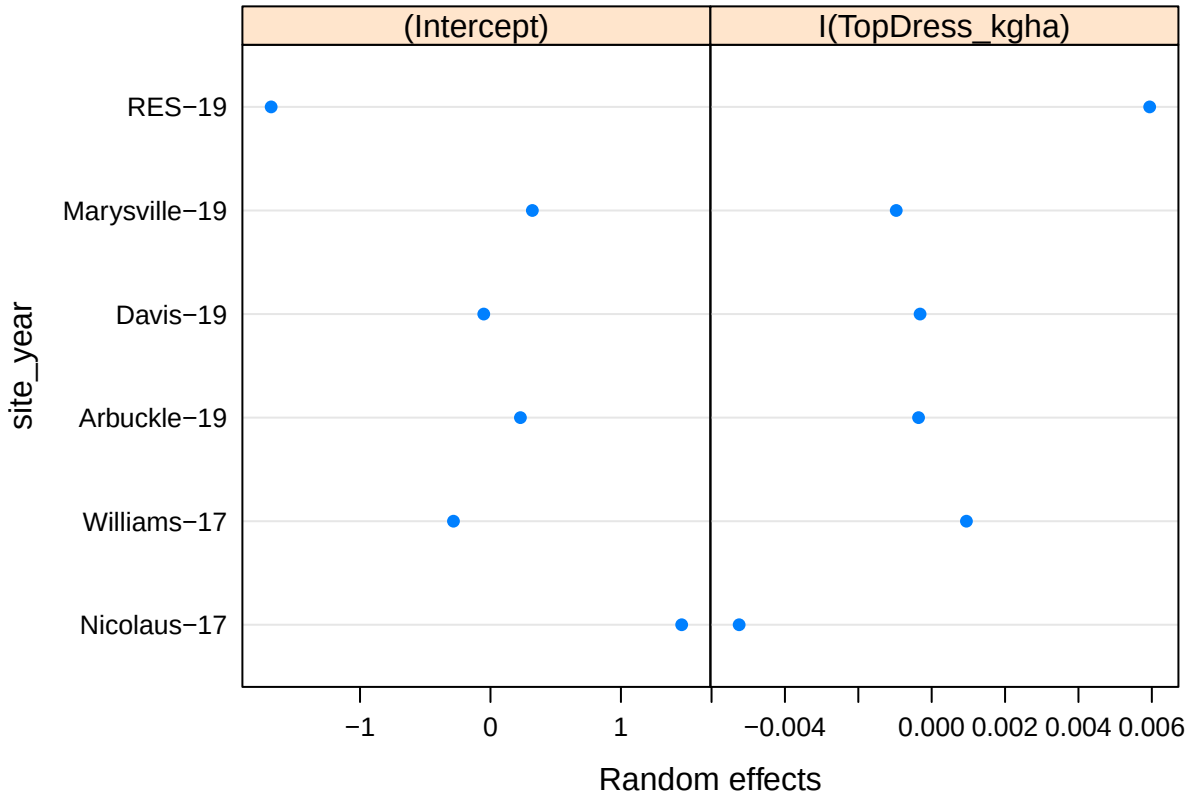

```
qqnorm(resid(sUAS_ndre_model))
qqline(resid(sUAS_ndre_model))
```



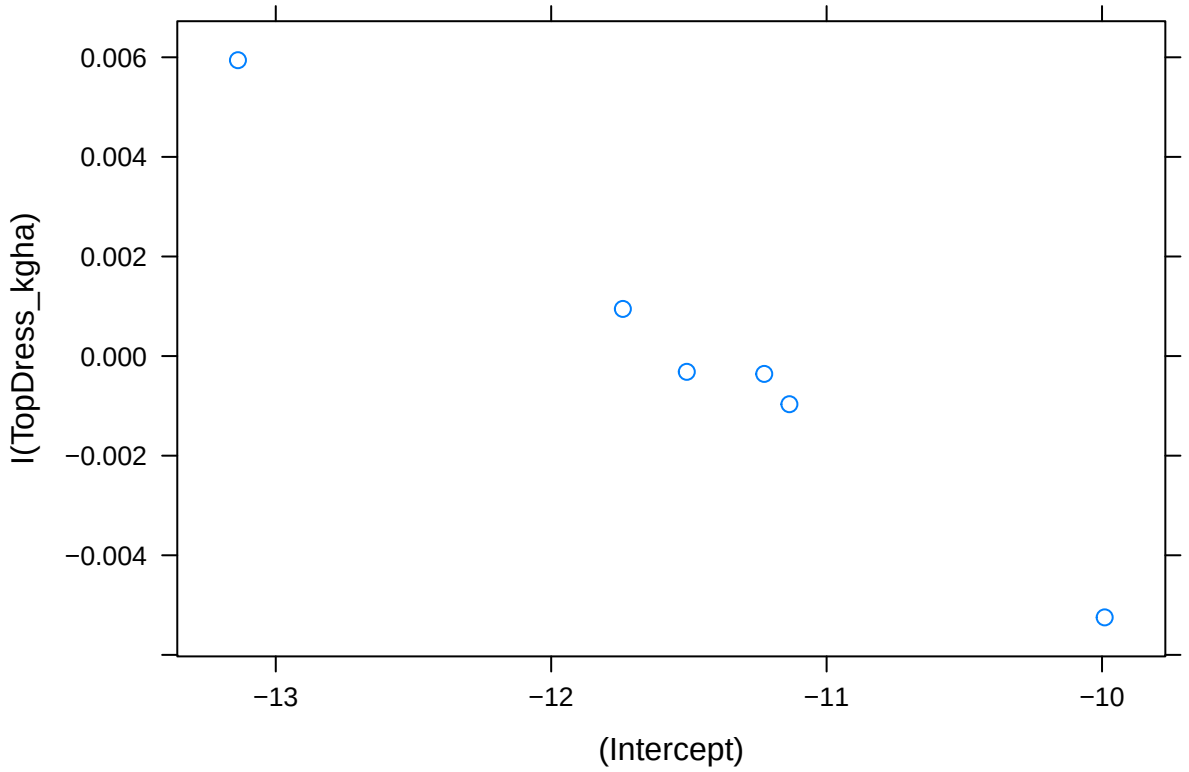
```
qqnorm(sUAS_ndre_model , ~resid(.) | site_year)
```



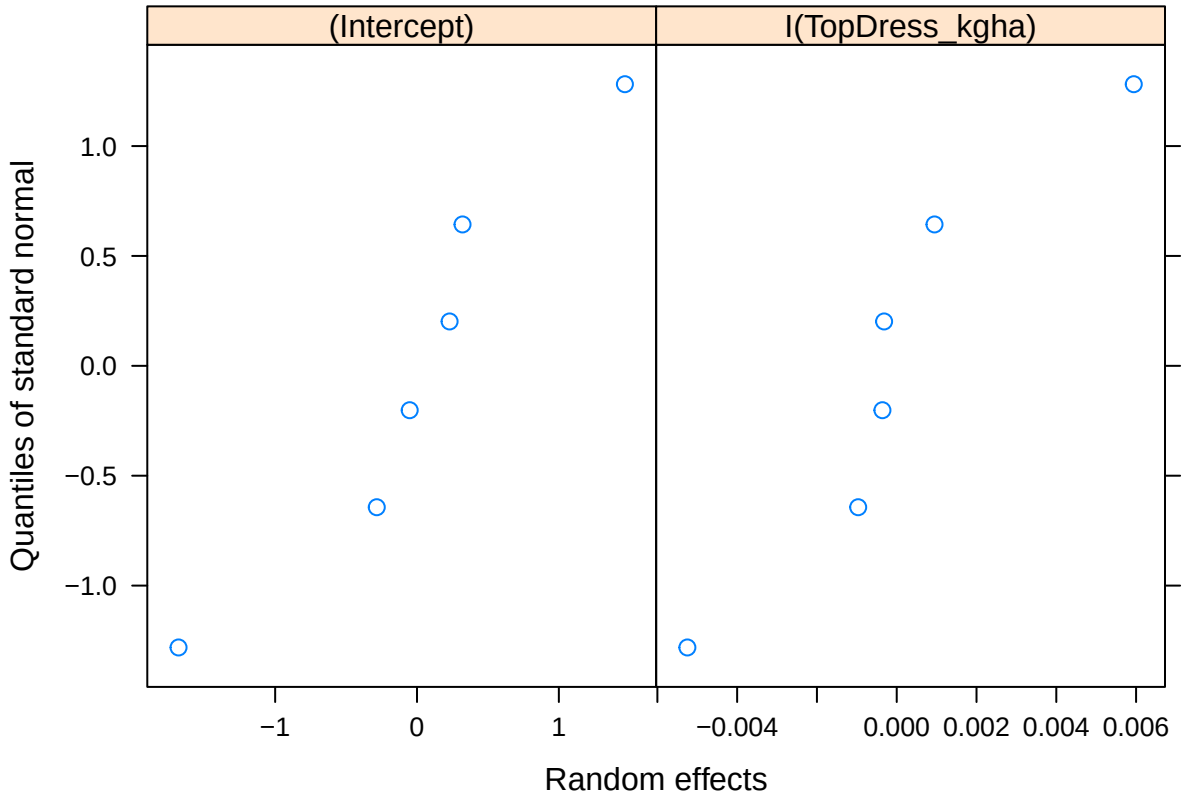

```
plot(ranef(sUAS_ndre_model))
```



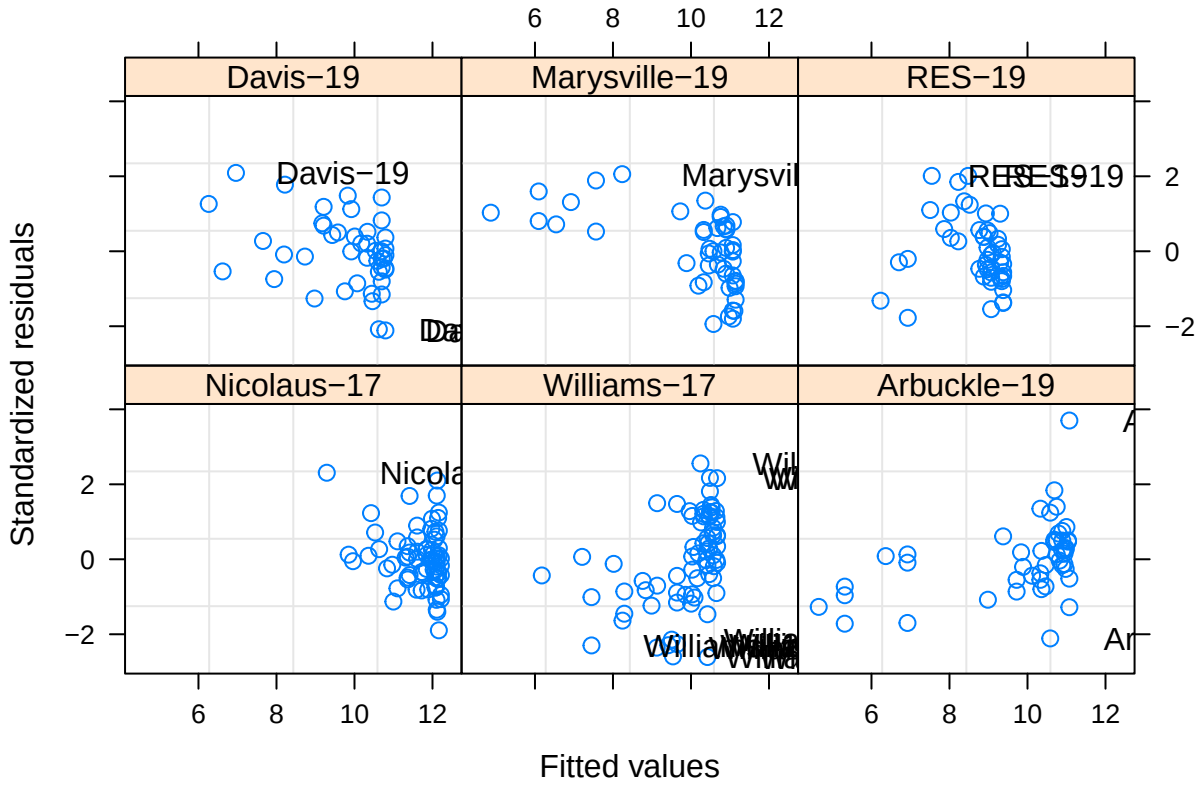
```
pairs(sUAS_ndre_model , id = 0.1)
```




```
qqnorm(sUAS_ndre_model , ~ranef(.))
```



```
plot( sUAS_ndre_model, resid(., type = "p") ~ fitted(.) | site_year,
      id = 0.05, adj = -0.3 )
```




