Cal Poly Herpetology CURE - Capture Data Analyses

Savannah Weaver

June 2021

Contents

Packages

Background and Goals

This data was collected April - May 2021 during a course-based undergraduate research experience (CURE) in Dr. Emily Taylor's Herpetology class of Spring Quarter 2021 at Cal Poly, San Luis Obispo. This part of the study was conducted to describe the variation of osmoregulation (cutaneous evaporative water loss) and osmotic balance (plasma osmolality and hematocrit) in *Sceloporus occidentalis* and to investigate what drives that variation. Please refer to **doi:** for full details.

Data

Morphometrics and Blood Data

This data was collected upon capture of each lizard.

Variables in this dataframe: - date - collection/capture time for each lizard - individual ID for each lizard - sock ID used to capture each lizard (removed, not relevant to analyses) - SVL = snout-vent length - mass in grams - sex - if female, whether or not gravid (with eggs) - which eye the blood sample was taken from - percent hematocrit = percent of blood that's red blood cells - osmolality = the concentration of solutes in the blood (this is the average of 1-3 replicates) - cloacal temperature at the time of CEWL measurement - processing time for each lizard, when all measurements were finished - hemolyzed = whether or not red blood cells burst and contaminated plasma

Before loading in this data, some incorrectly-measured hematocrit and osmolality were omitted: - hematocrit for individuals 1-16, due to observer error - osmolality for individual 19, due to instrumental error

```r
# load and format data
morpho_blood_dat <- read.csv("./data/Herpetology_Data.csv", # filename
                             na.strings=c("","NA") # fix empty cells
                             ) %>%
  dplyr::mutate(# put date and time together
                collect_date_time = (paste(date, collect_time)),
                # replace some date-time values that have missing times
                collect_date_time = replace(collect_date_time,
                                            collect_date_time == "4/5/21 NA", NA),
                # correctly format date-time variable
                collect_date_time = as.POSIXct(collect_date_time,
                                               format = "%m/%d/%y %H:%M"),
                # correctly format date-only variable
                date = as.Date(date, format = "%m/%d/%y"),
                # correctly format collection time variable
                # format extracts just time after posix adds arbitrary date
                collect_time = (as.POSIXct(collect_time, format = "%H:%M")),
                # correctly format processing time variable
                processing_time = (as.POSIXct(processing_time, format = "%H:%M")),
                # set individual_ID variable as a factor, not numeric
                individual_ID = as.factor(individual_ID),
                # set sex variable as a factor, not character
                sex_M_F = as.factor(sex_M_F),
                # set gravidity variable as a factor, not character
                gravid_Y_N = as.factor(gravid_Y_N),
                # set blood sample eye variable as a factor, not character
                blood_sample_eye = as.factor(blood_sample_eye),
                # set hemolyzed variable as a factor, not character
                hemolyzed = as.factor(hemolyzed),
                # compute holding time as capture time - cloacal measurement time:
                hold_time = as.numeric(processing_time - collect_time)
                ) %>%
  # remove two columns not relevant for statistics
  dplyr::select(-sock_ID, -notes)

# check
summary(morpho_blood_dat)
```

```
##       date             collect_time                     individual_ID
##  Min.   :2021-04-05   Min.   :2021-10-08 10:17:00   1      :  1
##  1st Qu.:2021-04-19   1st Qu.:2021-10-08 12:36:00   2      :  1
##  Median :2021-04-26   Median :2021-10-08 12:48:00   3      :  1
##  Mean   :2021-04-27   Mean   :2021-10-08 12:51:12   4      :  1
##  3rd Qu.:2021-05-10   3rd Qu.:2021-10-08 13:03:00   5      :  1
##  Max.   :2021-05-17   Max.   :2021-10-08 15:57:00   6      :  1
##                       NA's   :3                     (Other):142
##      SVL_mm          mass_g       sex_M_F gravid_Y_N blood_sample_eye
##  Min.   :42.00   Min.   : 2.300   F: 48   N    : 22   both:  2
##  1st Qu.:63.00   1st Qu.: 9.125   M:100   Y    : 26   L   :  4
##  Median :67.00   Median :11.200           NA's:100   R   :142
##  Mean   :64.97   Mean   :10.586
##  3rd Qu.:69.00   3rd Qu.:12.725
##  Max.   :73.00   Max.   :15.000
##
##  hematocrit_percent osmolality_mmol_kg cloacal_temp_C
##  Min.   :16.00      Min.   :293        Min.   :20.00
##  1st Qu.:33.00      1st Qu.:341        1st Qu.:22.00
##  Median :35.00      Median :366        Median :23.00
##  Mean   :35.36      Mean   :365        Mean   :23.48
##  3rd Qu.:38.00      3rd Qu.:387        3rd Qu.:25.00
##  Max.   :54.00      Max.   :436        Max.   :28.00
##  NA's   :27         NA's   :3          NA's   :7
##  processing_time               hemolyzed collect_date_time
##  Min.   :2021-10-08 12:44:00   N    :85   Min.   :2021-04-05 10:17:00
##  1st Qu.:2021-10-08 14:09:00   Y    :39   1st Qu.:2021-04-19 12:49:00
##  Median :2021-10-08 15:17:30   NA's:24   Median :2021-04-26 15:34:00
##  Mean   :2021-10-08 15:12:09             Mean   :2021-04-28 20:28:01
##  3rd Qu.:2021-10-08 16:15:15             3rd Qu.:2021-05-10 12:44:00
##  Max.   :2021-10-08 17:38:00             Max.   :2021-05-17 13:01:00
##  NA's   :8                               NA's   :3
##    hold_time
##  Min.   : 21.0
##  1st Qu.: 95.0
##  Median :141.5
##  Mean   :143.8
##  3rd Qu.:197.5
##  Max.   :268.0
##  NA's   :10
```

```r
unique(morpho_blood_dat$date)
```

```
## [1] "2021-04-05" "2021-04-19" "2021-04-26" "2021-05-03" "2021-05-10"
## [6] "2021-05-17"
```

```r
# get info
morpho_blood_dat %>%
  dplyr::filter(complete.cases(hold_time)) %>%
  summarise(mean_hold_time_minutes = mean(hold_time),
            mean_hold_time_hrs = mean_hold_time_minutes/60)
```

```
##   mean_hold_time_minutes mean_hold_time_hrs
## 1               143.8333           2.397222
```

```r
# export
#write.csv(morpho_blood_dat, "exported_data/capture_hydration.csv")
```

