AC_Va_h2_G2_2023

Helen Payne

2024-07-07

```
#load packages
library(lme4)
```

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

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x tidyr::expand() masks Matrix::expand()
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x tidyr::pack()   masks Matrix::pack()
## x tidyr::unpack() masks Matrix::unpack()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(lmerTest)
```

```
##
## Attaching package: 'lmerTest'
##
## The following object is masked from 'package:lme4':
##
##     lmer
##
## The following object is masked from 'package:stats':
##
##     step
```

```
library(car)
```

```
## Loading required package: carData
##
## Attaching package: 'car'
##
## The following object is masked from 'package:dplyr':
##
##     recode
##
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

##########**2023**##########

Read in the data:

```
AC_23 <- read_csv(here::here("data_sheets", "compiled_sheets", "AC_mastersheet_Fitness-mains_2023.csv"))
```

```
## Rows: 545 Columns: 50
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (5): Gen, Donor, Recipient, sample_ID_SEG, SegPos
## dbl  (35): Year, Transect, Sequence, Plant_ID, days_sow2flower, days_plant2f...
## lgl   (2): F_plant, F_Num_03
## date  (8): Sow_Date, Plant_Date, FFD, LFD, F_Num_01, F_Num_02, photo_date, p...
##
## 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.
```

```
AC_23_fit <- AC_23 %>%
  filter(Gen == "G2")

AC_22_23_full <- read_csv(here::here("data_sheets", "compiled_sheets", "AC_22_23_full.csv"))
```

```
## Rows: 6733 Columns: 65
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (7): Gen, Donor, Recipient, sample_ID_SEG, SegPos, Block, Replicated
## dbl  (45): Year, Transect, Sequence, Plant_ID, days_sow2flower, days_plant2f...
## lgl   (4): F_plant, F_Num_03, Rep_FitP, any_FitP
## date  (9): Sow_Date, Plant_Date, FFD, LFD, F_Num_01, F_Num_02, photo_date, p...
##
## 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.
```

```
# Calculate the mean surv_to_flower for each group in AC_22_23_full and add it as a new column
AC_22_23_full <- AC_22_23_full %>%
  group_by(Year, Gen, Transect, Sequence, Donor, Recipient, SegPos) %>%
  mutate(prop_surv_to_flower = mean(surv_to_flower, na.rm = TRUE)) %>%
  ungroup()  # Remove grouping

AC_prop_sample <- AC_22_23_full %>%
  select(c(Year, Gen, Transect, Sequence, Donor, Recipient, SegPos, prop_surv_to_flower))  %>%
    distinct()

AC_23_fit <- AC_23_fit %>%
  left_join(AC_prop_sample %>% select(Year, Gen, Transect, Sequence, Donor, Recipient, SegPos, prop_surv
            by = c("Year", "Gen", "Transect", "Sequence", "Donor", "Recipient"))
```

```
## Warning in left_join(., AC_prop_sample %>% select(Year, Gen, Transect, Sequence, : Detected an unexpe
## i Row 132 of `x` matches multiple rows in `y`.
## i Row 285 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

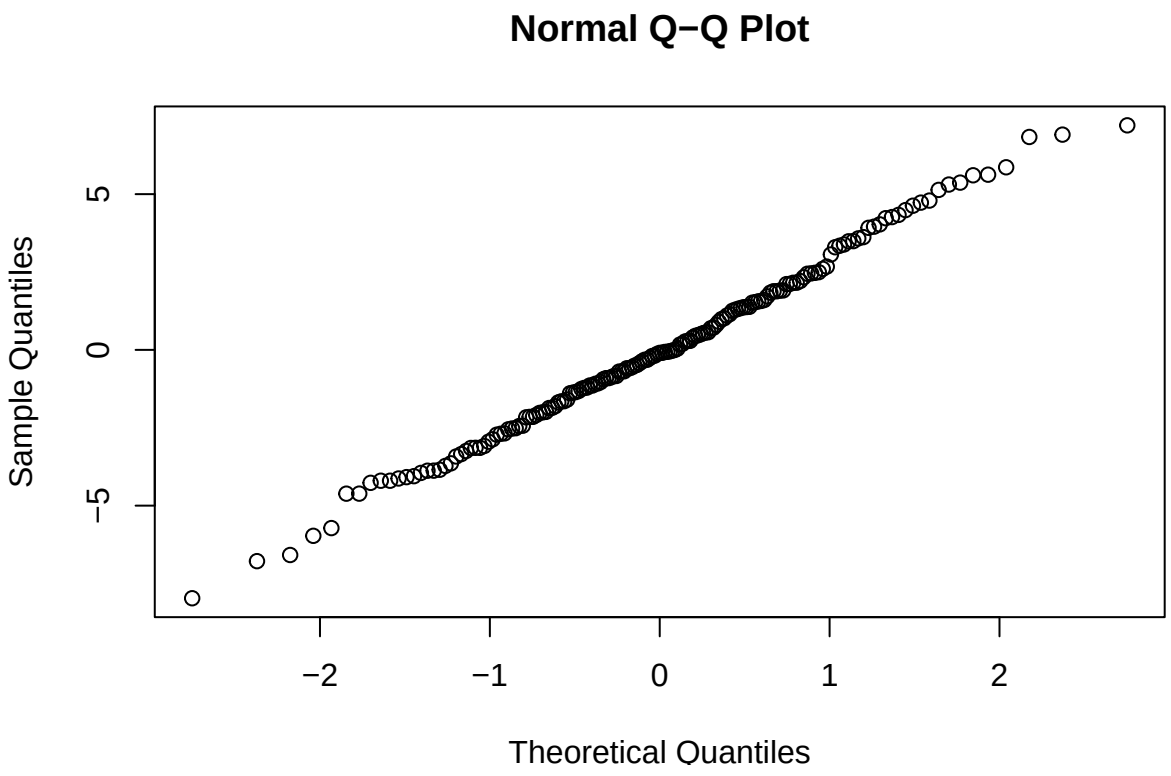
```r
# Create the new column
AC_23_fit <- AC_23_fit %>%
  mutate(est_fitness = prop_surv_to_flower * est_fecundity)
```

```r
#log transform certain variables
AC_23_fit$skel_dryweight_mg_SEG <- log(AC_23_fit$skel_dryweight_mg_SEG)
AC_23_fit$msm_all <- log(AC_23_fit$msm_all)
AC_23_fit$SLA_SEG <- log(AC_23_fit$SLA_SEG)
AC_23_fit$est_fitness <- sqrt(AC_23_fit$est_fitness)


#mean center the traits of interest

traits <- c("corolla_diam_mm_SEG", "skel_dryweight_mg_SEG", "fl_duration", "est_fecundity", "msm_all",