###outlier removal

```
sUAS_ndre_data_OR <- sUAS_ndre_data[-(c(which(abs(residuals(sUAS_ndre_model, type = "normalized"))>qnorm
```

###rerun model sans outliers

```
ctrl <- lmeControl(opt = "optim") #changes control to "optimal" settings

sUAS_ndre_model_OR <- lme(GrainYield_Mgha ~ SI * TopDress_kgha + I(SI*SI)  ,
             control = ctrl ,
             random = ~ I(TopDress_kgha) | site_year  ,
             data = sUAS_ndre_data_OR)

summary(sUAS_ndre_model_OR)
```

```
## Linear mixed-effects model fit by REML
##   Data: sUAS_ndre_data_OR
##        AIC      BIC    logLik
##   768.7183 802.8004 -375.3591
##
## Random effects:
##  Formula: ~I(TopDress_kgha) | site_year
##  Structure: General positive-definite, Log-Cholesky parametrization
##                   StdDev      Corr
## (Intercept)      1.023988033 (Intr)
## I(TopDress_kgha) 0.003917507 -0.938
## Residual         0.711974429
```

```
##
## Fixed effects:  GrainYield_Mgha ~ SI * TopDress_kgha + I(SI * SI)
##                       Value Std.Error  DF   t-value p-value
## (Intercept)      -11.61691  1.684234 321 -6.897447       0
## SI                39.64963  4.040201 321  9.813778       0
## TopDress_kgha      0.10523  0.016450 321  6.397171       0
## I(SI * SI)       -17.29309  2.477001 321 -6.981462       0
## SI:TopDress_kgha  -0.10278  0.018162 321 -5.659094       0
##  Correlation:
##                   (Intr) SI     TpDrs_ I(SI*S
## SI                -0.957
## TopDress_kgha     -0.140  0.012
## I(SI * SI)         0.933 -0.993  0.074
## SI:TopDress_kgha  0.116 -0.013 -0.987 -0.075
##
## Standardized Within-Group Residuals:
##          Min           Q1          Med           Q3          Max
## -2.661180960 -0.649624675 -0.002562116  0.613738418  2.604603880
##
## Number of Observations: 331
## Number of Groups: 6
```

```
summary(sUAS_ndre_model_OR)$tTable
```

```
##                       Value  Std.Error  DF   t-value      p-value
## (Intercept)      -11.6169128 1.68423378 321 -6.897447 2.824738e-11
## SI                39.6496313 4.04020072 321  9.813778 4.707103e-20
## TopDress_kgha      0.1052309 0.01644959 321  6.397171 5.593719e-10
## I(SI * SI)       -17.2930895 2.47700116 321 -6.981462 1.684349e-11
## SI:TopDress_kgha  -0.1027811 0.01816211 321 -5.659094 3.374042e-08
```

```
Anova(sUAS_ndre_model_OR , type = 3)
```

```
## Analysis of Deviance Table (Type III tests)
##
## Response: GrainYield_Mgha
##                   Chisq Df Pr(>Chisq)
## (Intercept)      47.575  1  5.295e-12 ***
## SI               96.310  1  < 2.2e-16 ***
## TopDress_kgha    40.924  1  1.583e-10 ***
## I(SI * SI)       48.741  1  2.921e-12 ***
## SI:TopDress_kgha 32.025  1  1.522e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
ndre_r_sq <- r.squaredGLMM(sUAS_ndre_model_OR)

ndre_r_sq_fixed <- round(ndre_r_sq[1] , digits = 2)
ndre_r_sq_fixed
```

```
## [1] 0.51
```

```
ndre_r_sq_total <- round(ndre_r_sq[2] , digits = 2)
ndre_r_sq_total
```

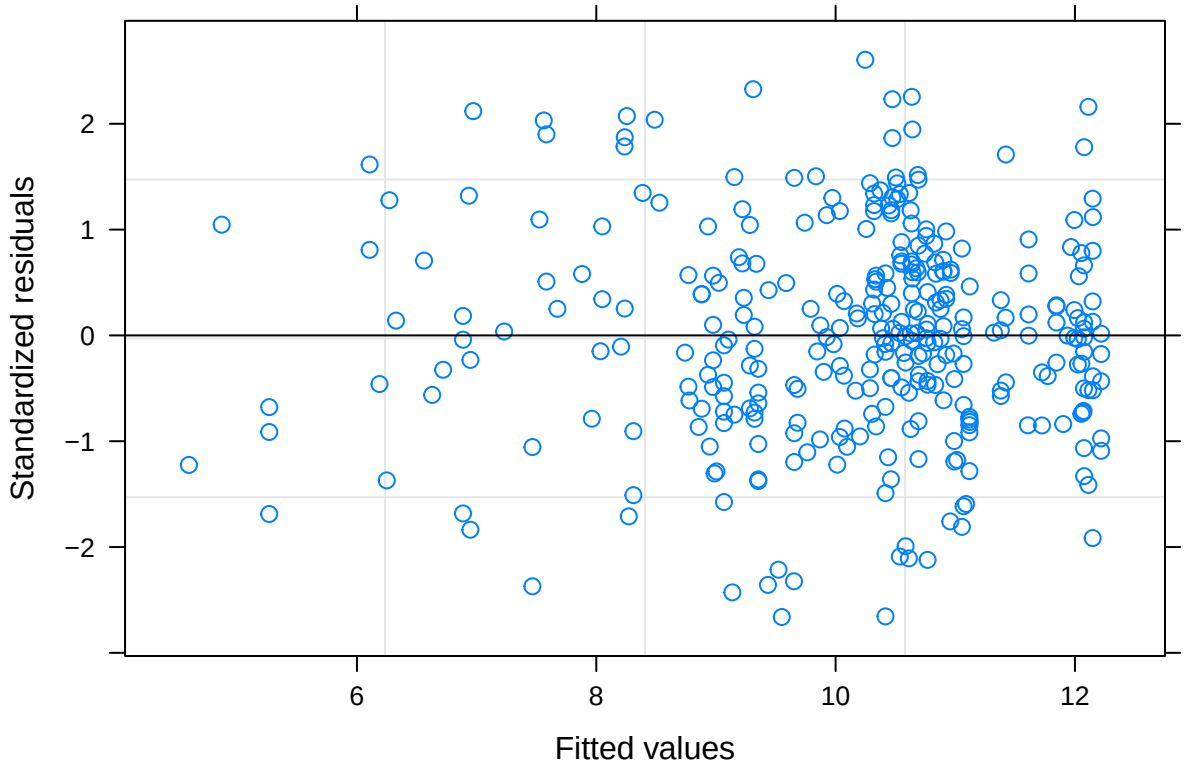
```
## [1] 0.82
```

```
ndre_r_sq_random <- ndre_r_sq_total - ndre_r_sq_fixed
ndre_r_sq_random
```