I want to test if any IDs are missing, and which ones if so.

```r
test <- c(seq(1, 150, by = 1))
lost <- test[test %nin% morpho_blood_dat$individual_ID]
lost
```

```
## [1] 23 56
```

Individuals 23 and 56 actually both do not exist because those numbers were skipped when assigning IDs, so we have all the individuals measured in the dataframe.

CEWL Data

First, load it all in and merge.

Variables in this dataframe are: - date - time - date_time combined variable - individual_ID for each lizard measured - region = where on the body CEWL was measured - TEWL_g_m2h = CEWL measurement value in grams/sq-meter/hour - ambient_temp_C = temperature when and where measurement was taken - ambient_RH_percent = relative humidity when and where measurement was taken - e_s_kPa = saturation vapor pressure at a given temperature (calculated using the Clausius-Clapeyron equation from Riddell et al. 2017, cited in the published paper using this data) - e_a_kPa = actual ambient vapor pressure (e_a = e_s * RH proportion) - VPD_kPa = vapor pressure deficit, which is essentially the drying power of the air (VPD = e_s - e_a)

```r
# week 1
CEWL_April_05 <- read.csv("./data/capture_CEWL/4-5-21-CEWL.csv", # filename
                          na.strings=c("","NA")) %>% # fix empty cells
  # rename and select the pertinent variables/cols
  # I have to do this for each one
  # so they all have the same number of columns for joining
  dplyr::select(date = Date,
                Time, Status,
                ID = Comments,
                TEWL_g_m2h = TEWL..g..m2h.., # rename
                ambient_temp_C = AmbT..C., # rename
                ambient_RH_percent = AmbRH....
                )

# week 2
CEWL_April_19 <- read.csv("./data/capture_CEWL/4-19-21-CEWL.csv",
                          na.strings=c("","NA")) %>%
  dplyr::select(date = Date,
                Time, Status,
                ID = Comments,
                TEWL_g_m2h = TEWL..g..m2h.., # rename
                ambient_temp_C = AmbT..C., # rename
                ambient_RH_percent = AmbRH....
                )

# week 3
CEWL_April_26 <- read.csv("./data/capture_CEWL/4-26-21-CEWL.csv",
                          na.strings=c("","NA")) %>%
  dplyr::select(date = Date,
                Time, Status,
                ID = Comments,
                TEWL_g_m2h = TEWL..g..m2h.., # rename
                ambient_temp_C = AmbT..C., # rename
                ambient_RH_percent = AmbRH....
                )

# week 4
CEWL_May_3 <- read.csv("./data/capture_CEWL/5-3-21-CEWL.csv",
                          na.strings=c("","NA")) %>%
  dplyr::select(date = Date,
                Time, Status,
                ID = Comments,
                TEWL_g_m2h = TEWL..g..m2h.., # rename
                ambient_temp_C = AmbT..C., # rename
                ambient_RH_percent = AmbRH....
                )

# week 5
CEWL_May_10 <- read.csv("./data/capture_CEWL/5-10-21-CEWL.csv",
                          na.strings=c("","NA")) %>%
  dplyr::select(date = Date,
                Time, Status,
                ID = Comments,
                TEWL_g_m2h = TEWL..g..m2h.., # rename
                ambient_temp_C = AmbT..C., # rename
                ambient_RH_percent = AmbRH....
                )

# week 6
CEWL_May_17 <- read.csv("./data/capture_CEWL/5-17-21-CEWL.csv",
                          na.strings=c("","NA")) %>%
  dplyr::select(date = Date,
                Time, Status,
                ID = Comments,
                TEWL_g_m2h = TEWL..g..m2h.., # rename
                ambient_temp_C = AmbT..C., # rename
                ambient_RH_percent = AmbRH....
                )

# merge all CEWL datafiles & reformat
CEWL <- CEWL_April_05 %>% # week 1
  # join with weeks 2-6
  rbind(., CEWL_April_19,
        CEWL_April_26,
        CEWL_May_3,
        CEWL_May_10,
        CEWL_May_17
        ) %>%
  # remove any unsuccessful measurements
  dplyr::filter(Status == "Normal") %>%
  # extract individual_ID and region separately from the "ID" variable
  separate(ID, c("individual_ID", "region")) %>%
  # reformat data
  dplyr::mutate(# paste and format date-time variable
                CEWL_date_time = as.POSIXct(paste(date, Time),
                                            format = "%m/%d/%y %I:%M:%S %p"),
                # reformat date only
                date = as.Date(date, format = "%m/%d/%y"),
                # reformat time
                # format extracts just time after posix adds arbitrary date
                # but then it's a character again...
                Time = format(as.POSIXct(Time, format = "%I:%M:%S %p"),
                              format = "%H:%M:%S"),
                # format individual ID as a factor
                individual_ID = as.factor(individual_ID),
                # set body region as a factor variable after getting only the consistent characters due
                region = as.factor(substring(region, 1, 4)),
                # calculate VPD
                ambient_temp_K = ambient_temp_C + 273.15,
                e_s_kPa = 0.611*exp((2500000/461.5)*
                                   ((1/273)-(1/ambient_temp_K))),
                e_a_kPa = e_s_kPa * (ambient_RH_percent/100),
                VPD_kPa = e_s_kPa - e_a_kPa
                ) %>%
  # remove cols not relevant to stats
  dplyr::select(-Status) %>%
  # remove any rows with missing values
  dplyr::filter(complete.cases(.))
summary(CEWL)
```

```
##       date                 Time           individual_ID  region
##  Min.   :2021-04-05   Length:699         01     :  5   dewl:139
##  1st Qu.:2021-04-19   Class  :character  02     :  5   dors:141
##  Median :2021-04-26   Mode   :character  03     :  5   head:141
##  Mean   :2021-04-28                      04     :  5   mite:137
##  3rd Qu.:2021-05-10                      05     :  5   vent:141
##  Max.   :2021-05-17                      06     :  5
##                                          (Other):669
##    TEWL_g_m2h    ambient_temp_C  ambient_RH_percent
##  Min.   : 3.41   Min.   :22.30   Min.   :34.00
##  1st Qu.:17.09   1st Qu.:23.00   1st Qu.:41.30
##  Median :22.00   Median :23.20   Median :45.20
##  Mean   :25.87   Mean   :23.44   Mean   :43.56
##  3rd Qu.:32.59   3rd Qu.:23.80   3rd Qu.:46.30
##  Max.   :96.16   Max.   :25.30   Max.   :53.10
##
##  CEWL_date_time                ambient_temp_K     e_s_kPa         e_a_kPa
##  Min.   :2021-04-05 13:24:15   Min.   :295.4   Min.   :2.760   Min.   :0.9779
##  1st Qu.:2021-04-19 14:07:34   1st Qu.:296.1   1st Qu.:2.882   1st Qu.:1.2086
##  Median :2021-04-26 17:10:23   Median :296.4   Median :2.918   Median :1.3315
##  Mean   :2021-04-28 23:39:45   Mean   :296.6   Mean   :2.964   Mean   :1.2910
##  3rd Qu.:2021-05-10 16:03:10   3rd Qu.:296.9   3rd Qu.:3.028   3rd Qu.:1.3948
##  Max.   :2021-05-17 17:22:31   Max.   :298.4   Max.   :3.318   Max.   :1.4956
##
##     VPD_kPa
##  Min.   :1.297
##  1st Qu.:1.541
##  Median :1.683
##  Mean   :1.673
##  3rd Qu.:1.779
##  Max.   :2.055
##
```