# Mean center each trait
for (trait in traits) {
  trait_mean <- mean(AC_23_fit[[trait]], na.rm = TRUE)
  AC_23_fit[[paste0(trait, "_centered")]] <- AC_23_fit[[trait]] - trait_mean
}
```

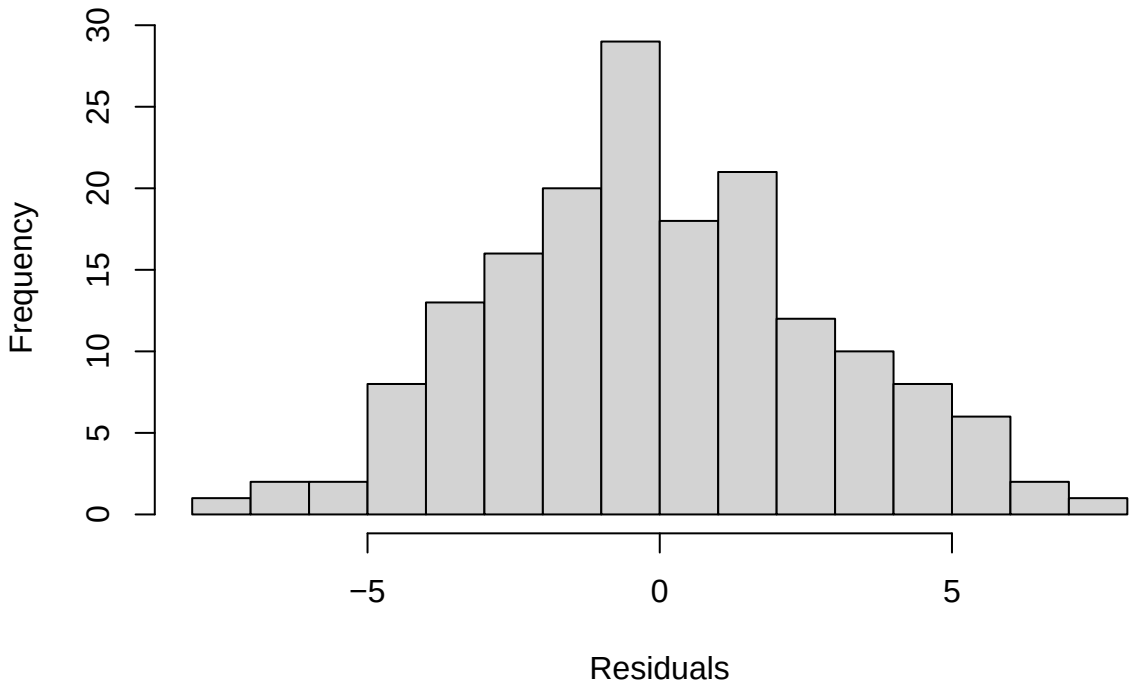
```r
# Create the mixed model for corolla area

#corolla_model <- lmer(corolla_diam_mm_SEG_centered ~ (1 | Recipient) + (1 | Donor), data = AC_23_fit)

corolla_model <- lmer(corolla_diam_mm_SEG_centered ~ (1 | Donor), data = AC_23_fit)

rand(corolla_model)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## corolla_diam_mm_SEG_centered ~ (1 | Donor)
##             npar  logLik    AIC     LRT Df Pr(>Chisq)
## <none>         3 -425.97 857.95
## (1 | Donor)    2 -426.22 856.44 0.49272  1     0.4827
```

```r
# Extract residuals from the model
residuals <- resid(corolla_model)

# Q-Q plot for normality
qqnorm(residuals) #looks good
```

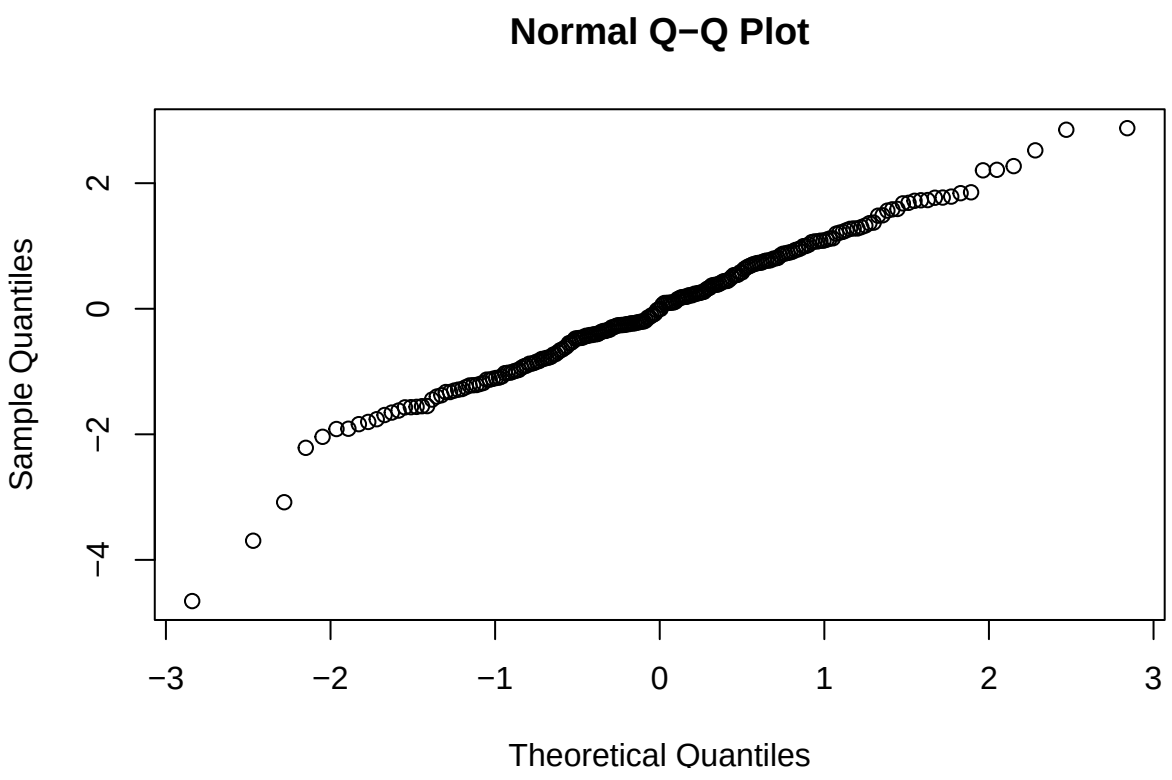



```
# Histogram for normality
hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal
```

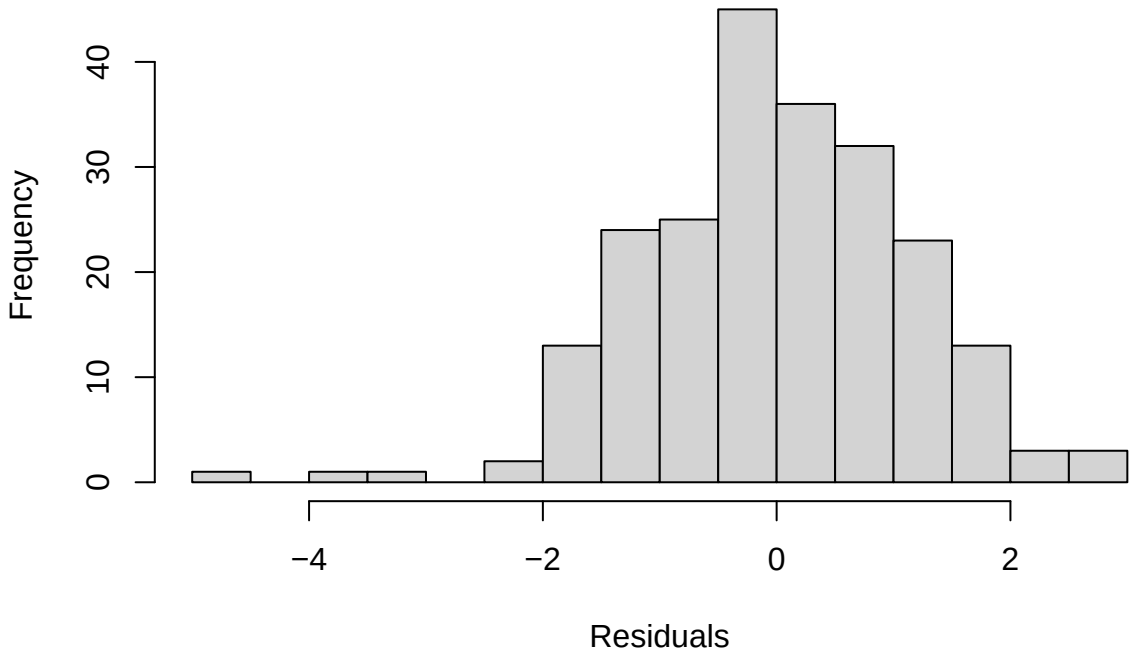



```
# Create the mixed model for skeleton weight, with skeleton weight log transformed
#skel_model <- lmer((skel_dryweight_mg_SEG_centered) ~ (1 | Recipient) + (1 | Donor), data = AC_23_fit)
```