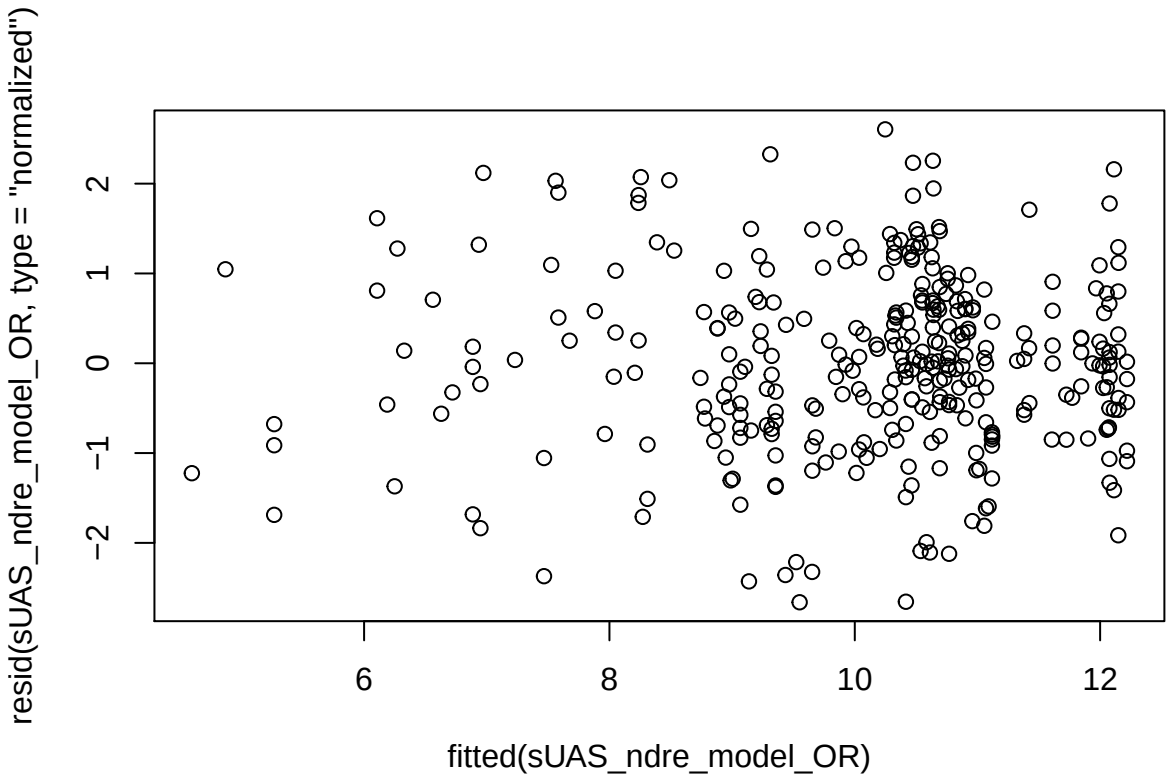
```
## [1] 0.31
```

### model diagnostics

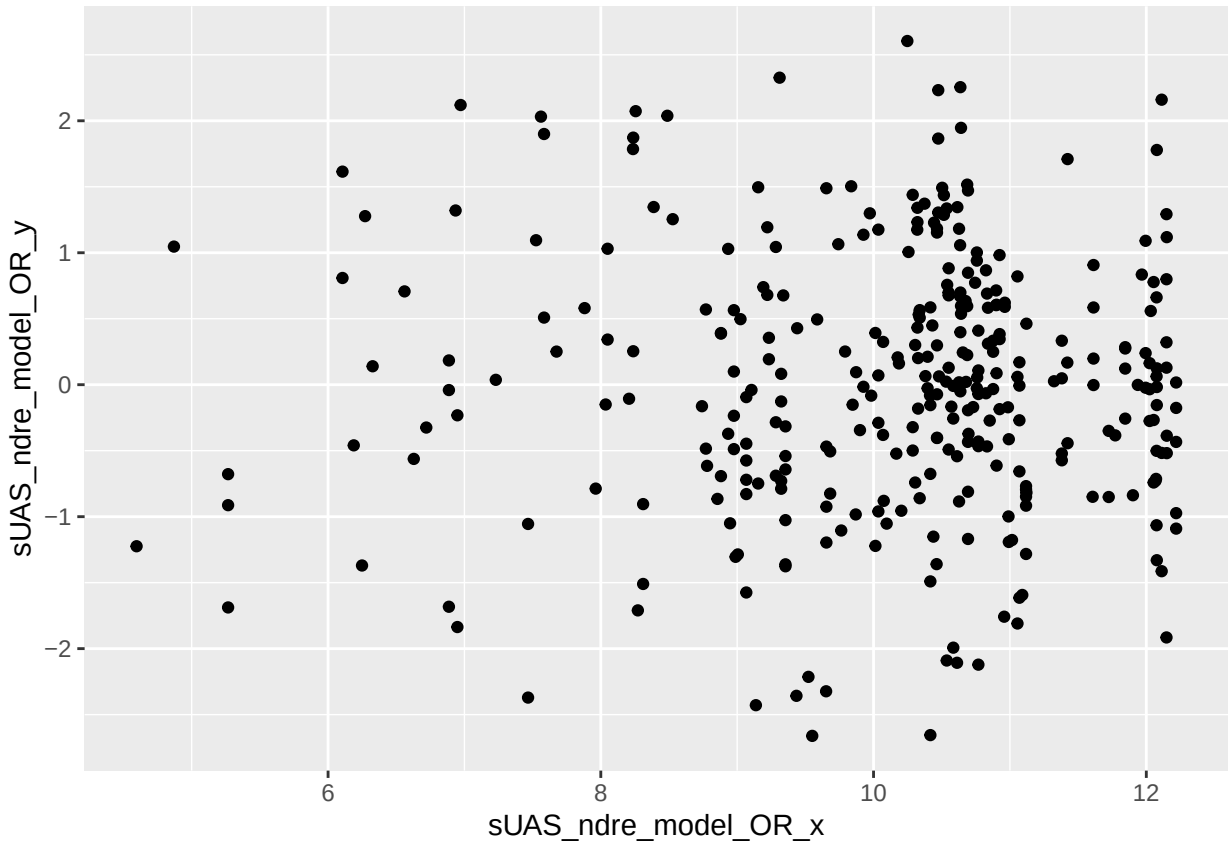
```
plot (sUAS_ndre_model_OR)
```



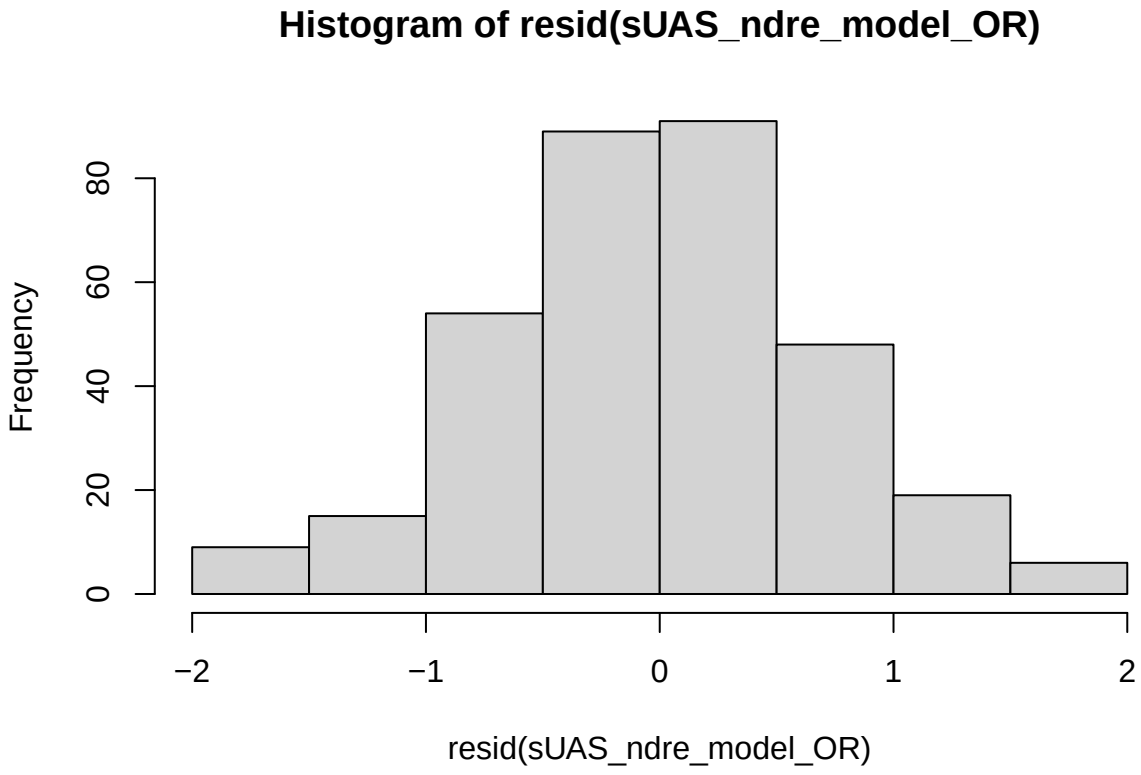

```
plot(resid(sUAS_ndre_model_OR, type = "normalized") ~fitted(sUAS_ndre_model_OR))
```



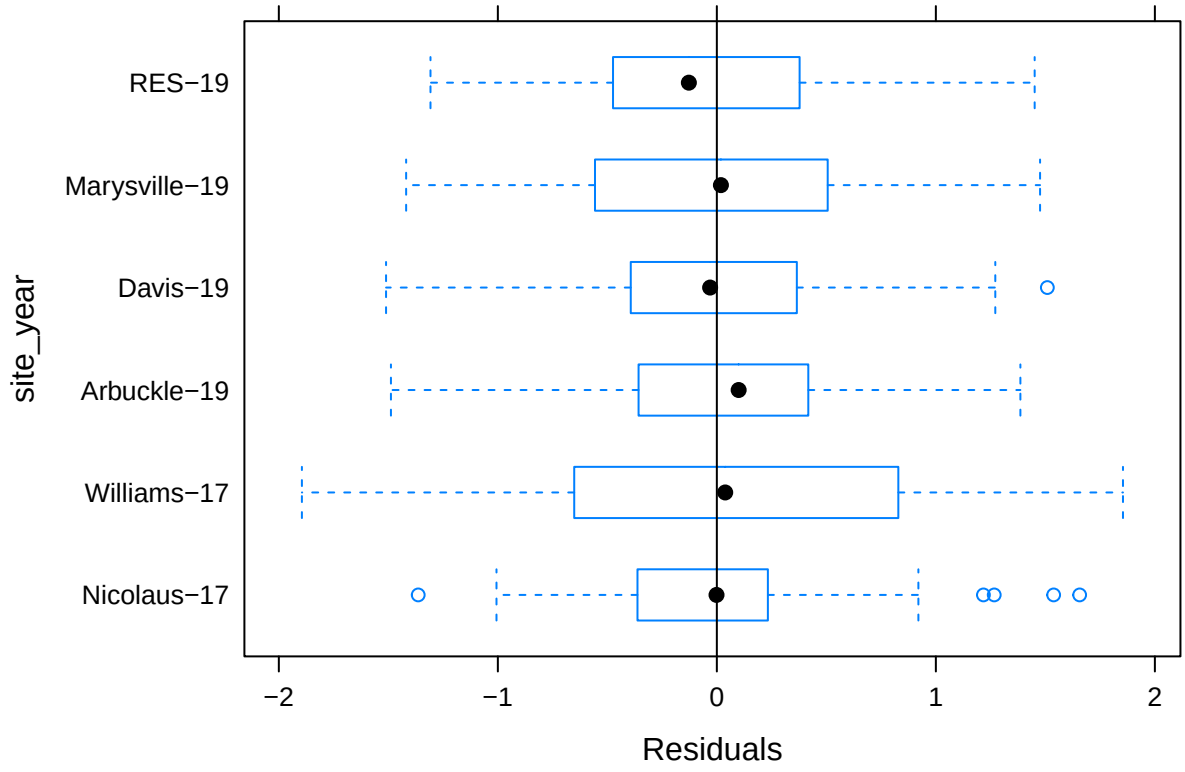

```
sUAS_ndre_model_OR_y <- resid(sUAS_ndre_model_OR, type = "normalized")
sUAS_ndre_model_OR_x <- fitted(sUAS_ndre_model_OR)

sUAS_ndre_model_ORresid_data <- data.frame(sUAS_ndre_model_OR_x , sUAS_ndre_model_OR_y)

ggplot( data = sUAS_ndre_model_ORresid_data , aes( x = sUAS_ndre_model_OR_x , y = sUAS_ndre_model_OR_y)
  geom_point(mapping = aes(sUAS_ndre_model_OR_x , sUAS_ndre_model_OR_y) , data = sUAS_ndre_model_ORresi
```



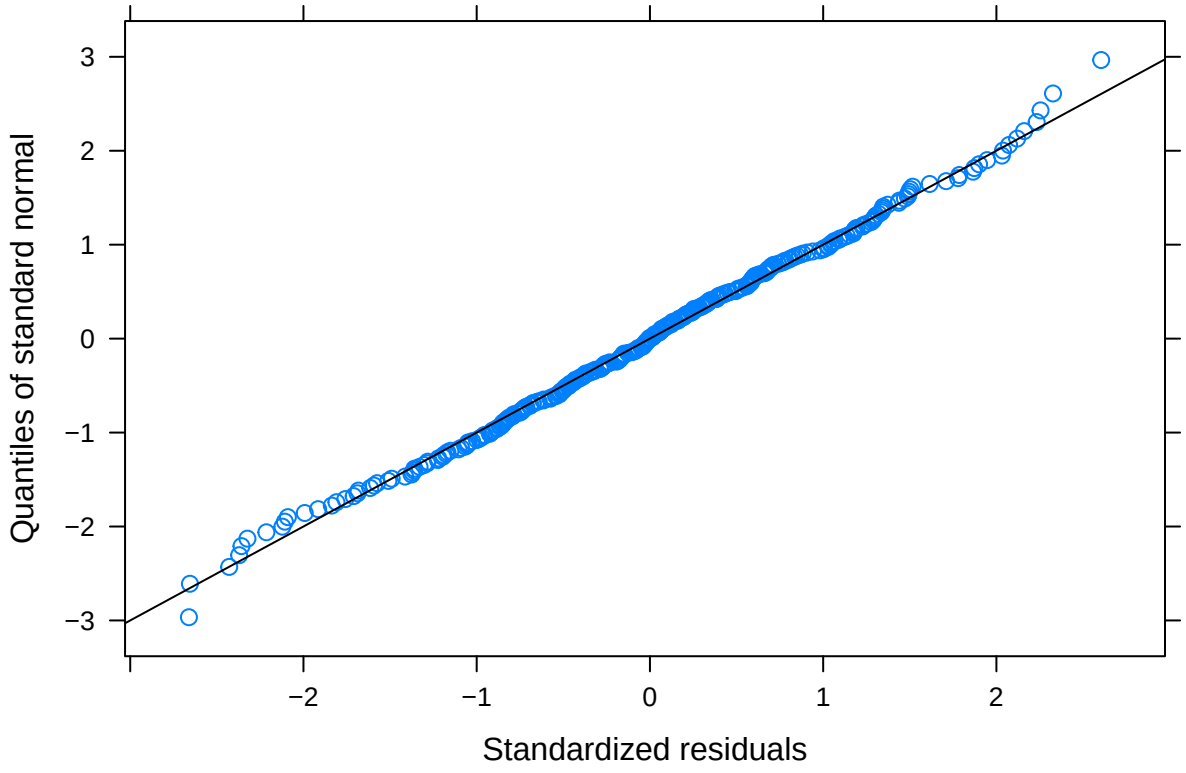

```
hist(resid(sUAS_ndre_model_OR))
```