Write CEWL dataframe as a csv for use in other analyses:

```
#write.csv(CEWL, "exported_data/capture_CEWL.csv")
```

Weather Data

This data was obtained from http://www.itrc.org/databases/precip/ (Adcon Server Data) to test the effect of ambient conditions on CEWL. This is different from the ambient conditions already measured with CEWL, which are the temperature and humidity around the measurement device at the time of measurement. We think that the temperature, humidity, wind speed, and solar radiation the lizard was exposed to prior to capture may also affect CEWL.

We didn't have a daylight savings time switchover during this study, so we don't need to worry about incorporating.

```
# load in csvs and put all in one dataframe
weather <- read.csv("./data/weather/4_5Weather.csv", sep = ';') %>%
  rbind(read.csv("./data/weather/4_19Weather.csv", sep = ';')) %>%
  rbind(read.csv("./data/weather/5_3Weather.csv", sep = ';')) %>%
  rbind(read.csv("./data/weather/5_10Weather.csv", sep = ';')) %>%
  rbind(read.csv("./data/weather/5_17Weather.csv", sep = ';')) %>%
  # add a variable for combined date-time
  mutate(collect_date_time = as.POSIXct(paste(Date, Time),
                                format = "%m/%d/%y %I:%M:%S %p")) %>%
  # remove lonely date and time
  dplyr::select(-Date, -Time)
```

The weather data is only every 15 minutes, but I want to match it to any minute measurement, so I need to interpolate the values for each minute.

First, make a separate dataframe with every minute for each of those days.

```
all_times <- data.frame(collect_date_time = c(# April 5
                           seq(from = as.POSIXct("2021-04-05 10:00"),
                               to = as.POSIXct("2021-04-05 16:00"),
                               by="min"),
                           # April 19
                           seq(from = as.POSIXct("2021-04-19 10:00"),
                               to = as.POSIXct("2021-04-19 16:00"),
                               by="min"),
                           # April 26
                           seq(from = as.POSIXct("2021-04-26 10:00"),
                               to = as.POSIXct("2021-04-26 16:00"),
                               by="min"),
                           # May 3
                           seq(from = as.POSIXct("2021-05-03 10:00"),
                               to = as.POSIXct("2021-05-03 16:00"),
                               by="min"),
                           # May 10
                           seq(from = as.POSIXct("2021-05-10 10:00"),
                               to = as.POSIXct("2021-05-10 16:00"),
                               by="min"),
                           # May 17
                           seq(from = as.POSIXct("2021-05-17 10:00"),
                               to = as.POSIXct("2021-05-17 16:00"),
                               by="min")
                           ))
```