```r
# Extract residuals from the model
#residuals <- resid(skel_model)

# Q-Q plot for normality
#qqnorm(residuals) #looks good


# Histogram for normality
#hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal


# Create the mixed model for skeleton weight, with skeleton weight log transformed
skel_model_2 <- lmer((skel_dryweight_mg_SEG_centered) ~ (1 | Donor), data = AC_23_fit)

# Extract residuals from the model
residuals <- resid(skel_model_2)

# Q-Q plot for normality
qqnorm(residuals) #looks good
```




```r
# Histogram for normality
hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal
```



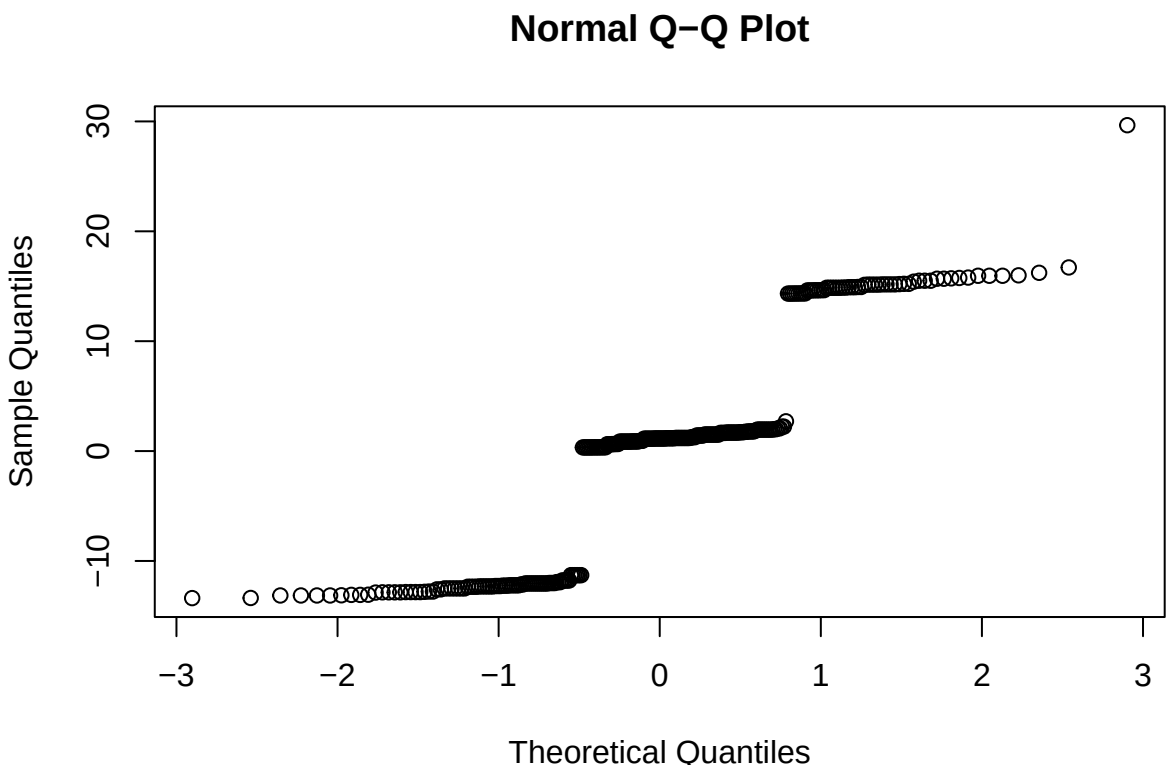

```
rand(skel_model_2)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## (skel_dryweight_mg_SEG_centered) ~ (1 | Donor)
##              npar  logLik    AIC       LRT Df Pr(>Chisq)
## <none>          3 -342.14 690.28
## (1 | Donor)     2 -342.14 688.29 0.0076466  1     0.9303
```

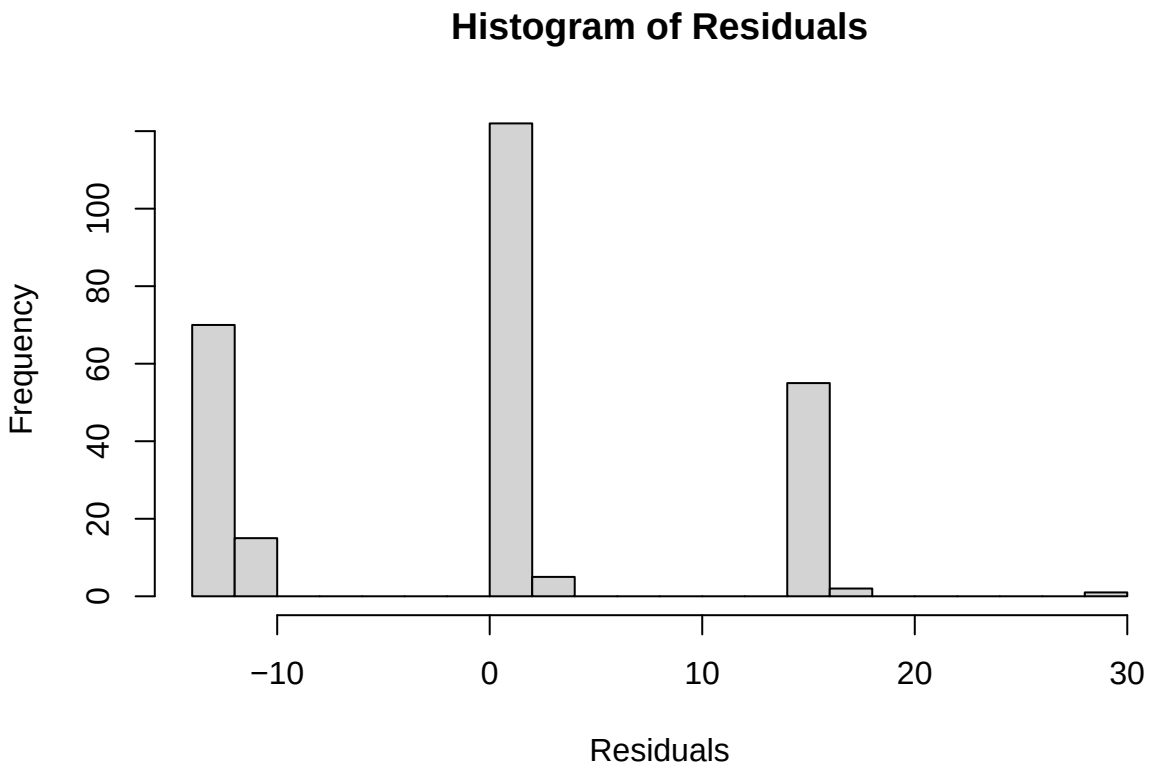
```
# Compare models using AIC, BIC, and log-likelihood
#skel_model_comparison <- data.frame(
  #Model = c("skel_model", "skel_model_2"),
  #AIC = c(AIC(skel_model), AIC(skel_model_2)),
  #BIC = c(BIC(skel_model), BIC(skel_model_2)),
  #LogLikelihood = c(logLik(skel_model), logLik(skel_model_2))
#)

# Print model comparison
#print(skel_model_comparison)

#model 1 is the best
```

```
# Create the mixed model for flowering duration
#fl_duration_model <- lmer(fl_duration_centered ~ (1 | Recipient) + (1 | Donor), data = AC_23_fit) #Sin

fl_duration_model <- lmer(fl_duration_centered ~ (1 | Donor), data = AC_23_fit)