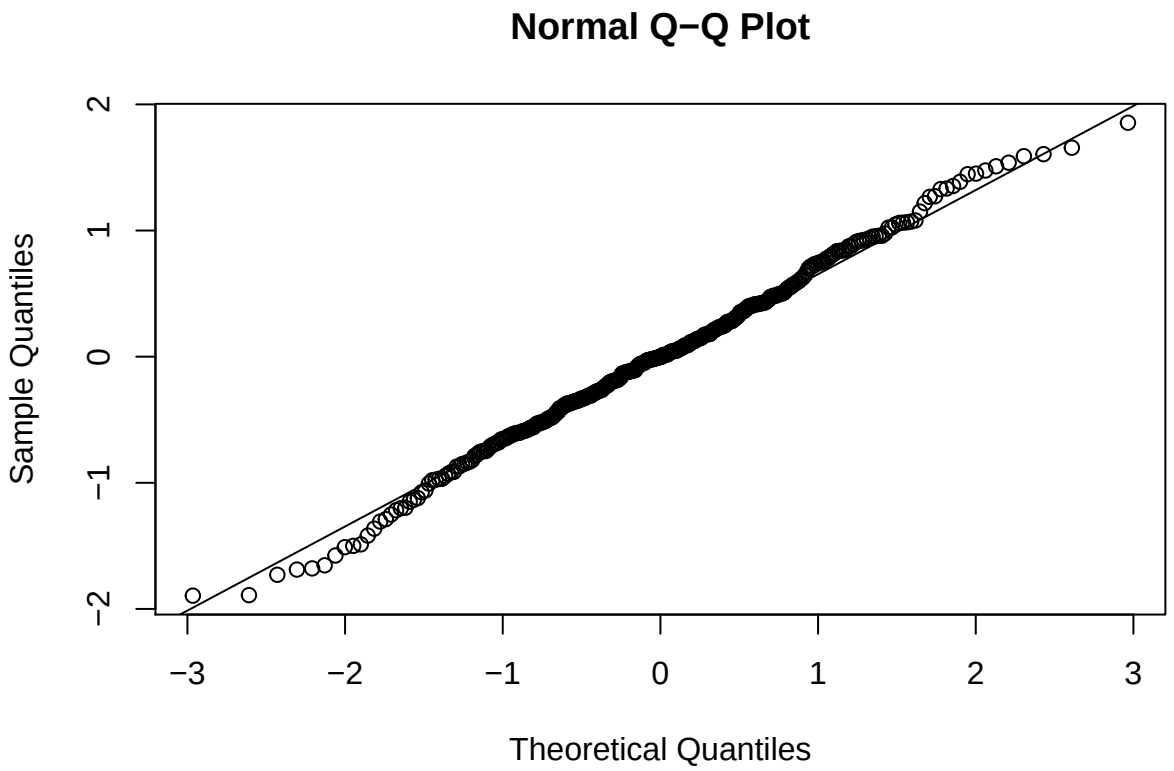
```
plot(sUAS_ndre_model_OR, site_year ~ resid(.), abline = 0)
```



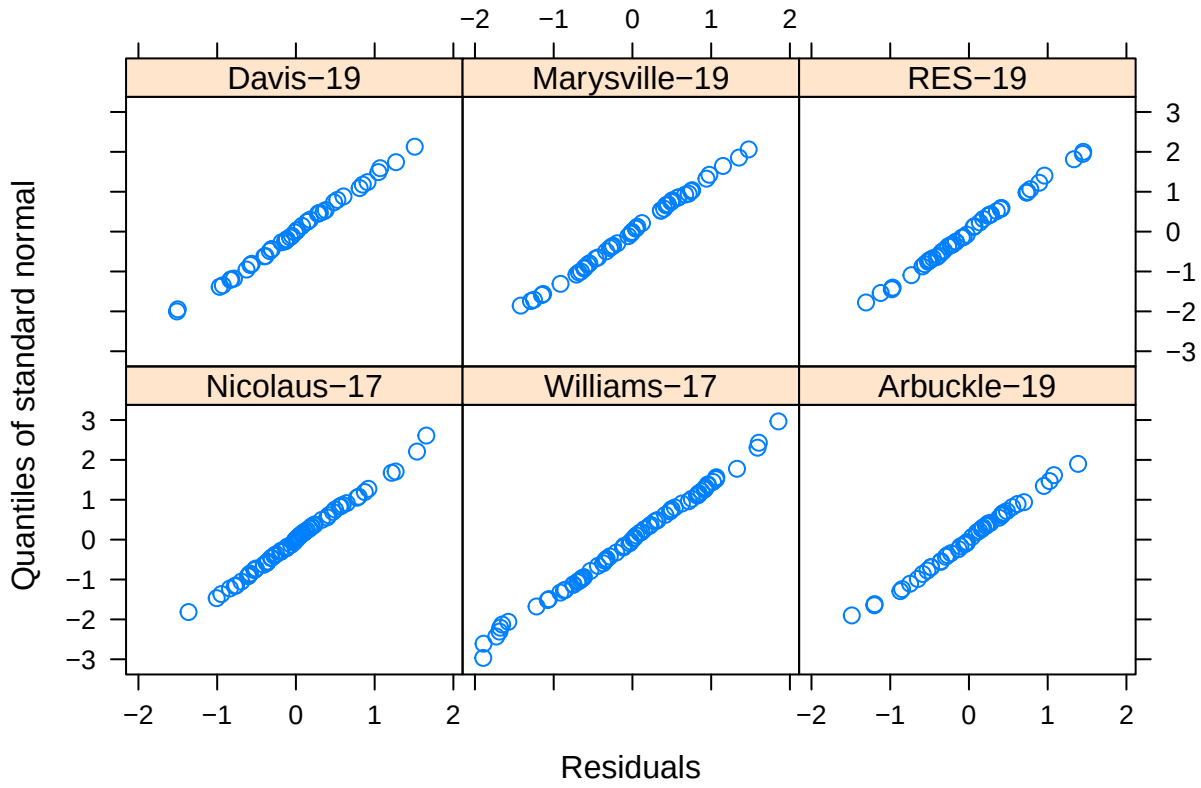

```
qqnorm(sUAS_ndre_model_OR, abline = c(0,1) )
```



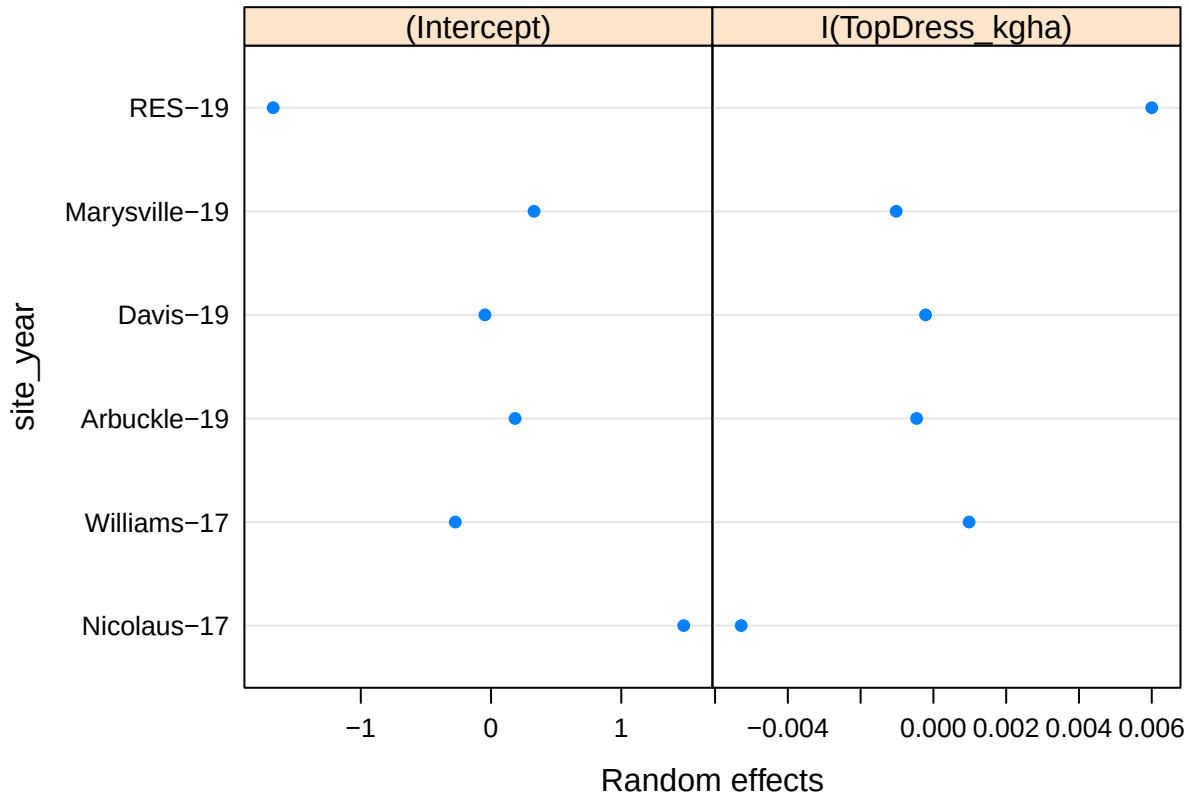
```
qqnorm(resid(sUAS_ndre_model_OR))
qqline(resid(sUAS_ndre_model_OR))
```



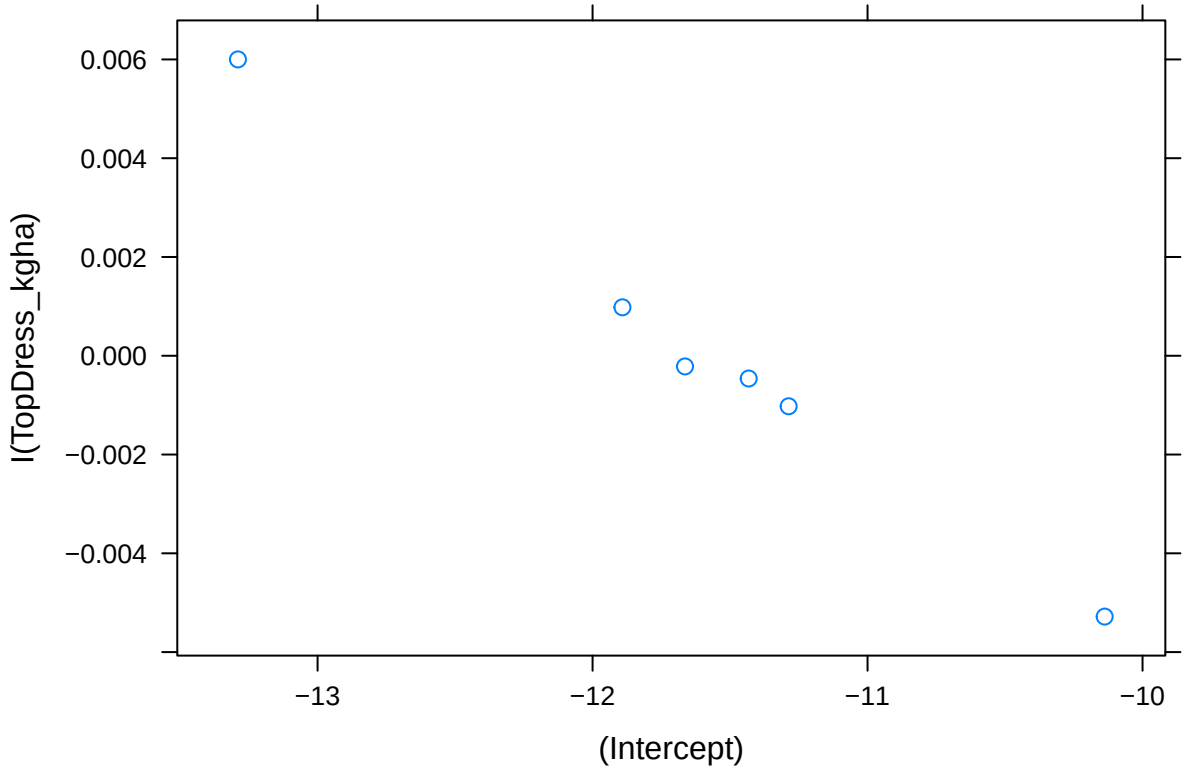

```
qqnorm(sUAS_ndre_model_OR , ~resid(.) | site_year)
```