Next, merge the weather data into the times dataframe and interpolate the temperature and humidity between measurements.

```r
all_times_weather <- all_times %>% # time only dataframe
  # add weather measurements based on matching date-time
  left_join(weather, by = 'collect_date_time') %>%
  # convert temperature units, thanks America
  mutate(temp_C = fahrenheit.to.celsius(Temperature_F, round = 2),
         # interpolate temperatures
         temp_C_interpol = na.approx(temp_C),
         # interpolate humidities
         RH_percent_interpol = na.approx(RH_percent),
         # interpolate Wind Speeds
         Wind_mph_interpol = na.approx(Wind_Speed_mph),
         # interpolate solar radiation
         Solar_rad_Wm2_interpol = na.approx(Pyranometer_W_m),
         # compute  VPD
         temp_K_interpol = temp_C_interpol + 273.15,
         e_s_kPa_int = 0.611*exp((2500000/461.5)*
                                   ((1/273)-(1/temp_K_interpol))),
         e_a_kPa_int = e_s_kPa_int * (RH_percent_interpol/100),
         VPD_kPa_int = e_s_kPa_int - e_a_kPa_int
         ) %>%
  # keep only the relevant variables
  dplyr::select(collect_date_time,
                temp_C_interpol,
                RH_percent_interpol,
                VPD_kPa_int,
                Wind_mph_interpol,
                Solar_rad_Wm2_interpol)
summary(all_times_weather)
```

```
##  collect_date_time             temp_C_interpol RH_percent_interpol
##  Min.   :2021-04-05 10:00:00   Min.   :13.28   Min.   :38.20
##  1st Qu.:2021-04-19 13:00:15   1st Qu.:16.54   1st Qu.:56.77
##  Median :2021-04-30 01:00:00   Median :17.78   Median :67.65
##  Mean   :2021-04-28 21:00:00   Mean   :18.78   Mean   :65.52
##  3rd Qu.:2021-05-10 12:59:45   3rd Qu.:20.48   3rd Qu.:72.30
##  Max.   :2021-05-17 16:00:00   Max.   :25.78   Max.   :92.10
##   VPD_kPa_int     Wind_mph_interpol Solar_rad_Wm2_interpol
##  Min.   :0.1224   Min.   :0.100     Min.   : 356.9
##  1st Qu.:0.5578   1st Qu.:4.340     1st Qu.: 743.2
##  Median :0.6430   Median :4.567     Median : 882.6
##  Mean   :0.8248   Mean   :4.574     Mean   : 860.2
##  3rd Qu.:1.0401   3rd Qu.:5.020     3rd Qu.: 979.5
##  Max.   :2.1079   Max.   :7.100     Max.   :1037.5
```

Compute Scaled Mass Index

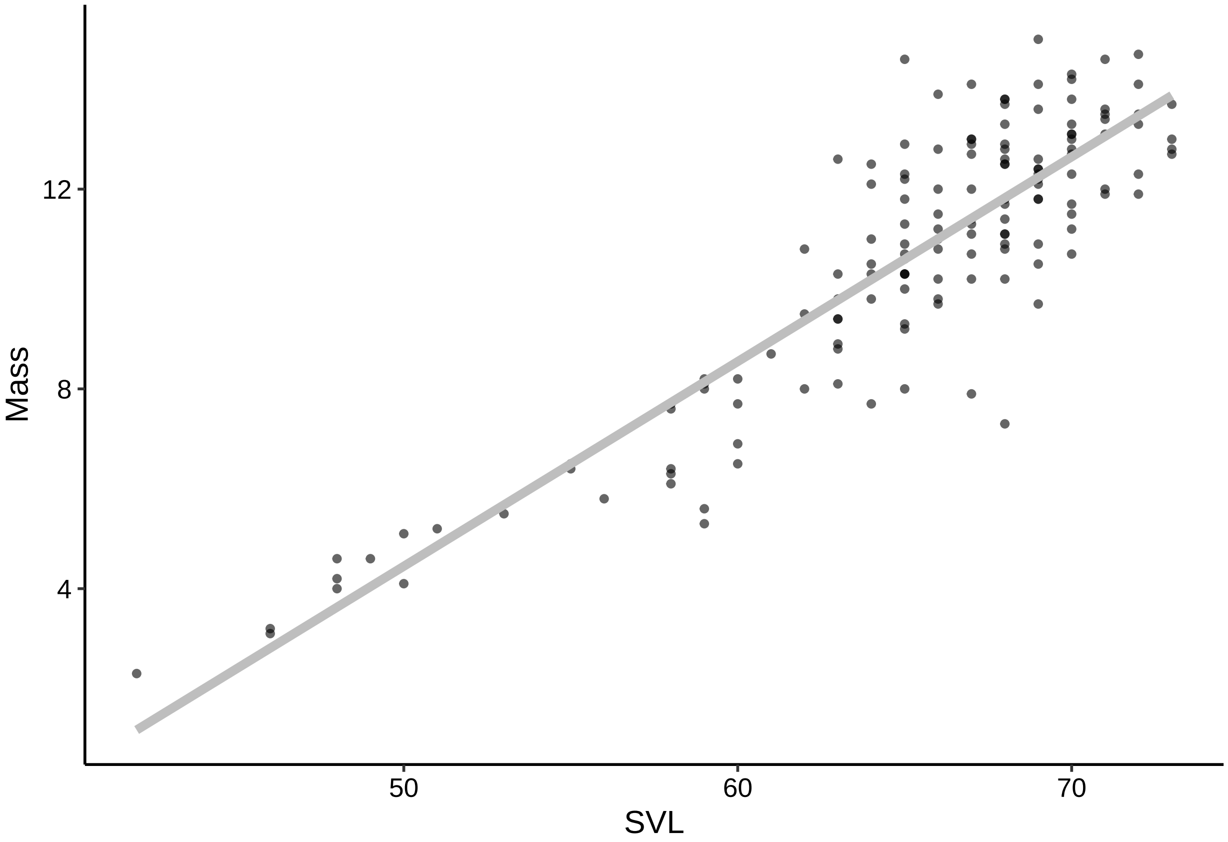
This is also known as the body condition index, or log-log residuals.