# Extract residuals from the model
residuals <- resid(fl_duration_model)
```

```r
# Q-Q plot for normality
qqnorm(residuals) #looks good
```



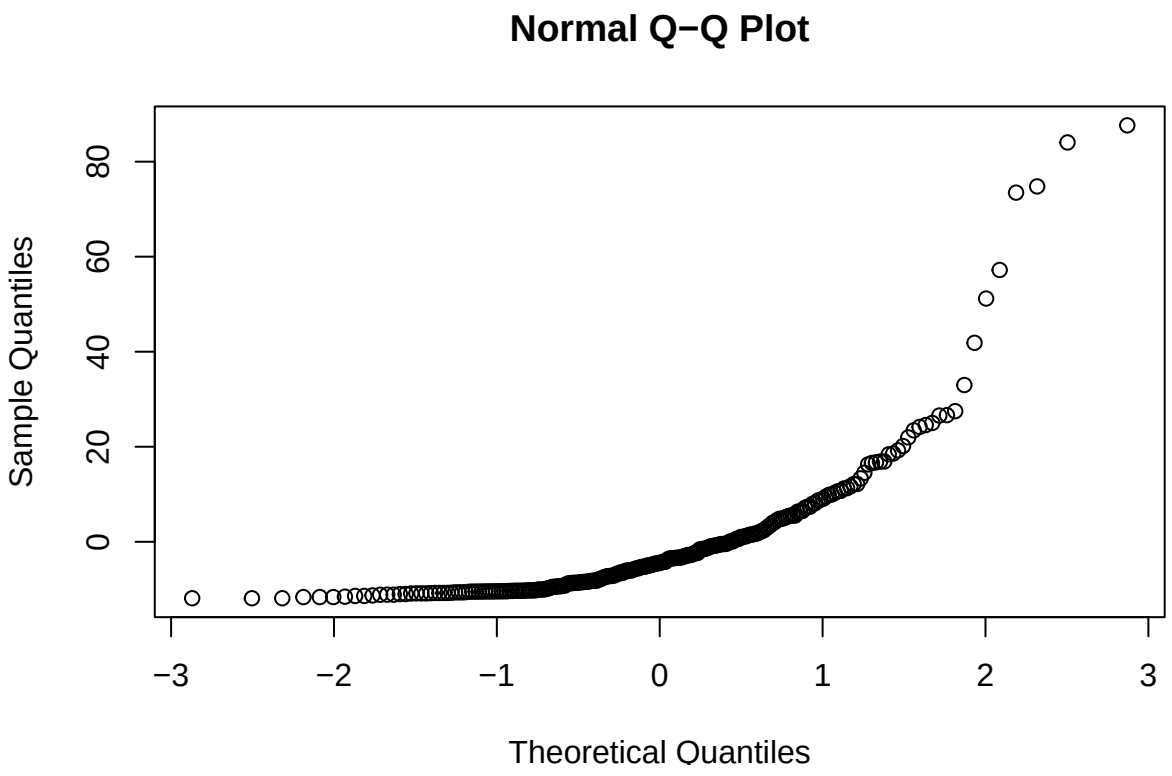

```r
# Histogram for normality
hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal
```



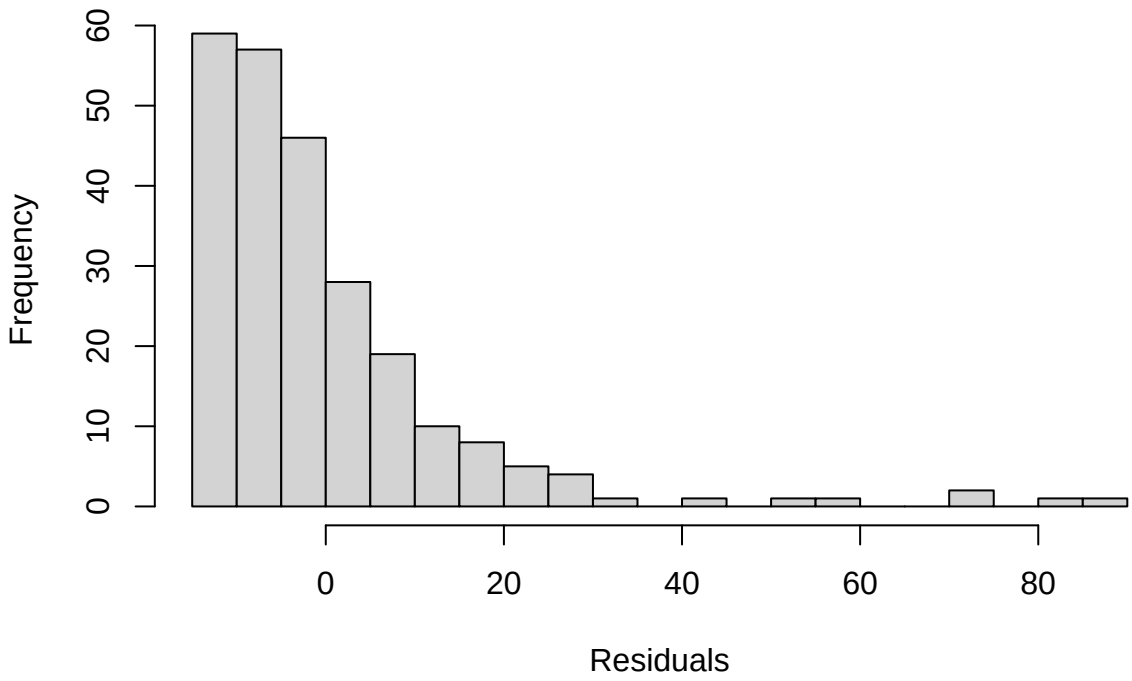
```r
rand(fl_duration_model)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## fl_duration_centered ~ (1 | Donor)
##             npar  logLik    AIC     LRT Df Pr(>Chisq)
## <none>         3 -1009.7 2025.3
## (1 | Donor)    2 -1009.9 2023.8 0.40791  1      0.523
```

```r
# Create the mixed model for estimated fecundity, sqrt transforming estimated fecundity
#est_fecundity_model <- lmer((est_fecundity_centered) ~ (1 | Recipient) + (1 | Donor), data = AC_23_fit

# Extract residuals from the model
#residuals <- resid(est_fecundity_model)

# Q-Q plot for normality
#qqnorm(residuals) #good enough


# Histogram for normality
#hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal-ish


# Create the mixed model for estimated fecundity, sqrt transforming estimated fecundity
est_fecundity_model_2 <- lmer((est_fecundity_centered) ~ (1 | Donor), data = AC_23_fit)

# Extract residuals from the model
residuals <- resid(est_fecundity_model_2)

# Q-Q plot for normality
qqnorm(residuals) #good enough
```




```r
# Histogram for normality
hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal-ish
```



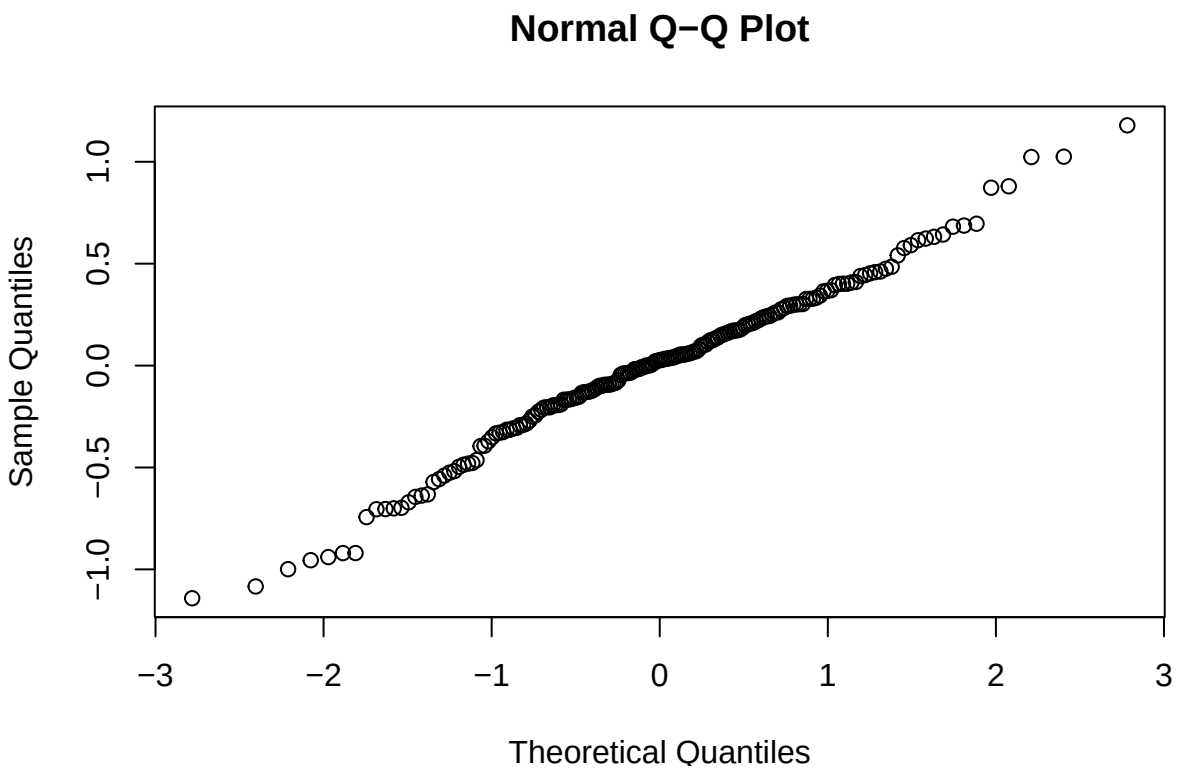