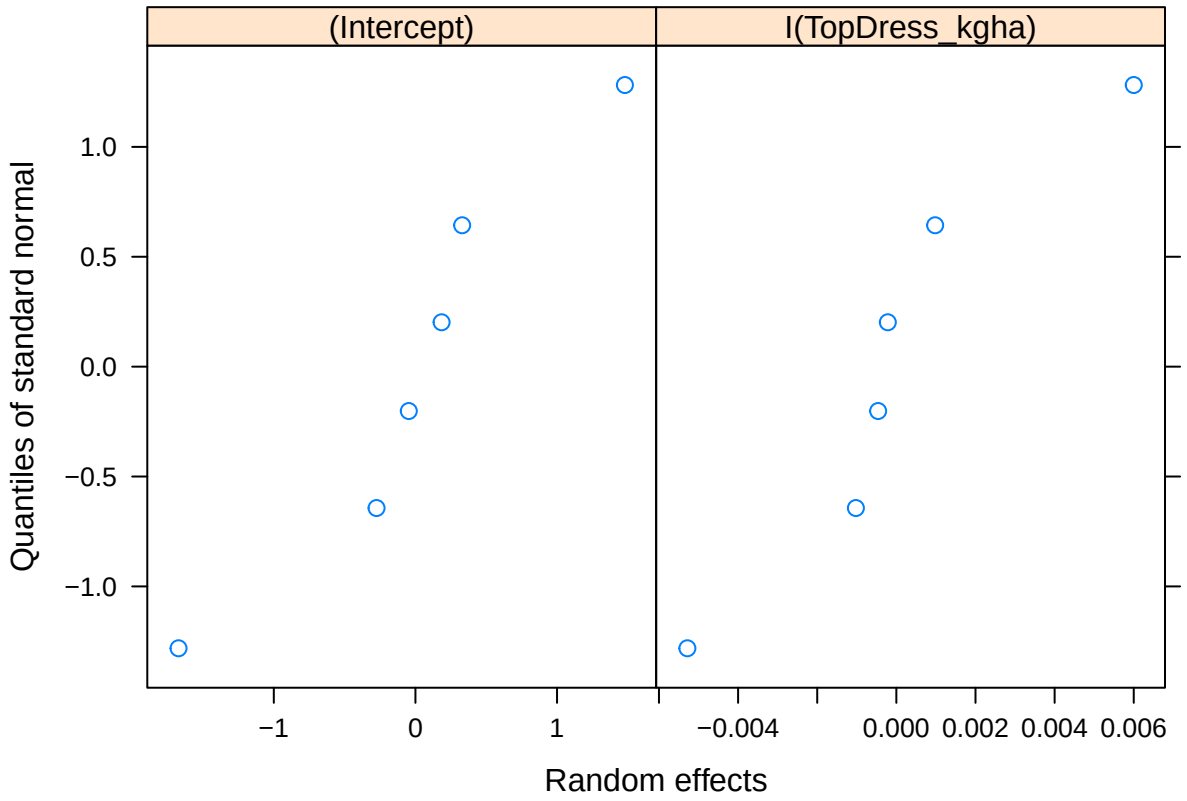
```
plot(ranef(sUAS_ndre_model_OR))
```



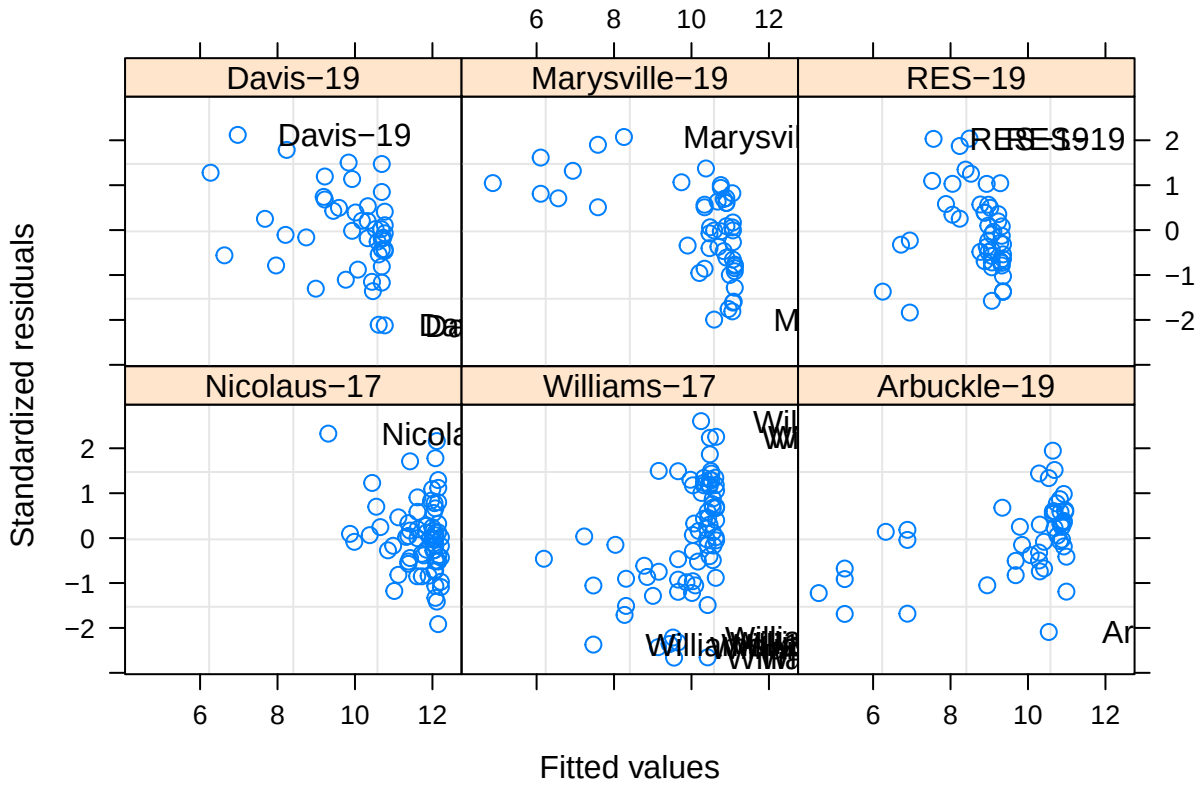

```
pairs(sUAS_ndre_model_OR , id = 0.1)
```



```
qqnorm(sUAS_ndre_model_OR , ~ranef(.))
```




```
plot( sUAS_ndre_model_OR, resid(., type = "p") ~ fitted(.) | site_year,
      id = 0.05, adj = -0.3 )
```




###emmeans

```
mylist <- list(SI=seq(0.528 , 1 , by = .0001),TopDress_kgha =c( 34 , 0))

sUAS_ndre_emmeans <- emmeans(sUAS_ndre_model_OR , ~TopDress_kgha * SI , at = mylist )

sUAS_ndre_emmeans_contrast <- as.data.frame(summary(contrast(sUAS_ndre_emmeans , "pairwise" , side = ">

sUAS_ndre_emmeans_contrast <- sUAS_ndre_emmeans_contrast %>%
  mutate(sUAS_ndre_Response_Index = 1 / SI )

sUAS_ndre_emmeans_contrast <- sUAS_ndre_emmeans_contrast %>%
  mutate(prob_postive_resp = (1 - p.value) * 100)

sUAS_ndre_emmeans_contrast <- sUAS_ndre_emmeans_contrast %>%
  mutate(t_score = abs((estimate - 0.26)) / SE)

sUAS_ndre_emmeans_contrast <- sUAS_ndre_emmeans_contrast %>%
  mutate(prob_greater_than_26 = if_else(estimate < 0.26 , pt(q = t_score , df = df , lower.tail = F) , p

sUAS_ndre_emmeans_contrast$sUAS_ndre_Response_Index_r <- (round(sUAS_ndre_emmeans_contrast$sUAS_ndre_Res

sUAS_ndre_run1 <- 125 - 100
sUAS_ndre_rise1 <- 0.7890828 - 0.08455915
sUAS_ndre_slope1 <- round((sUAS_ndre_rise1 / sUAS_ndre_run1) * 5 , digits = 2)
sUAS_ndre_slope1
```

```
## [1] 0.14
```

```
sUAS_ndre_run2 <- 189 - 125
sUAS_ndre_rise2 <- 1.747235 - 0.7890828
sUAS_ndre_slope2 <- round((sUAS_ndre_rise2 / sUAS_ndre_run2) * 5 , digits = 3)
sUAS_ndre_slope2
```

```
## [1] 0.075
```

```
sUAS_ndre_mean_se <- sUAS_ndre_emmeans_contrast %>%
  select(SE) %>%
  summarise(mean_se = mean(SE))

sUAS_ndre_mean_se$mean_se <- round(sUAS_ndre_mean_se$mean_se , digits = 2)

sUAS_ndre_mean_se$mean_se
```

```
## [1] 0.14
```

###confidence interval

```
#at every 0.10 interval, what is the 90% confidence interval of the yield response, and what RI value d

str(sUAS_ndre_emmeans_contrast , give.attr = F)
```

```
## 'data.frame':    4721 obs. of  12 variables:
##  $ contrast                 : Factor w/ 1 level "34 - 0": 1 1 1 1 1 1 1 1 1 1 ...
##  $ SI                       : num  0.528 0.528 0.528 0.528 0.528 ...
##  $ estimate                 : num  1.73 1.73 1.73 1.73 1.73 ...
##  $ SE                       : num  0.243 0.243 0.243 0.243 0.243 ...
##  $ df                       : num  321 321 321 321 321 321 321 321 321 321 ...
##  $ t.ratio                  : num  7.12 7.12 7.12 7.12 7.13 ...
##  $ p.value                  : num  3.47e-12 3.46e-12 3.46e-12 3.45e-12 3.45e-12 ...
##  $ sUAS_ndre_Response_Index : num  1.89 1.89 1.89 1.89 1.89 ...
##  $ prob_postive_resp        : num  100 100 100 100 100 ...
##  $ t_score                  : num  6.06 6.06 6.06 6.06 6.06 ...
##  $ prob_greater_than_26     : num  1 1 1 1 1 ...
##  $ sUAS_ndre_Response_Index_r: num  1.89 1.89 1.89 1.89 1.89 ...
```

```
sUAS_ndre_emmeans_contrast <- sUAS_ndre_emmeans_contrast %>%
  mutate(response_mgha_r = round(estimate , digits = 2),
         se_mgha_r = round(SE , digits = 3))

#estimate = 0.10 , SE = 0.112

print(round(0.10 - z_score*0.112 , digits = 2))
```

```
## [1] -0.08
```

```
print(round(0.10 + z_score*0.112 , digits = 2))
```

```
## [1] 0.28
```

```
#estimate = 0.20 , SE = 0.103

print(round(0.20 - z_score*0.103 , digits = 2))
```

```
## [1] 0.03
```

```
print(round(0.20 + z_score*0.103 , digits = 2))
```

```
## [1] 0.37
```

```
#estimate = 0.30 , SE = 0.097

print(round(0.30 - z_score*0.097 , digits = 2))
```

```
## [1] 0.14
```

```
print(round(0.30 + z_score*0.097 , digits = 2))
```

```
## [1] 0.46
```

```
#estimate = 0.40 , SE = 0.093

print(round(0.40 - z_score*0.093 , digits = 2))
```

```
## [1] 0.25
```

```
print(round(0.40 + z_score*0.093 , digits = 2))
```

```
## [1] 0.55
```

```
#estimate = 0.50 , SE = 0.093

print(round(0.50 - z_score*0.093 , digits = 2))
```

```
## [1] 0.35
```