I calculate as described by: Peig, J., & Green, A. J. (2009). New perspectives for estimating body condition from mass/length data: The scaled mass index as an alternative method. Oikos, 118(12), 1883–1891. https://doi.org/10.1111/j.1600-0706.2009.17643.x

Step 1: mass ~ SVL

plot:

```
morpho_blood_dat %>%
  ggplot(data = .) +
  geom_point(aes(x = SVL_mm,
                 y = mass_g,
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = SVL_mm,
                  y = mass_g,
                  ),
              formula = y ~ x,
              method = "lm",
              color = "gray",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("SVL") +
  ylab("Mass") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0
)
```



create a simple linear regression

```
mass_SVL_SLR <- lm(data = morpho_blood_dat, mass_g ~ SVL_mm)
summary(mass_SVL_SLR)
```

```
##
## Call:
## lm(formula = mass_g ~ SVL_mm, data = morpho_blood_dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5265 -0.8762 -0.0024  0.6735  4.0031
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -16.04514    1.14303  -14.04   <2e-16 ***
## SVL_mm        0.40988    0.01751   23.40   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.316 on 146 degrees of freedom
## Multiple R-squared:  0.7895, Adjusted R-squared:  0.7881
## F-statistic: 547.7 on 1 and 146 DF,  p-value: < 2.2e-16
```

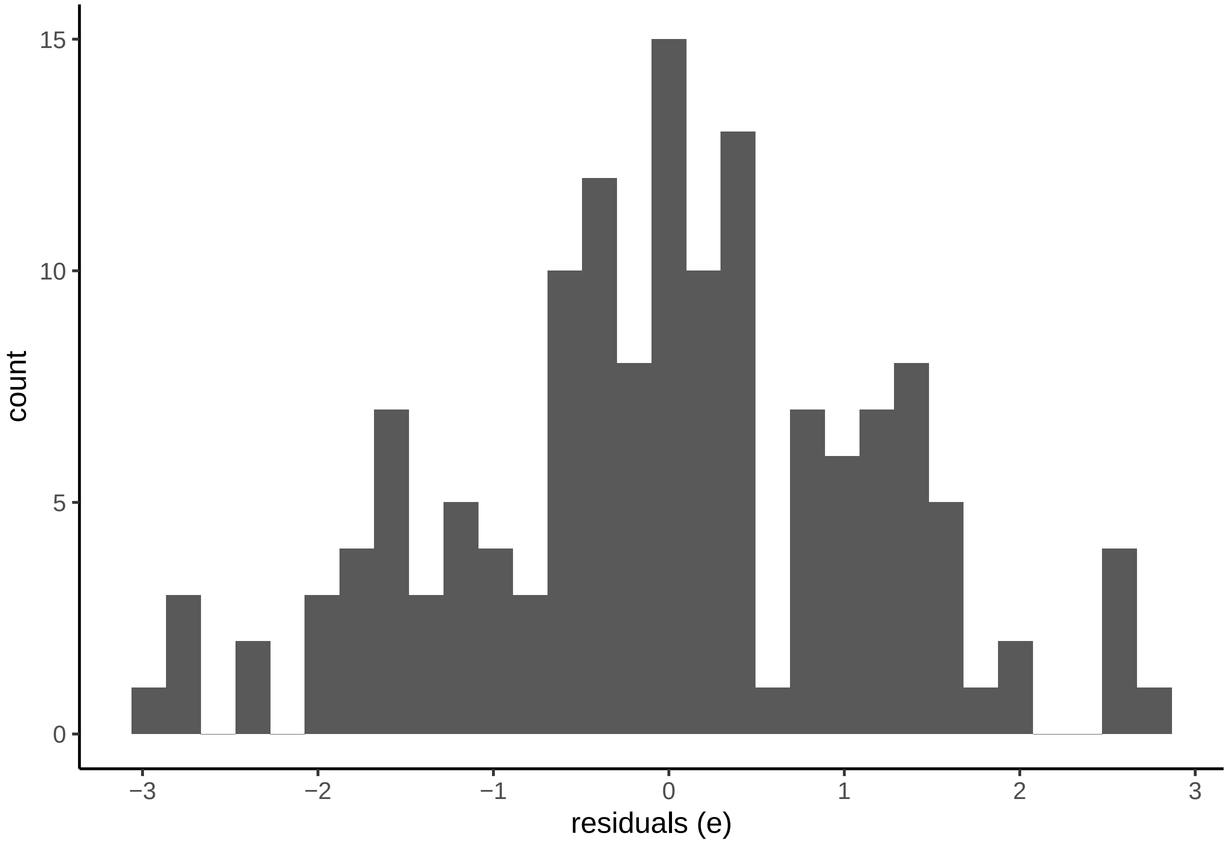
look for outliers by calculating residuals

```
mass_SVL_SLR_residuals <- morpho_blood_dat %>%
  # not working with function, so just compute by hand...
  mutate(#y_hat = predict(mass_SVL_SLR),
         #e = residuals(mass_SVL_SLR),
         y_hat = 0.40988*SVL_mm - 16.04514,
         e = y_hat - mass_g
         ) %>%
  dplyr::filter(e < 3 & e > -3)
```