```r
# Compare models using AIC, BIC, and log-likelihood
#est_fecundity_model_comparison <- data.frame(
  #Model = c("est_fecundity_model", "est_fecundity_model_2"),
```

```
  #AIC = c(AIC(est_fecundity_model), AIC(est_fecundity_model_2)),
  #BIC = c(BIC(est_fecundity_model), BIC(est_fecundity_model_2)),
  #LogLikelihood = c(logLik(est_fecundity_model), logLik(est_fecundity_model_2))
#)

rand(est_fecundity_model_2)
```

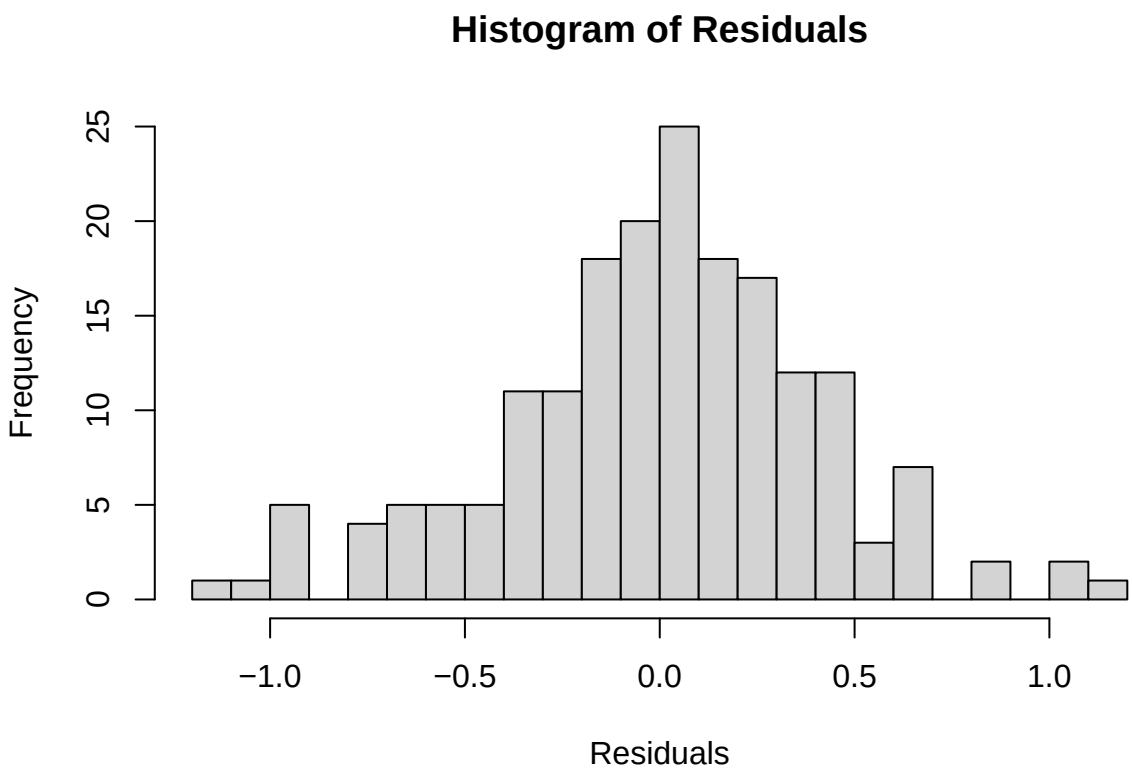
```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## (est_fecundity_centered) ~ (1 | Donor)
##             npar  logLik    AIC     LRT Df Pr(>Chisq)
## <none>         3 -1011.6 2029.2
## (1 | Donor)    2 -1011.7 2027.4 0.14879  1     0.6997
```

```
# Print model comparison
#print(est_fecundity_model_comparison)


#model 2 is the best
```

```
# Create the mixed model for mean seed mass, log transformed mean seed mass
#msm_model <- lmer((msm_all_centered) ~ (1 | Recipient) + (1 | Donor), data = AC_23_fit) #singular

# Extract residuals from the model
#residuals <- resid(msm_model)

# Q-Q plot for normality
#qqnorm(residuals) #good enough


# Histogram for normality
#hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal-ish

# Create the mixed model for mean seed mass, log transformed mean seed mass
msm_model_2 <- lmer((msm_all_centered) ~ (1 | Donor), data = AC_23_fit)

# Extract residuals from the model
residuals <- resid(msm_model_2)

# Q-Q plot for normality
qqnorm(residuals) #good enough
```




```
# Histogram for normality
hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal-ish
```




```
rand(msm_model_2)
```

```
## ANOVA-like table for random-effects: Single term deletions
```

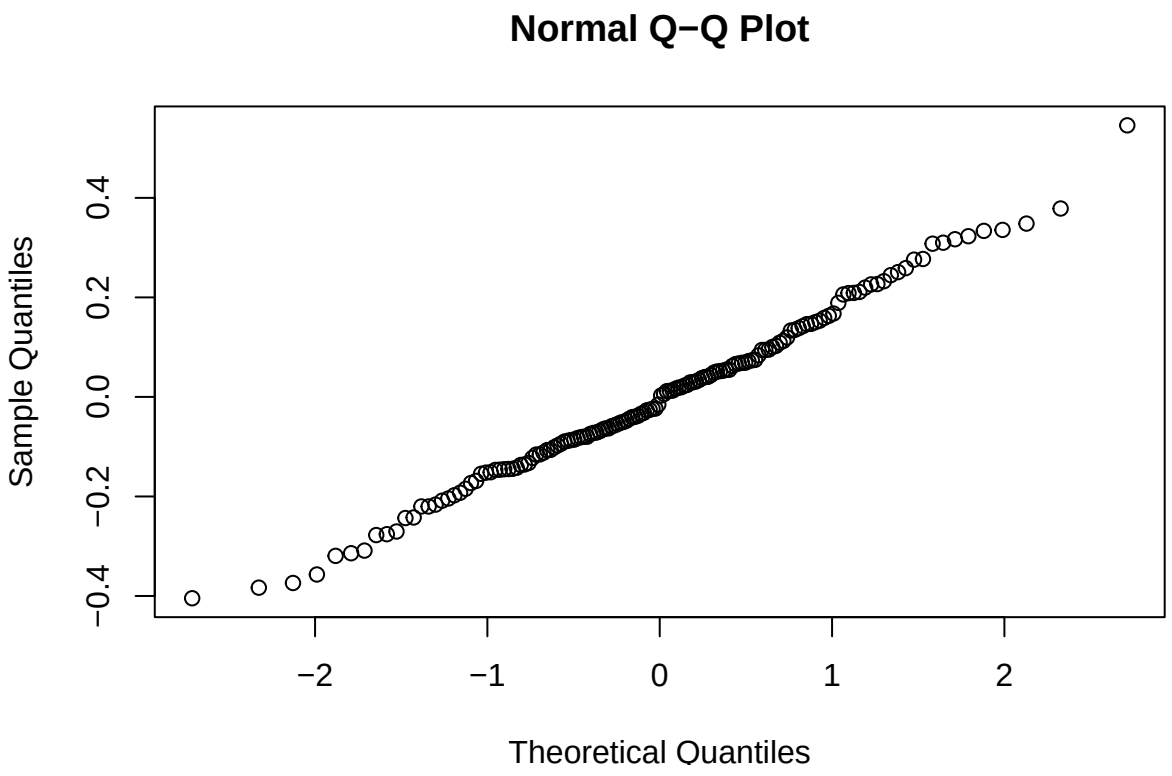
```
##
## Model:
## (msm_all_centered) ~ (1 | Donor)
##             npar  logLik    AIC     LRT Df Pr(>Chisq)
## <none>         3 -104.38 214.76
## (1 | Donor)    2 -104.65 213.31 0.54049  1     0.4622
```