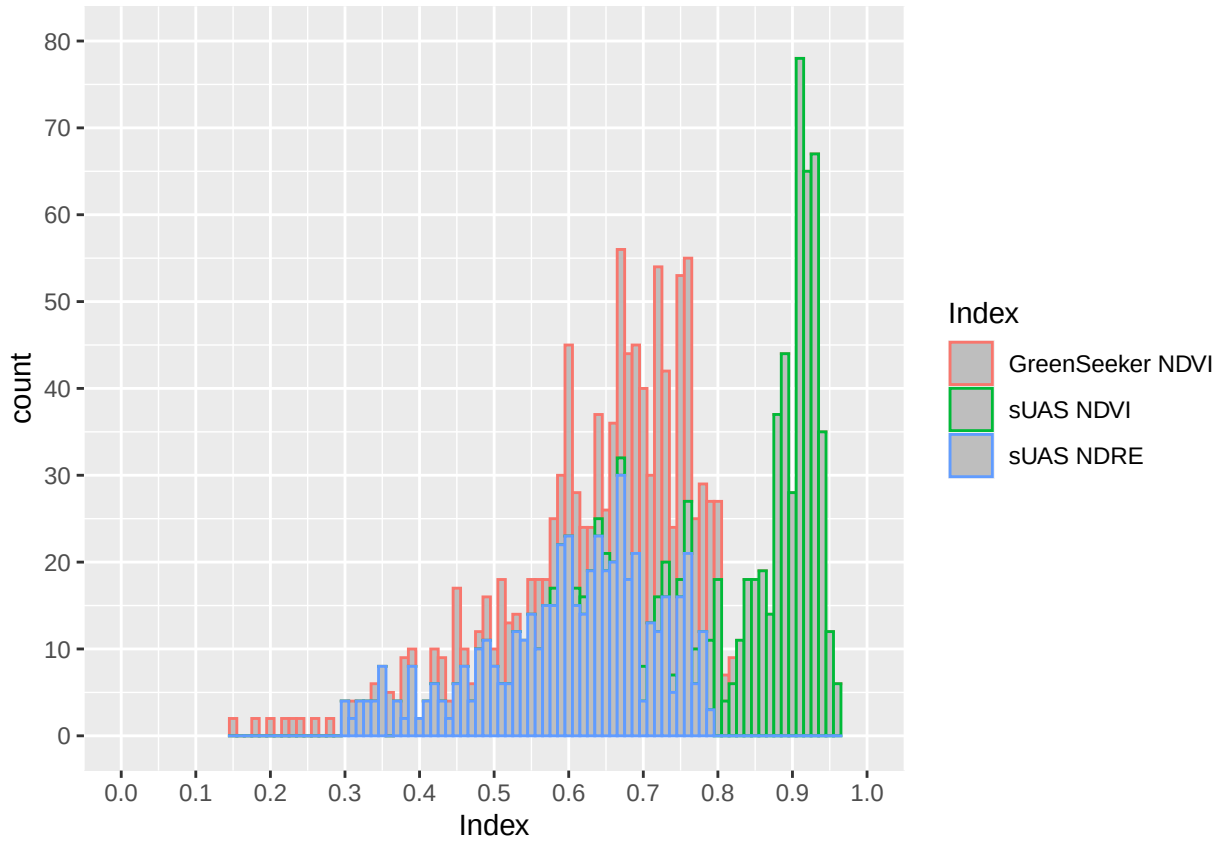
```
print(round(0.50 + z_score*0.093 , digits = 2))
```

```
## [1] 0.65
```

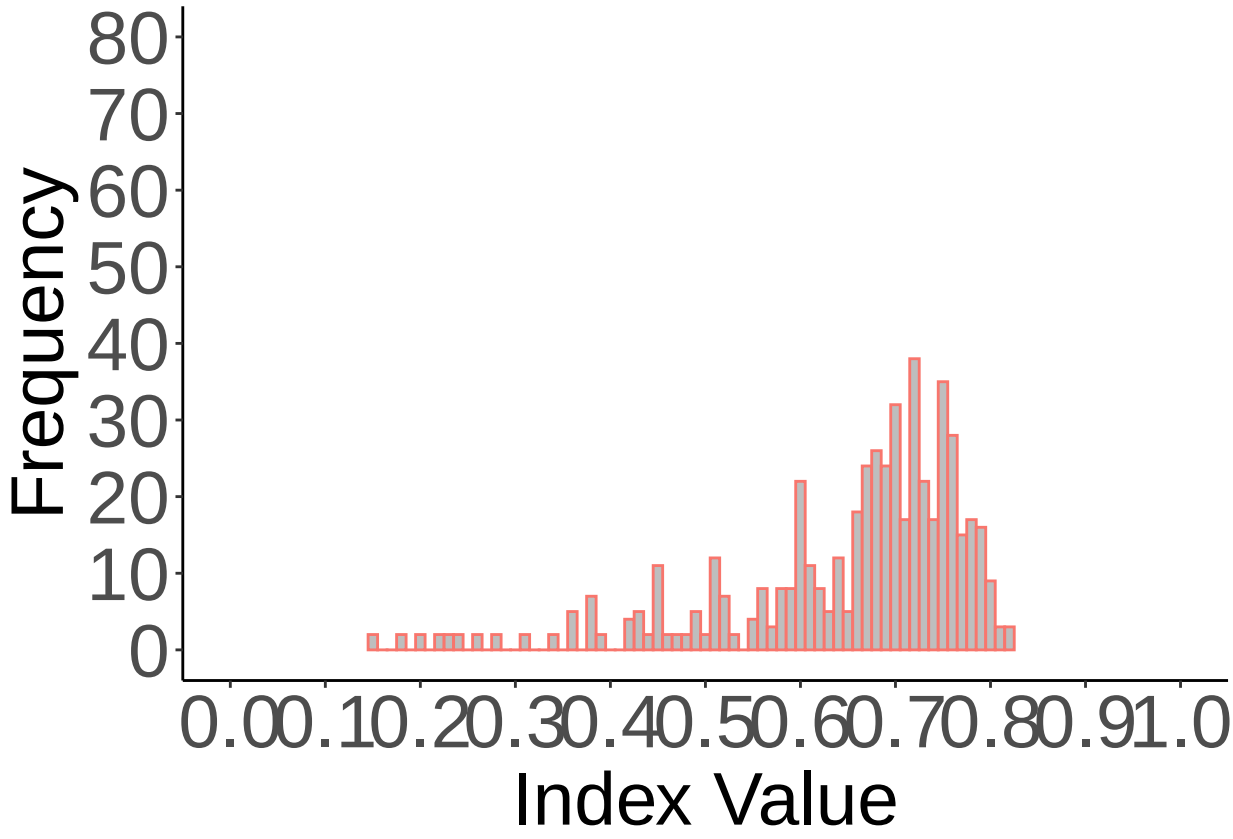
#FIGURES

##FIGURE 2

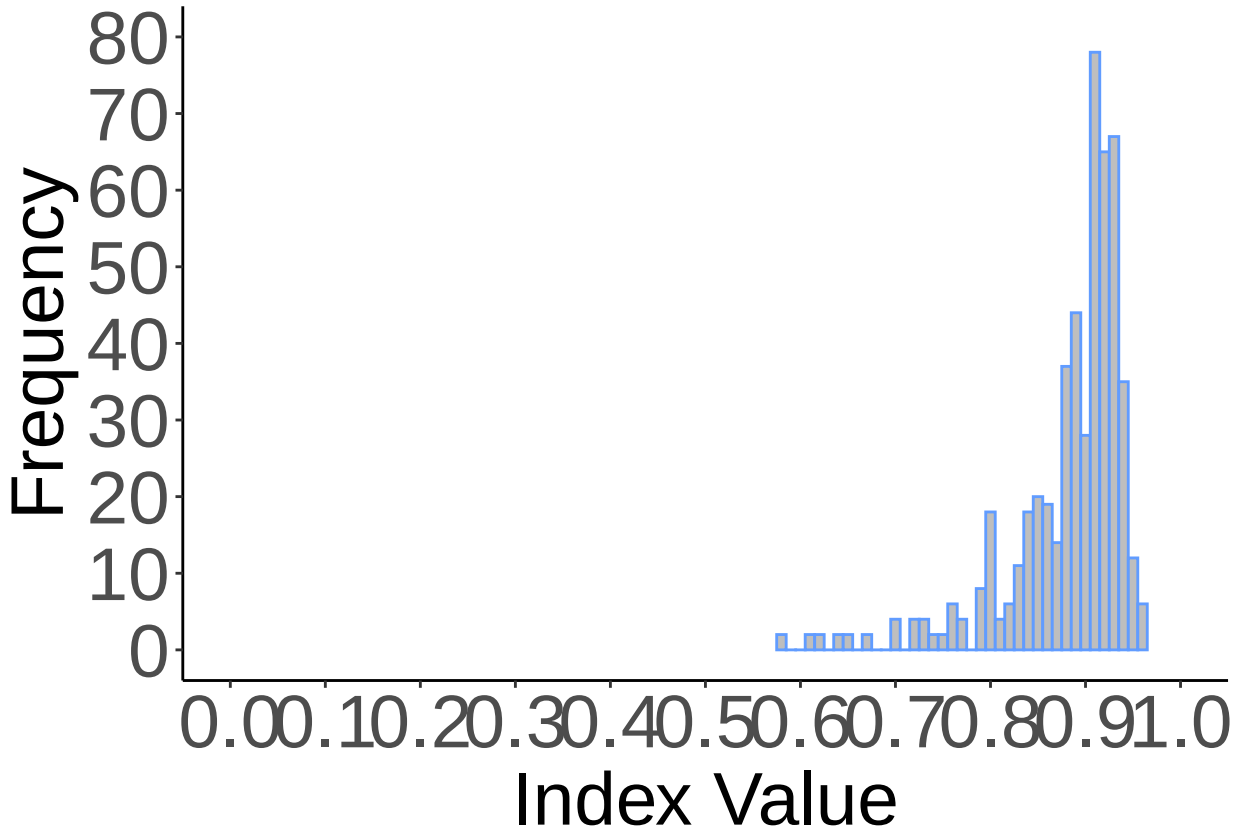
```
hist_data <- paper3_data %>%
  select(Platform , Index) %>%
  group_by(Platform) %>%
  summarise(mean = mean(Index) , sd = sd(Index))

hist_data$mean = round(hist_data$mean , digits = 2)
hist_data$sd = round(hist_data$sd , digits = 2)

#This plot is for the legend
ggplot( data = paper3_data , aes(x = Index)) +
  geom_histogram( data = paper3_data , aes(x = Index , color = Platform) , binwidth = .01 , fill = "grey
  coord_cartesian(xlim = c(0 , 1) , ylim = c(0,80)) +
  scale_y_continuous(breaks = seq(0,100 , by = 10)) +
  scale_x_continuous(breaks = seq(0 , 1 , by = 0.1)) +
  scale_color_discrete( name = "Index" , labels = c("GreenSeeker NDVI" , "sUAS NDVI" , "sUAS NDRE"))
```