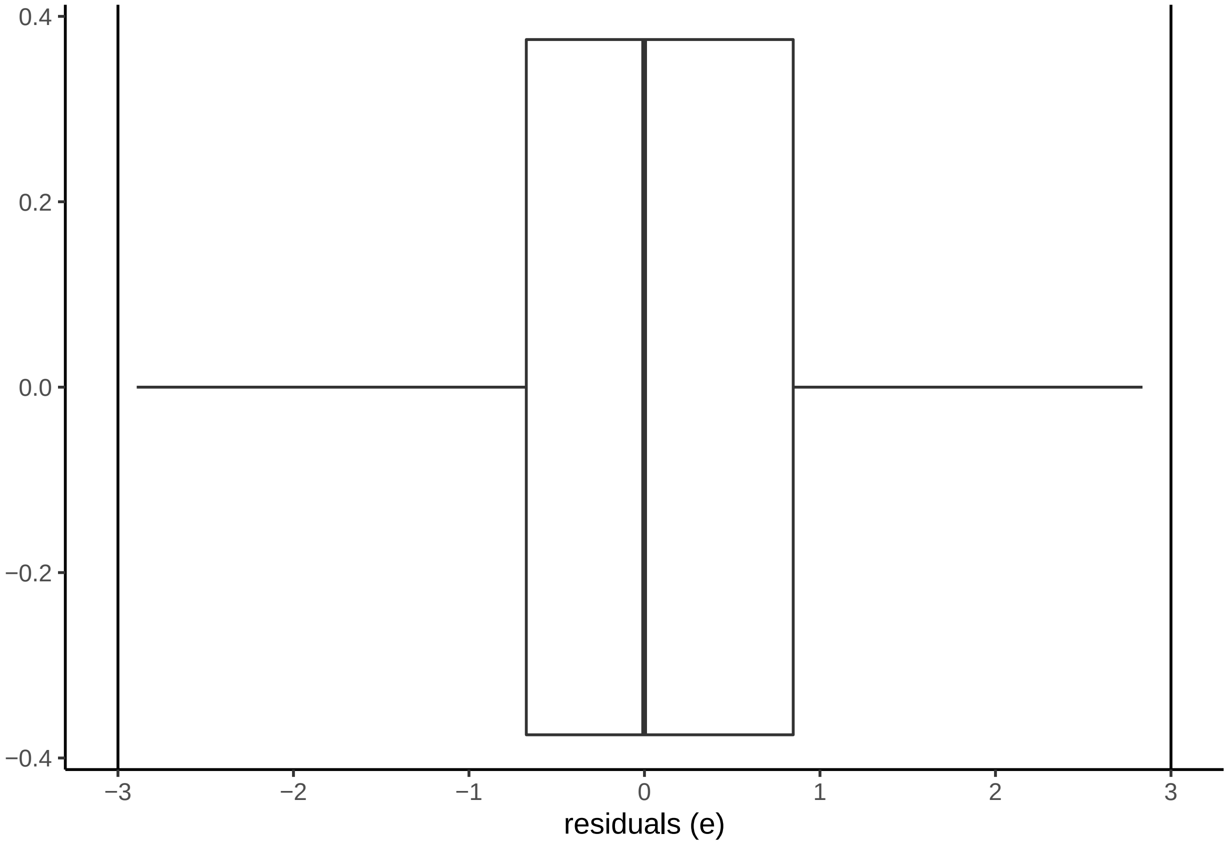
plot residuals

```
ggplot(data = mass_SVL_SLR_residuals,
       aes(x = e)) +
  geom_histogram() +
  theme_classic() +
  xlab("residuals (e)")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
ggplot(data = mass_SVL_SLR_residuals,
       aes(x = e)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(breaks = c(seq(-5, 5, 1))) +
  xlab("residuals (e)") +
  geom_vline(xintercept = -3, lty = 1) +
  geom_vline(xintercept = 3, lty = 1)
```



From the histogram, there are clearly points disconnected from the main curve. In the boxplot, we can distinguish that those outlying points have residuals <-3.5 and >3, so I'll go back and filter those out in the code chunk before making the distribution plots. Now the boxplot has no dots outside the main distribution.

And, check mean residual value:

```
mass_SVL_SLR_residuals %>%
  summarise(mean_res = mean(e),
            med = median(e))
```

```
##      mean_res      med
## 1 -0.02769117 -0.00174
```

Not zero, which is not ideal, but pretty close.

Next, check for high leverage points:

```
# compute values for observations
high_leverage <- data.frame(H = hatvalues(mass_SVL_SLR)
                           ) %>% mutate(row = rownames(.))

# compute cutoff value
h_bar <- (3*sum(high_leverage$H))/nrow(high_leverage)

# add to original dataframe
# see which observations have extremely high leverage (if any)
high_leverage_dat <- mass_SVL_SLR_residuals %>%
  mutate(row = rownames(.)) %>%
  left_join(., high_leverage, by = "row") %>%
  dplyr::filter(H > h_bar)
```

```r
high_leverage_dat
```

```
##          date        collect_time individual_ID SVL_mm mass_g sex_M_F
## 1  2021-04-05 2021-10-08 10:38:00             4     48    4.2       M
## 2  2021-04-05 2021-10-08 10:17:00             5     50    4.1       M
## 3  2021-04-05 2021-10-08 10:47:00             6     48    4.6       M
## 4  2021-04-05 2021-10-08 10:42:00             8     42    2.3       M
## 5  2021-04-05 2021-10-08 13:27:00             9     46    3.1       F
## 6  2021-04-26 2021-10-08 12:38:00            57     65   10.3       F
## 7  2021-04-26 2021-10-08 12:57:00            64     65   10.3       M
## 8  2021-04-26 2021-10-08 13:02:00            67     64   10.5       M
## 9  2021-05-03 2021-10-08 12:51:00            87     70   14.3       M
## 10 2021-05-17 2021-10-08 12:19:00           123     72   11.9       M
##    gravid_Y_N blood_sample_eye hematocrit_percent osmolality_mmol_kg
## 1        <NA>                R                 NA                341
## 2        <NA>                R                 NA                354
## 3        <NA>                L                 NA                355
## 4        <NA>                R                 NA                304
## 5           N                L                 NA                 NA
## 6           N                R                 30                388
## 7        <NA>                R                 30                378
## 8        <NA>                R                 32                402
## 9        <NA>                R                 35                394
## 10       <NA>                R                 52                362
##    cloacal_temp_C     processing_time hemolyzed   collect_date_time hold_time
## 1              26 2021-10-08 14:02:00         Y 2021-04-05 10:38:00       204
## 2              25 2021-10-08 13:59:00         Y 2021-04-05 10:17:00       222
## 3              24 2021-10-08 14:06:00         N 2021-04-05 10:47:00       199
## 4              23 2021-10-08 14:20:00         N 2021-04-05 10:42:00       218
## 5              23 2021-10-08 14:43:00      <NA> 2021-04-05 13:27:00        76
## 6              26 2021-10-08 13:27:00         N 2021-04-26 12:38:00        49
## 7              22 2021-10-08 15:20:00         Y 2021-04-26 12:57:00       143
## 8              24 2021-10-08 15:41:00         N 2021-04-26 13:02:00       159
## 9              24 2021-10-08 14:32:00         N 2021-05-03 12:51:00       101
## 10             27 2021-10-08 12:47:00      <NA> 2021-05-17 12:19:00        28
##       y_hat        e row          H
## 1   3.62910 -0.57090   4 0.05776372
## 2   4.44886  0.34886   5 0.04645120
## 3   3.62910 -0.97090   6 0.05776372
## 4   1.16982 -1.13018   8 0.10020003
## 5   2.80934 -0.29066   9 0.07049270
## 6  10.59706  0.29706  54 0.07049270
## 7  10.59706  0.29706  60 0.04132611
## 8  10.18718 -0.31282  63 0.05776372
## 9  12.64646 -1.65354  83 0.05193040
## 10 13.46622  1.56622 118 0.04645120
```

The points for individuals 4, 5, 6, 8, 9, 57, 64, 67, 87, and 123 seem to be high-leverage, so we will try removing them.