```
# Compare models using AIC, BIC, and log-likelihood
#msm_model_comparison <- data.frame(
  #Model = c("msm_model", "msm_model_2"),
  #AIC = c(AIC(msm_model), AIC(msm_model_2)),
  #BIC = c(BIC(msm_model), BIC(msm_model_2)),
  #LogLikelihood = c(logLik(msm_model), logLik(msm_model_2))
#)

# Print model comparison
#print(msm_model_comparison)

#model 1 is the best
```

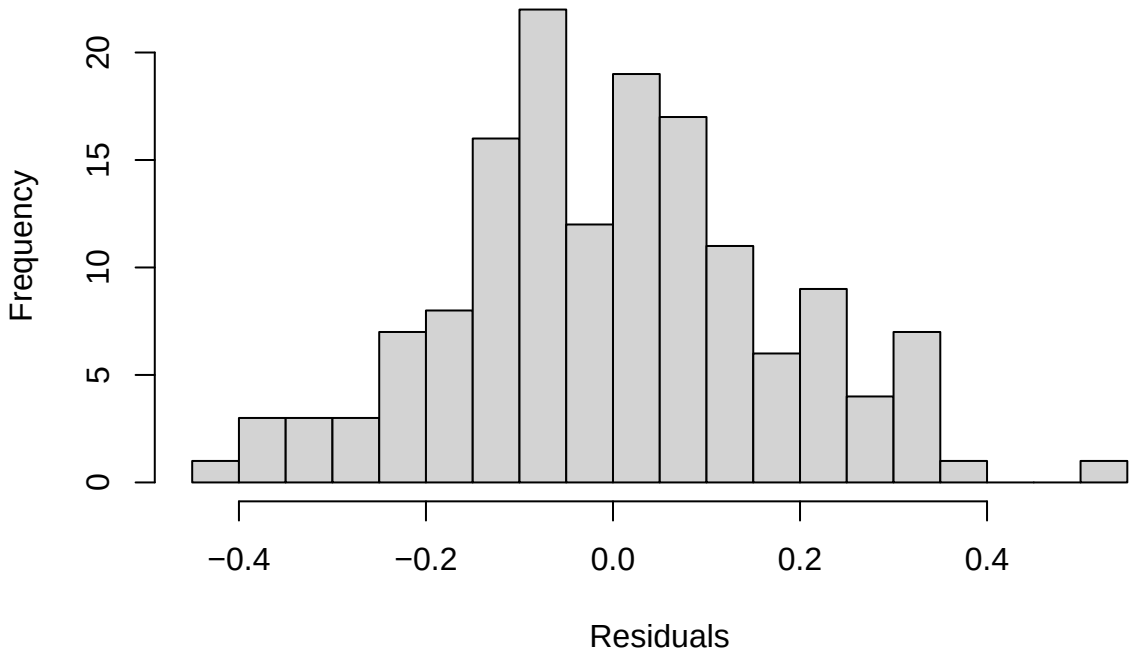
```
# Create the mixed model for SLA
#SLA_model <- lmer((SLA_SEG_centered) ~ (1 | Recipient) + (1 | Donor), data = AC_23_fit) #Singular


##adding in transect##
# Create the mixed model for SLA
SLA_model <- lmer((SLA_SEG_centered) ~ (1 | Transect) + (1 | Donor), data = AC_23_fit)

rand(SLA_model)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## (SLA_SEG_centered) ~ (1 | Transect) + (1 | Donor)
##                 npar logLik     AIC    LRT Df Pr(>Chisq)
## <none>             4 32.390 -56.780
## (1 | Transect)     3 29.308 -52.616 6.1641  1    0.01304 *
## (1 | Donor)        3 31.855 -57.709 1.0709  1    0.30074
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Extract residuals from the model
residuals <- resid(SLA_model)