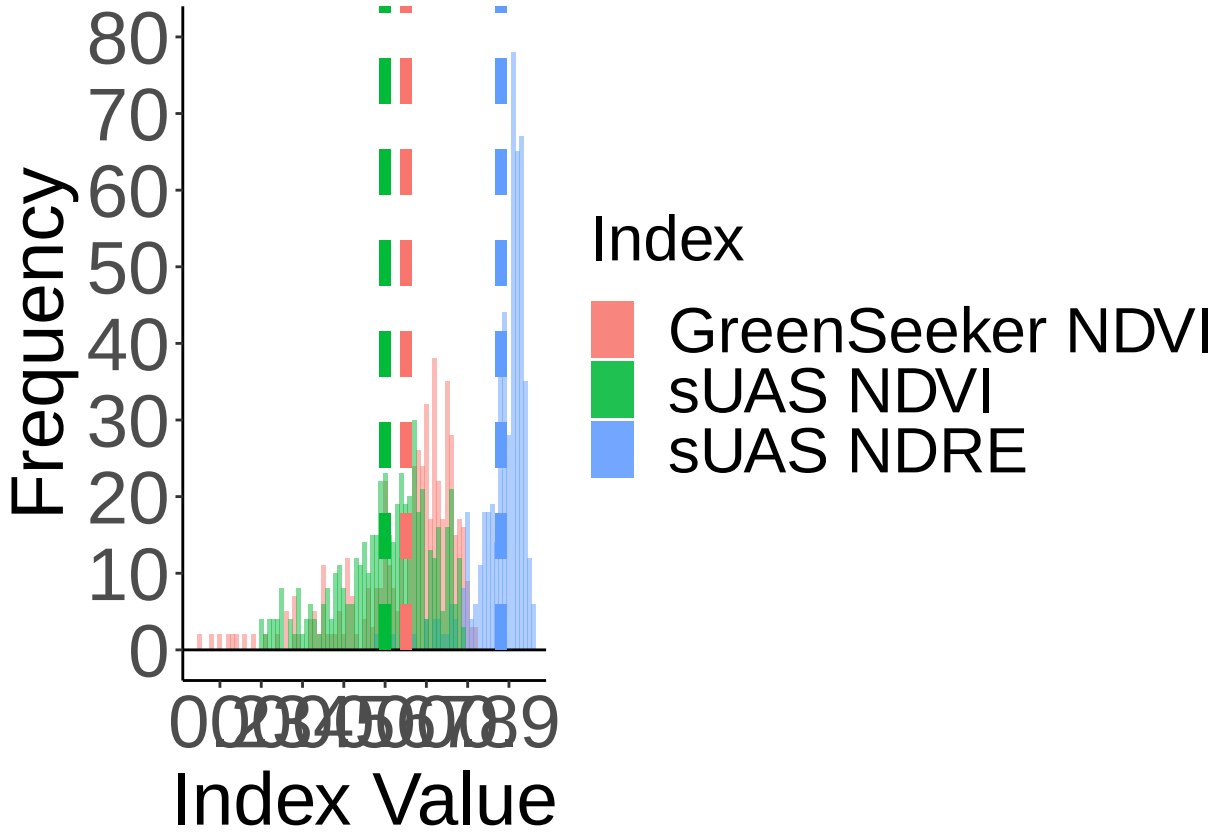

```
gs_ndvi_hist <- ggplot( data = paper3_gsdata , aes(x = Index)) +
  geom_histogram( data = paper3_gsdata , aes(x = Index , color = Platform) , binwidth = .01 , fill = "g
  theme_classic() +
  coord_cartesian(xlim = c(0 , 1) , ylim = c(0,80)) +
  scale_y_continuous(breaks = seq(0,100 , by = 10)) +
  scale_x_continuous(breaks = seq(0 , 1 , by = 0.1)) +
  theme(legend.position = "none") +
  labs(x = "Index Value" , y = "Frequency") +
  theme(axis.title = element_text(size = 28),
        axis.text = element_text(size = 28))

gs_ndvi_hist
```




```
suas_ndvi_hist <- ggplot( data = paper3_uas_ndvi_data , aes(x = Index)) +
  geom_histogram( data = paper3_uas_ndvi_data , aes(x = Index ) , binwidth = .01 , fill = "grey" , colo
  theme_classic() +
  coord_cartesian(xlim = c(0 , 1) , ylim = c(0,80)) +
  scale_y_continuous(breaks = seq(0,100 , by = 10)) +
  scale_x_continuous(breaks = seq(0 , 1 , by = 0.1)) +
  theme(legend.position = "none") +
  labs(x = "Index Value" , y = "Frequency") +
  theme(axis.title = element_text(size = 28),
        axis.text = element_text(size = 28))

suas_ndvi_hist
```




```
Fig2 <- ggplot( data = paper3_data, aes( x = Index)) +
    geom_histogram(data = subset(paper3_data , Platform == 'GreenSeeker_NDVI') , aes(x = Index , fill =
    geom_histogram(data = subset(paper3_data , Platform == 'sUAS_NDVI'), aes(x = Index , fill = Platfor
    geom_histogram(data = subset(paper3_data , Platform == 'sUAS_NDRE'), aes(x = Index , fill = Platfor
  theme_classic() +
  coord_cartesian(xlim = c(0.15 , .95) , ylim = c(0,80)) +
  scale_y_continuous(breaks = seq(0,100 , by = 10)) +
  scale_x_continuous(breaks = seq(0 , 1 , by = 0.1)) +
  labs(x = "Index Value" , y = "Frequency") +
  theme(axis.title = element_text(size = 28),
        axis.text = element_text(size = 28),
        legend.text = element_text(size = 24),
        legend.title = element_text(size = 24)) +
  geom_hline(yintercept = 0) +
  scale_fill_discrete( name = "Index" , labels = c("GreenSeeker NDVI" , "sUAS NDVI" , "sUAS NDRE")) +
  geom_segment(aes(x = 0.65, y = 0, xend = 0.65, yend = Inf), linetype = "dashed" , size  = 2 , alpha =
  geom_segment(aes(x = 0.88, y = 0, xend = 0.88, yend = Inf) ,linetype = "dashed" , size = 2 , alpha = 
  geom_segment(aes(x = 0.60, y = 0, xend = 0.60, yend = Inf), linetype = "dashed" , size = 2 , alpha = 

Fig2
```



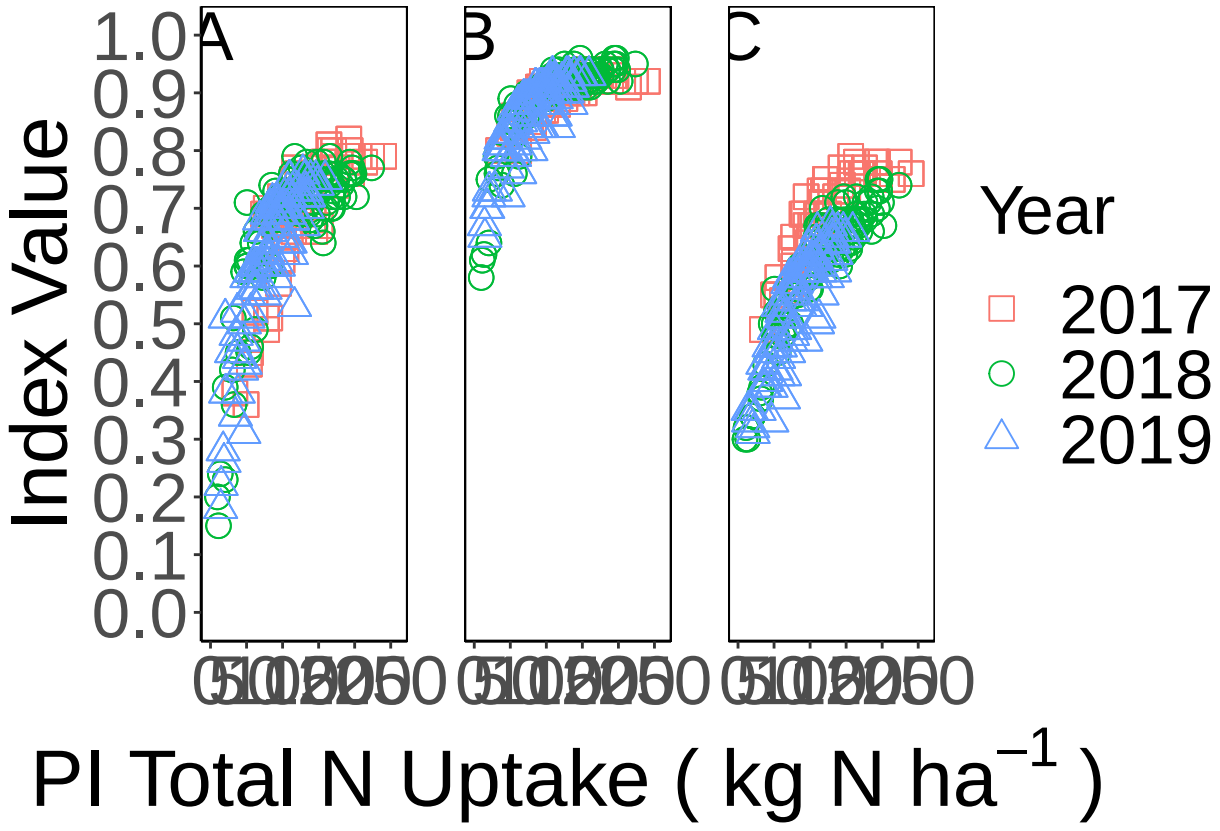

```
ggsave("FIGURES/Paper3_Fig_2.tiff" , Fig2 , compression = "lzw" , width = 16 , height = 12, type = "cai
```

##FIGURE 3

```
platform <- data.frame(
  label = c("NDVI[GS]" ,
            "NDVI[UAS]",
            "NDRE[UAS]"
) ,
  Platform = c("GreenSeeker_NDVI" ,
               "sUAS_NDVI" ,
            "sUAS_NDRE"
            ),
  x_position = c(Inf , Inf , Inf ),
  x_just_var = c( 2.5 , 2.2 , 2) ,
  y_position = c(-Inf , -Inf , -Inf ),
  y_just_var = c(-12 , -12 , -12)
  )

equation <- data.frame(
  label = c(gs_ndvi_qm_eqn,
            sUAS_ndvi_qm_eqn,
            sUAS_ndre_qm_eqn
            ) ,
  Platform = c("GreenSeeker_NDVI" ,
               "sUAS_NDVI" ,
            "sUAS_NDRE"
            ),
```

```
  x_position = c(Inf , Inf , Inf ),
  x_just_var = c(1.05 , 1.05 , 1.05) ,
  y_position = c(-Inf , -Inf , -Inf ),
  y_just_var = c(-8 , -8 , -8)
  )

rsquared <- data.frame(
  label = c("R^2 == '0.80'",
            "R^2 == '0.80'" ,
            "R^2 == '0.82'"
            ) ,
  Platform = c("GreenSeeker_NDVI" ,
               "sUAS_NDVI",
            "sUAS_NDRE"
            ),
  x_position = c(Inf , Inf , Inf ),
  x_just_var = c(2.25 , 2.25 , 2) ,
  y_position = c(-Inf , -Inf , -Inf ),
  y_just_var = c(-8.5 , -8.5 , -8.5)
  )

asymptote <- data.frame(
  label = c("N[UP] == 173~kg~N~ha^-1",
            "N[UP] == 164~kg~N~ha^-1" ,
            "N[UP] == 208~kg~N~ha^-1"
            ) ,
  Platform = c("GreenSeeker_NDVI" ,
               "sUAS_NDVI",
            "sUAS_NDRE"
            ),
  x_position = c(Inf , Inf , Inf ),
  x_just_var = c(1.85 , 1.85 , 1.65) ,
  y_position = c(-Inf , -Inf , -Inf ),
  y_just_var = c(-7.75 , -7.75 , -7.75)
  )

platform$Platform = factor(platform$Platform, levels=c("GreenSeeker_NDVI" , "sUAS_NDVI" , "sUAS_NDRE"))
equation$Platform = factor(equation$Platform, levels=c("GreenSeeker_NDVI" , "sUAS_NDVI" , "sUAS_NDRE"))
rsquared$Platform = factor(rsquared$Platform, levels=c("GreenSeeker_NDVI" , "sUAS_NDVI" , "sUAS_NDRE"))
asymptote$Platform = factor(asymptote$Platform, levels=c("GreenSeeker_NDVI" , "sUAS_NDVI" , "sUAS_NDRE"
gs_ndvi_qm_df$Platform = as.factor(gs_ndvi_qm_df$Platform)
sUAS_ndvi_qm_df$Platform = as.factor(sUAS_ndvi_qm_df$Platform)
sUAS_ndre_qm_df$Platform = as.factor(sUAS_ndre_qm_df$Platform)
asym_data_gs_ndvi <- data.frame(x = gs_ndvi_qm_sym_x , y = gs_ndvi_qm_sym_y , xend = gs_ndvi_qm_sym_x ,
asym_data_sUAS_ndvi <- data.frame(yint = sUAS_ndvi_qm_sym_y , Platform = factor("sUAS_NDVI"))
asym_data_sUAS_ndre <- data.frame(yint = sUAS_ndre_qm_sym_y , Platform = factor("sUAS_NDRE"))

label_text <- data.frame(
  label = c("A" ,
            "B",
            "C"
) ,
  Platform = factor(c("GreenSeeker_NDVI" ,
```