Check for influential points based on Cook's distance:

```r
# get Cook's distance
cooks <- data.frame(c = cooks.distance(mass_SVL_SLR) # specify model name
                    ) %>% mutate(row = rownames(.))
```

```r
# add to original dataframe
influential <- mass_SVL_SLR_residuals %>%
  mutate(row = rownames(.)) %>%
  left_join(., cooks, by = "row")

# see moderately influential points
cook_mod_inf <- influential %>%
  dplyr::filter(c>0.5)
cook_mod_inf
```

```
##  [1] date               collect_time       individual_ID      SVL_mm
##  [5] mass_g             sex_M_F            gravid_Y_N         blood_sample_eye
##  [9] hematocrit_percent osmolality_mmol_kg cloacal_temp_C     processing_time
## [13] hemolyzed          collect_date_time  hold_time          y_hat
## [17] e                  row                c
## <0 rows> (or 0-length row.names)
```

There are no even moderately-infuential points, at least based on Cook's distance, so there's nothing to potentially remove.

Create new SLR and check mean residual value after removing outliers and high leverage points:

```r
# create new dataframe with filtered data
cleaned_SMI_dat <- mass_SVL_SLR_residuals %>% # already had outliers removed
  mutate(row = rownames(.)) %>%
  # add high leverage point info
  left_join(., high_leverage, by = "row") %>%
  # remove high leverage points
  dplyr::filter(H < h_bar)

# model
mass_SVL_SLR2 <- lm(data = cleaned_SMI_dat, mass_g ~ SVL_mm)
summary(mass_SVL_SLR2)
```

```
##
## Call:
## lm(formula = mass_g ~ SVL_mm, data = cleaned_SMI_dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.72134 -0.88611  0.00146  0.70819  2.87193
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.32109    1.28410  -13.49   <2e-16 ***
## SVL_mm        0.42953    0.01953   21.99   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.209 on 133 degrees of freedom
## Multiple R-squared:  0.7843, Adjusted R-squared:  0.7827
## F-statistic: 483.6 on 1 and 133 DF,  p-value: < 2.2e-16
```

```r
# compute residuals
mass_SVL_SLR2_residuals <- cleaned_SMI_dat %>%
  mutate(y_hat = predict(mass_SVL_SLR2),
         e = residuals(mass_SVL_SLR2))

# check residuals values
mass_SVL_SLR2_residuals %>%
  summarise(mean_res = mean(e),
            med = median(e))
```

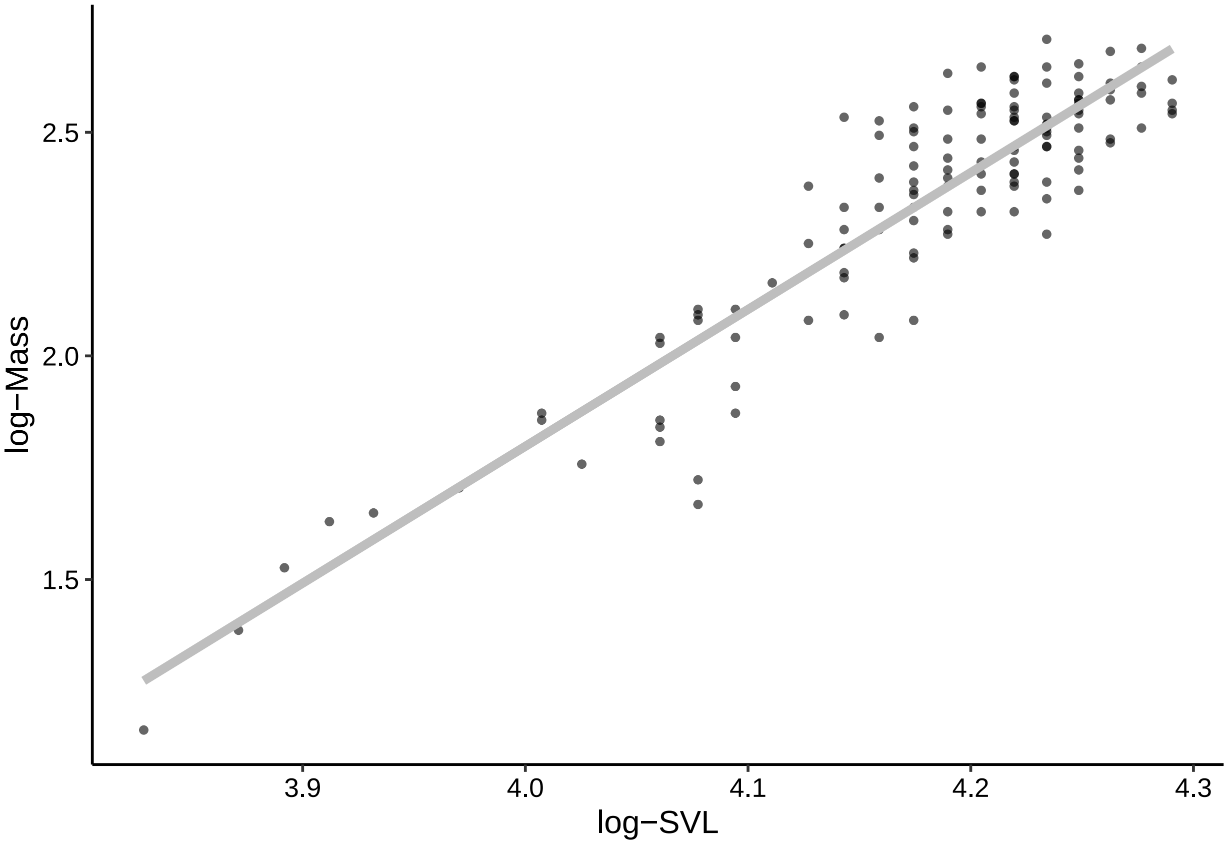
```
##       mean_res         med
## 1 4.654038e-17 0.001458016
```

The residuals are much much smaller, so I believe it is the right choice to remove the high leverage points. The “cleaned_SMI_dat” will be used to calculate the equation for SMI.