# Q-Q plot for normality
qqnorm(residuals) #okay..
```



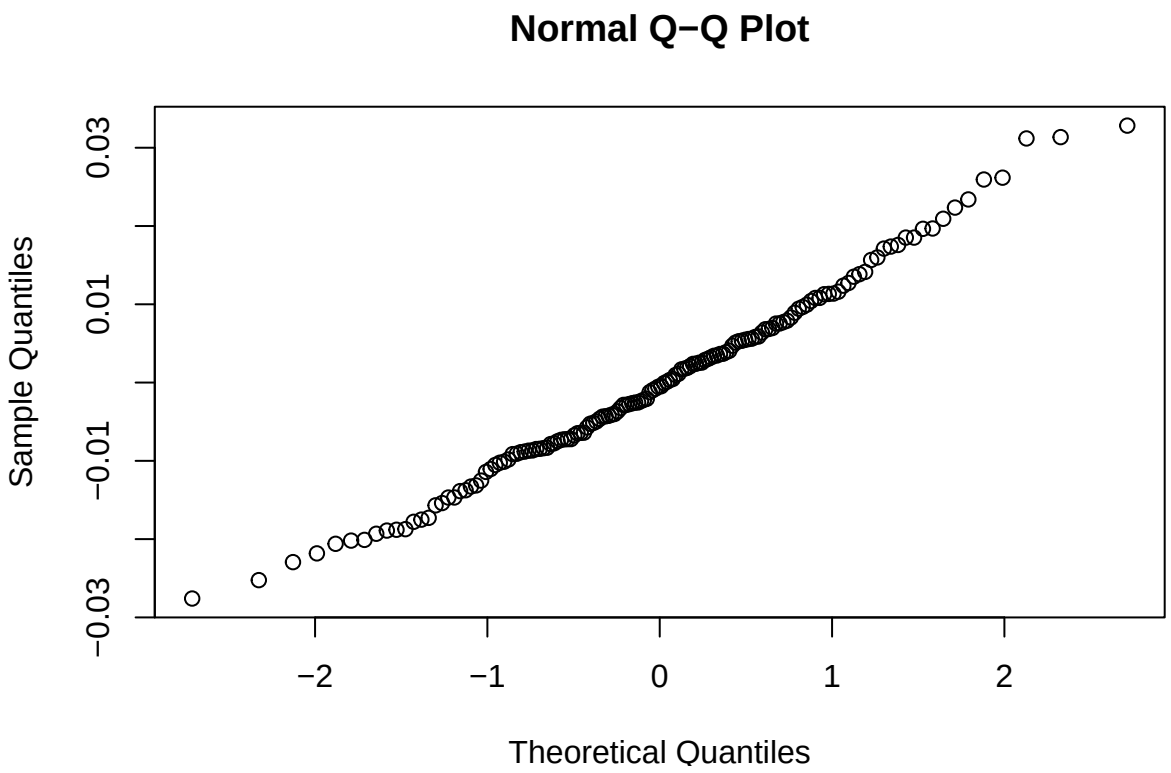

```
# Histogram for normality
hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #definitely a few out
```



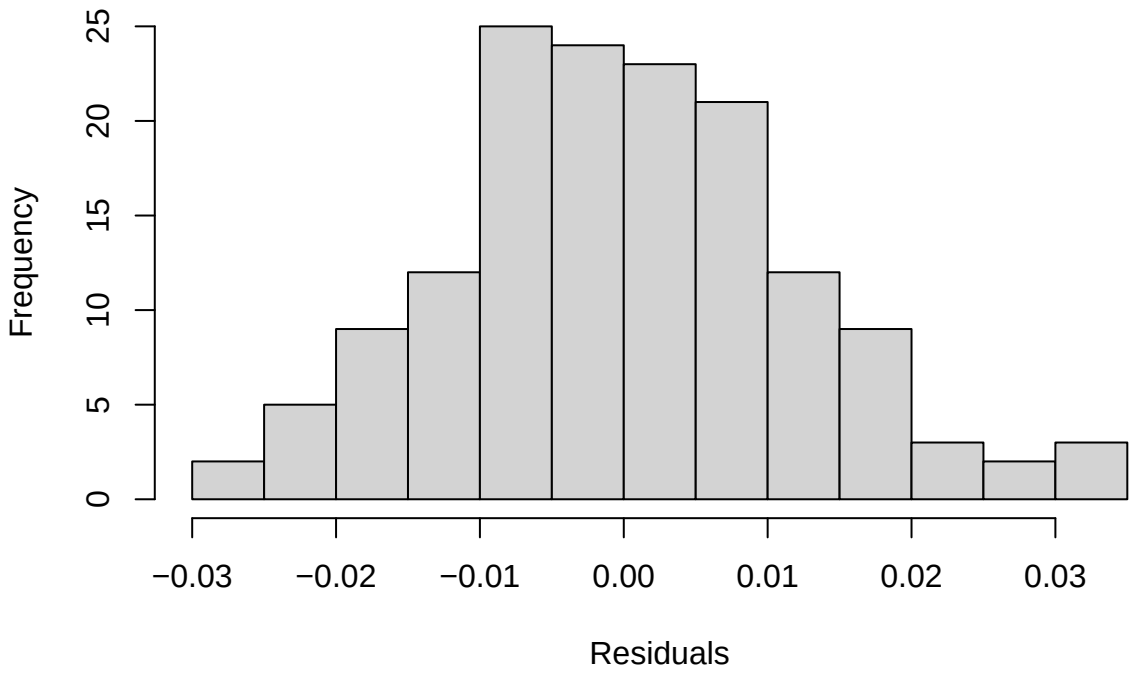

```
# Compare models using AIC, BIC, and log-likelihood
#SLA_model_comparison <- data.frame(
 # Model = c("SLA_model", "SLA_model_2"),
```

```
 # AIC = c(AIC(SLA_model), AIC(SLA_model_2)),
 # BIC = c(BIC(SLA_model), BIC(SLA_model_2)),
 # LogLikelihood = c(logLik(SLA_model), logLik(SLA_model_2))
#)

# Print model comparison
#print(SLA_model_comparison)

#model 2 is the best
```

```
# Create the mixed model for mean seed mass, log transformed mean seed mass
#LMA_model <- lmer(LMA_SEG_centered ~ (1 | Recipient) + (1 | Donor), data = AC_23_fit) #Singular

LMA_model <- lmer(LMA_SEG_centered ~ (1 | Donor), data = AC_23_fit)

# Extract residuals from the model
residuals <- resid(LMA_model)

# Q-Q plot for normality
qqnorm(residuals) #good enough
```




```
# Histogram for normality
hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal-ish
```



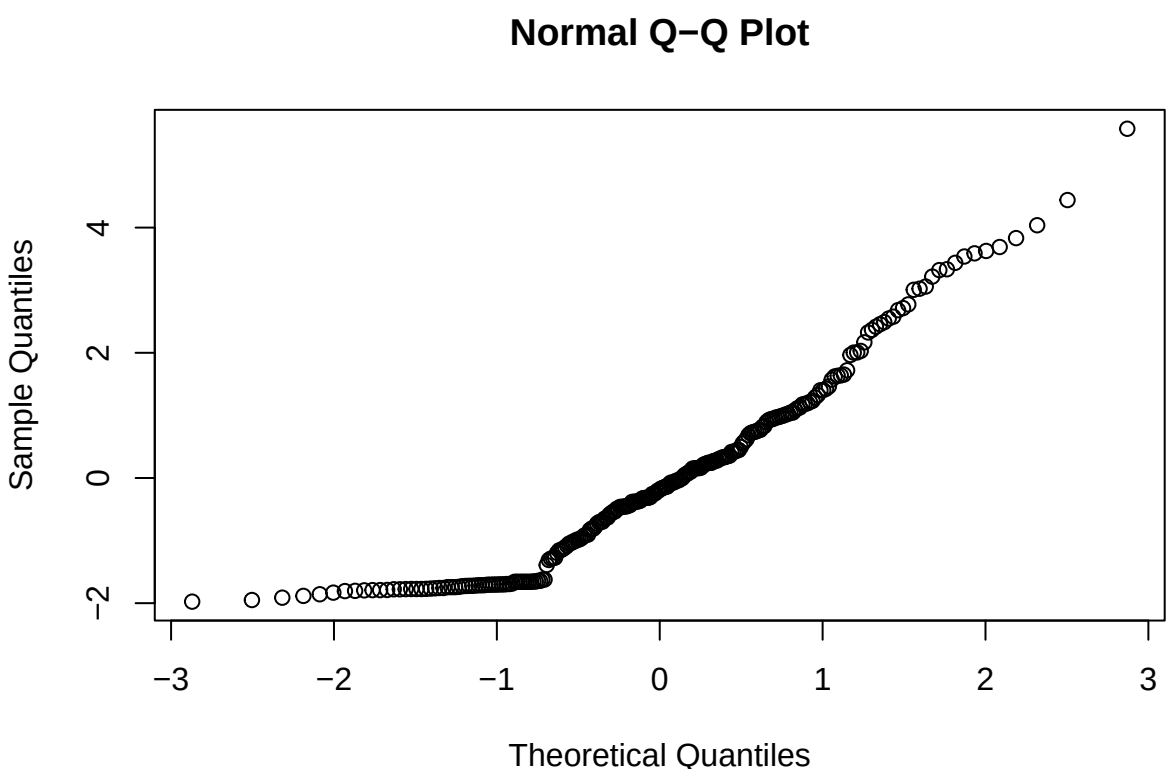

```
# Create the mixed model for d13C, log transformed mean seed mass
#d13C_model <- lmer(d13C_SEG_centered ~ (1 | Recipient) + (1 | Donor), data = AC_23_fit)#Singular

# Extract residuals from the model
#residuals <- resid(d13C_model)

# Q-Q plot for normality
#qqnorm(residuals) #good enough


# Histogram for normality
#hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal-ish

# Create the mixed model for d13C, log transformed mean seed mass
#d13C_model_2 <- lmer(d13C_SEG_centered ~ (1 | Donor), data = AC_23_fit)

# Extract residuals from the model
#residuals <- resid(d13C_model_2)

# Q-Q plot for normality
#qqnorm(residuals) #good enough


# Histogram for normality
#hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal-ish

# Compare models using AIC, BIC, and log-likelihood
#d13C_model_comparison <- data.frame(
  #Model = c("d13C_model", "d13C_model_2"),
  #AIC = c(AIC(d13C_model), AIC(d13C_model_2)),
```

```
  #BIC = c(BIC(d13C_model), BIC(d13C_model_2)),
  #LogLikelihood = c(logLik(d13C_model), logLik(d13C_model_2))
#)