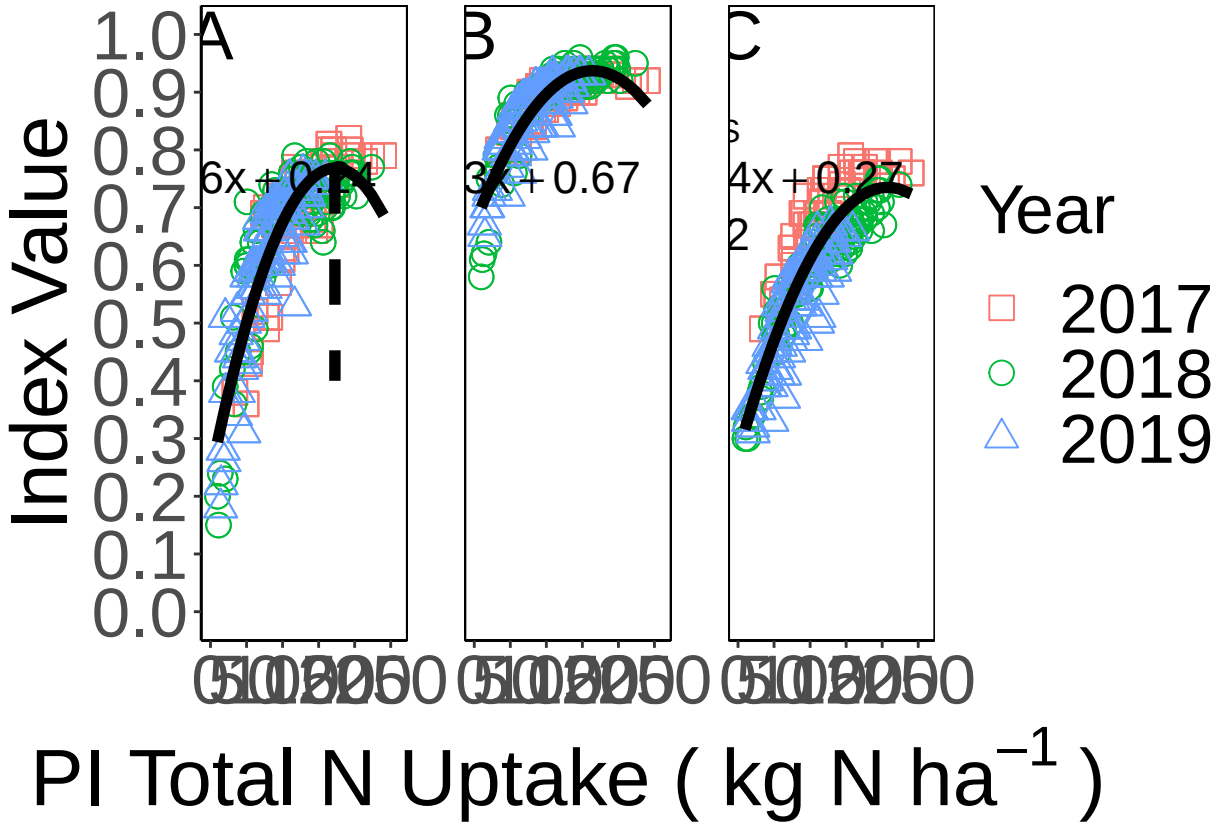
```
              "sUAS_NDVI" ,
            "sUAS_NDRE"
            )),
  x_position = c(0 , 0 , 0 ) ,
  y_position = c(1 , 1 , 1 ))

Nup_fig <- ggplot( data = paper3_data , aes ( x = PI_N_Uptake , y = Index )) +
  geom_point( mapping = aes( x = PI_N_Uptake ,
                             y = Index ,
                             shape = year,
                             color = year) ,
              size = 4 ,
              data = paper3_data) +
  labs( x = "PI Total N Uptake ( kg N ha"^-1~")" ,
        y = "Index Value",
        shape = "Year",
        color = "Year") +
  facet_grid(~Platform ) +
  theme_classic() +
  theme(axis.title = element_text(size = 30),
        axis.text = element_text(size = 26),
        legend.text = element_text(size = 26),
        legend.title = element_text(size = 26),
        #legend.position = "none",
        panel.spacing = unit(1.5, "lines"),
        panel.background = element_rect(fill = "white", color = "grey0"),
        strip.text.x = element_blank()) +
  coord_cartesian(ylim = c(0 , 1) , xlim = c(0, 260)) +
  scale_y_continuous(breaks = seq(0 , 1 , by = .1)) +
  scale_x_continuous(breaks = seq(0 , 250, by = 50)) +
  scale_shape_manual(values = c(0 , 1 , 2)) +
  geom_text( data = label_text ,
             mapping = aes( x = x_position ,
                            y = y_position ,
                            label = label) ,
             size = 9 ,
             parse = T)


Nup_fig
```




```
Fig3 <- Nup_fig +
  geom_line(data = gs_ndvi_qm_df , aes(x = gsndvi_df$PI_N_Uptake , y = gs_ndvi_fit_qm ) , size = 2 , co
  geom_segment(data = asym_data_gs_ndvi , aes(x = x, y = y, xend = xend , yend = yend), linetype = "dash
  geom_line(data = sUAS_ndvi_qm_df , aes(x = sUASndvi_df$PI_N_Uptake , y = sUAS_ndvi_fit_qm ),  size = 1
  geom_line(data = sUAS_ndre_qm_df , aes(x = sUASndre_df$PI_N_Uptake , y = sUAS_ndre_fit_qm ),  size = 1
  geom_text( data = platform ,
             mapping = aes( x = x_position ,
                            hjust = x_just_var ,
                            y = y_position ,
                            vjust = y_just_var ,
                            label = label) ,
             size = 6 ,
             parse = T) +
  geom_text( data = equation ,
             mapping = aes( x = x_position ,
                            hjust = x_just_var ,
                            y = y_position ,
                            vjust = y_just_var ,
                            label = label) ,
             size = 6 ,
             parse = T) +
  geom_text( data = rsquared ,
             mapping = aes( x = x_position ,
                            hjust = x_just_var ,
                            y = y_position ,
                            vjust = y_just_var ,
                            label = label) ,
             size = 6 ,
```

```
            parse = T) +
  geom_text( data = asymptote ,
             mapping = aes( x = x_position ,
                            hjust = x_just_var ,
                            y = y_position ,
                            vjust = y_just_var ,
                            label = label) ,
             size = 6 ,
             parse = T)
```

Fig3




```
ggsave("FIGURES/Paper3_Fig_3.tiff" , Fig3 , compression = "lzw" , width = 19 , height = 9.5, type = "ca
```

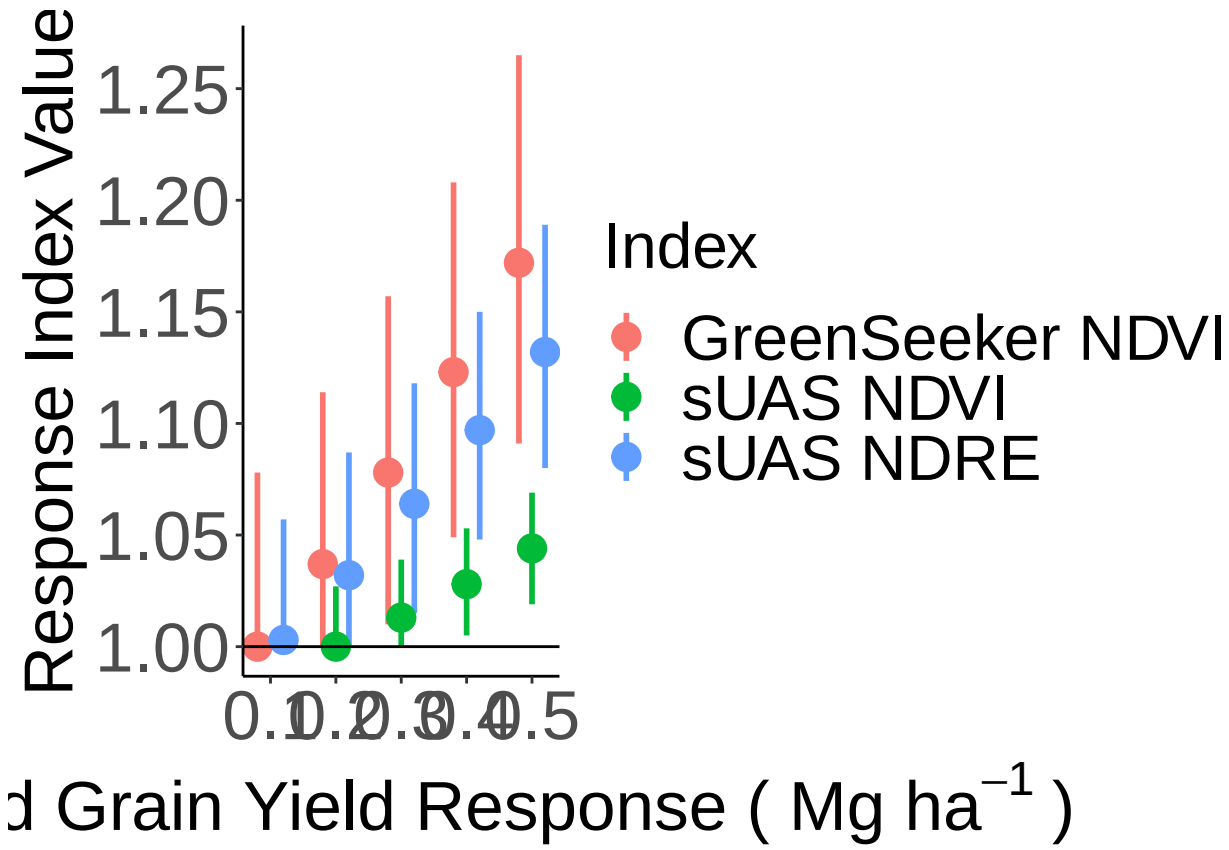
##FIGURE 4

```
estimated_yield_response <- seq(0.1 , 0.5 , by = 0.1)
index <- as.factor("GreenSeeker NDVI")
RI <- c('1.000' , '1.037' , '1.078' , '1.123' , '1.172')
lower_limit <- c('1.000' , '1.000', '1.010' , '1.049' , '1.091')
upper_limit <- c('1.078' , '1.114' , '1.157' , '1.208' , '1.265')
position <- seq(0.08 , 0.48 , by = 0.1)

ci_plot_data1 <- data.frame(index ,
                            estimated_yield_response ,
                            RI ,
                            lower_limit ,
                            upper_limit ,
```

```
                              position)

estimated_yield_response <- seq(0.1 , 0.5 , by = 0.1)
index <- as.factor("sUAS NDVI")
RI <- c(NA , '1.000' , '1.013' , '1.028' , '1.044')
lower_limit <- c(NA , '1.000' , '1.000', '1.005' , '1.019' )
upper_limit <- c(NA , '1.027' , '1.039' , '1.053' , '1.069' )
position <- seq(0.1 , 0.5 , by = 0.1)

ci_plot_data2 <- data.frame(index ,
                              estimated_yield_response ,
                              RI ,
                              lower_limit ,
                              upper_limit ,
                              position)

estimated_yield_response <- seq(0.1 , 0.5 , by = 0.1)
index <- as.factor("sUAS NDRE")
RI <- c('1.003' , '1.032' , '1.064' , '1.097' , '1.132')
lower_limit <- c('1.000' , '1.000' , '1.015', '1.048' , '1.080' )
upper_limit <- c('1.057' , '1.087' , '1.118' , '1.150' , '1.189' )
position <- seq(0.12 , 0.52 , by = 0.1)

ci_plot_data3 <- data.frame(index ,
                              estimated_yield_response ,
                              RI ,
                              lower_limit ,
                              upper_limit ,
                              position)

ci_plot_data <- rbind(ci_plot_data1 , ci_plot_data2 , ci_plot_data3)

ci_plot_data$RI <- as.numeric(ci_plot_data$RI)
ci_plot_data$lower_limit <- as.numeric(ci_plot_data$lower_limit)
ci_plot_data$upper_limit <- as.numeric(ci_plot_data$upper_limit)
```

```
Fig4 <- ggplot(data = ci_plot_data , aes (y = RI , x = estimated_yield_response)) +
  geom_pointrange( data = ci_plot_data , aes(y = RI , x = position , ymin = lower_limit , ymax = upper_l
  theme_classic() +
  scale_y_continuous(breaks = seq(1 , 1.28 , by = 0.05)) +
  geom_hline(yintercept = 1) +
  labs(x = "Estimated Grain Yield Response ( Mg ha"^-1~")" , y = "Response Index Value" , color = "Index
  theme(axis.title = element_text(size = 26),
        axis.text = element_text(size = 26),
        legend.text = element_text(size = 24),
        legend.title = element_text(size = 24)
        )

Fig4
```

```
## Warning: Removed 1 rows containing missing values (geom_pointrange).
```




```
ggsave("FIGURES/Paper3_Fig_4.tiff" , Fig4 , compression = "lzw" , width = 15 , height = 12, type = "cai
```

```
## Warning: Removed 1 rows containing missing values (geom_pointrange).
```