# Print model comparison
#print(d13C_model_comparison)

#model 2 is the best
```

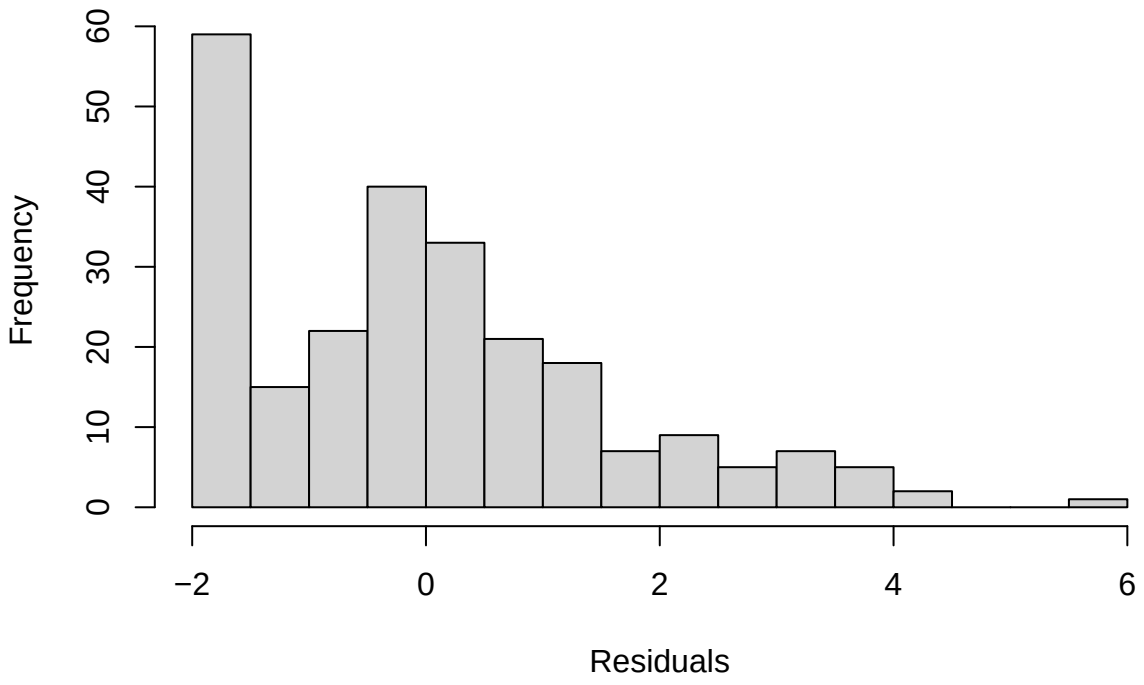
```
# Create the mixed model for d13C, log transformed mean seed mass
est_fitness_model <- lmer(est_fitness_centered ~ (1 | Recipient) + (1 | Donor), data = AC_23_fit)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
# Extract residuals from the model
residuals <- resid(est_fitness_model)

# Q-Q plot for normality
qqnorm(residuals) #good enough
```




```
# Histogram for normality
hist(residuals, breaks = 20, main = "Histogram of Residuals", xlab = "Residuals") #normal-ish
```




```
# Test the significance of the random effects
rand(est_fitness_model) #all Random effects significantly improve the models fit!
```

```
## boundary (singular) fit: see help('isSingular')
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## est_fitness_centered ~ (1 | Recipient) + (1 | Donor)
##                 npar  logLik    AIC     LRT Df Pr(>Chisq)
## <none>             4 -462.64 933.27
## (1 | Recipient)    3 -462.86 931.73 0.45578  1     0.4996
## (1 | Donor)        3 -462.64 931.27 0.00000  1     1.0000
```

```
# Function to round values to a specified number of significant digits
round_df <- function(df, digits) {
  df[] <- lapply(df, function(x) if(is.numeric(x)) signif(x, digits) else x)
  return(df)
}

# Function to extract variance components and calculate required values
calculate_variances <- function(model, trait_name) {
  var_components <- as.data.frame(VarCorr(model))

  # Initialize variables
  V_mat <- NA
  V_sd_mat <- NA
  Va_mat <- NA
  V_pat <- NA
  V_sd_pat <- NA
  Va_pat <- NA
```

```
  res_var <- NA
  Vp <- NA
  Vp_sd <- NA
  h2 <- NA

  # Check if "Recipient" is included as a random effect
  if ("Recipient" %in% var_components$grp) {
    V_mat <- var_components$vcov[var_components$grp == "Recipient"]
    V_sd_mat <- sqrt(V_mat)
    Va_mat <- V_mat * 4
  }

  # Calculate other variance components
  if ("Donor" %in% var_components$grp) {
    V_pat <- var_components$vcov[var_components$grp == "Donor"]
    V_sd_pat <- sqrt(V_pat)
    Va_pat <- V_pat * 16
  }

  if ("Residual" %in% var_components$grp) {
    res_var <- var_components$vcov[var_components$grp == "Residual"]
  }

  # Calculate total phenotypic variance and narrow-sense heritability if components are available
  if (!is.na(Va_mat) & !is.na(Va_pat) & !is.na(res_var)) {
    Vp <- Va_mat + Va_pat + res_var
    Vp_sd <- sqrt(Vp)
    h2 <- Va_pat / Vp  # assumed calculation
  }

  # Extract the number of observations
  n_obs <- nobs(model)

  # Create the dataframe and add the traits column
  df <- data.frame(traits = trait_name, V_mat, V_sd_mat, Va_mat, V_pat, V_sd_pat, Va_pat, Vp, Vp_sd, h2

  # Round the dataframe values to four significant digits
  df <- round_df(df, 4)

  return(df)
}

# Calculate variances for each model and add trait names
corolla_variances <- calculate_variances(corolla_model, 'corolla_diameter')
skel_variances <- calculate_variances(skel_model_2, "skel_biomass_mg")
est_fecundity_variances <- calculate_variances(est_fecundity_model_2, "estimated_fecundity")
msm_variances <- calculate_variances(msm_model_2, "mean_seed_mass")
SLA_variances <- calculate_variances(SLA_model, "SLA")
LMA_variances <- calculate_variances(LMA_model, "LMA")
fl_duration_variances <- calculate_variances(fl_duration_model, "flowering duration")
#d13C_variances <- calculate_variances(d13C_model_2, "delta_C_13")
est_fitness_variances <- calculate_variances(est_fitness_model, "est_fitness")
```

```r
# Combine the results into a single dataframe
variance_AC_2023_G2 <- rbind(
  corolla_variances,
  skel_variances,
  fl_duration_variances,
  LMA_variances,
  SLA_variances,
  msm_variances,
  est_fecundity_variances,
  #d13C_variances,
  est_fitness_variances
)

# Print the dataframe
print(variance_AC_2023_G2)
```

```
##                traits  V_mat V_sd_mat Va_mat     V_pat V_sd_pat    Va_pat    Vp
## 1    corolla_diameter     NA       NA     NA 3.994e-01  0.63200 6.391e+00    NA
## 2     skel_biomass_mg     NA       NA     NA 4.626e-03  0.06802 7.402e-02    NA
## 3  flowering duration     NA       NA     NA 2.323e+00  1.52400 3.716e+01    NA
## 4                 LMA     NA       NA     NA 7.670e-06  0.00277 1.227e-04    NA
## 5                 SLA     NA       NA     NA 2.334e-03  0.04831 3.734e-02    NA
## 6      mean_seed_mass     NA       NA     NA 6.558e-03  0.08098 1.049e-01    NA
## 7 estimated_fecundity     NA       NA     NA 4.293e+00  2.07200 6.869e+01    NA
## 8         est_fitness 0.1342   0.3664 0.5369 0.000e+00  0.00000 0.000e+00 2.987
##   Vp_sd h2 n_obs
## 1    NA NA   169
## 2    NA NA   222
## 3    NA NA   270
## 4    NA NA   150
## 5    NA NA   150
## 6    NA NA   185
## 7    NA NA   244
## 8 1.728  0   244
```

```r
#Save the csv file if you want
write_csv(x = variance_AC_2023_G2, here::here("data_sheets", "compiled_sheets", "AC_Va_h2_R_2023.csv"))
```