Step 2: make log-log relationship

plot and calculate SLR for filtered data

```
cleaned_SMI_dat %>%
  ggplot(data = .) +
  geom_point(aes(x = log(SVL_mm),
                 y = log(mass_g),
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = log(SVL_mm),
                  y = log(mass_g),
                  ),
              formula = y ~ x,
              method = "lm",
              color = "gray",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("log-SVL") +
  ylab("log-Mass") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0
)
```



SLR

```
log_mass_SVL_SLR <- lm(data = cleaned_SMI_dat,
                       log(mass_g) ~ log(SVL_mm))
summary(log_mass_SVL_SLR)
```

```
##
## Call:
## lm(formula = log(mass_g) ~ log(SVL_mm), data = cleaned_SMI_dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.36738 -0.07518  0.00334  0.06423  0.29781
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -10.4465     0.4858  -21.50   <2e-16 ***
## log(SVL_mm)   3.0611     0.1162   26.34   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1175 on 133 degrees of freedom
## Multiple R-squared:  0.8391, Adjusted R-squared:  0.8379
## F-statistic: 693.6 on 1 and 133 DF,  p-value: < 2.2e-16
```

compute standardized major axis using this regression equation:

```
r <- sqrt(0.8391) # Pearson's correlection coefficient (sqrt of R-squared)
b_OLS <- 3.0611 # regression slope
```

```
b_SMA <- b_OLS/r
```

also get a value for L0:

```
L0 <- mean(cleaned_SMI_dat$SVL_mm)
```

Step 3: calculate scaled mass index

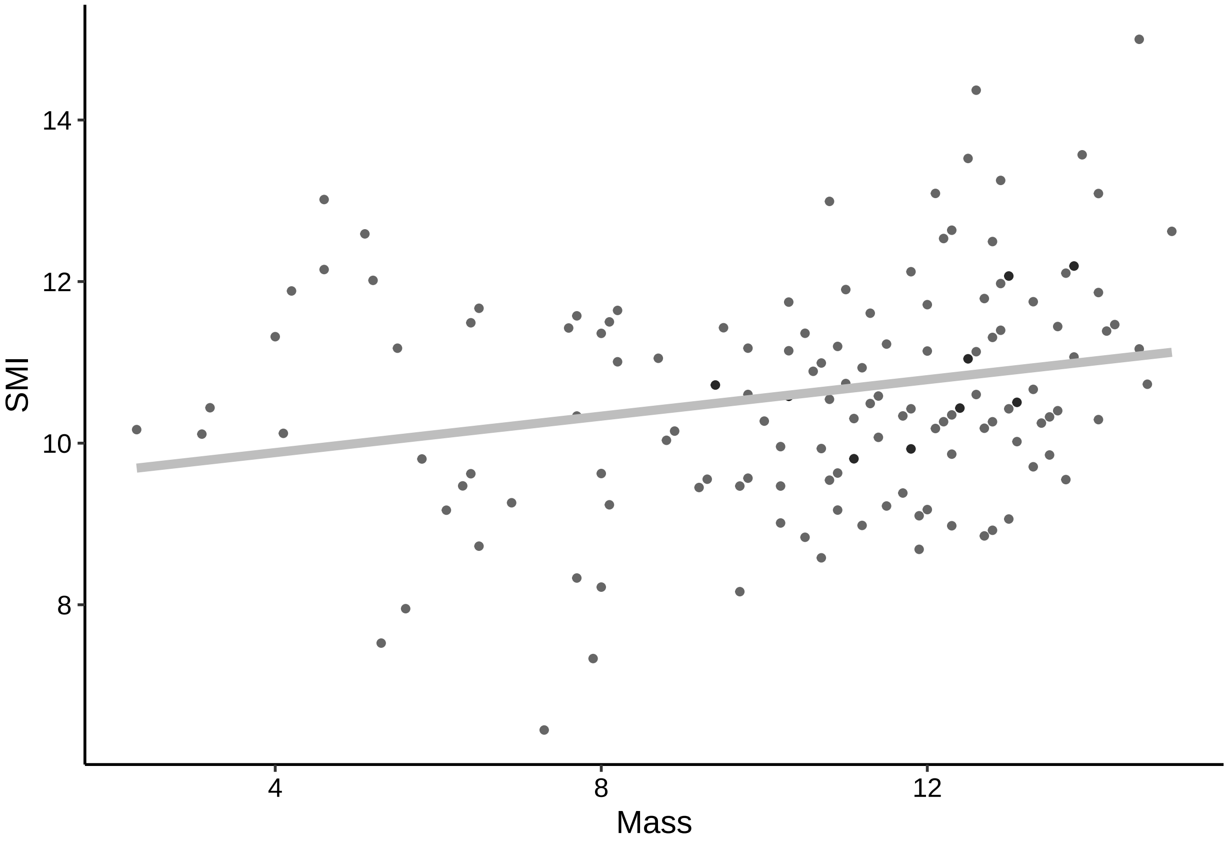
(And join weather data.)

```
morpho_blood_SMI <- morpho_blood_dat %>%
  mutate(SMI = mass_g * ((L0/SVL_mm) ^ b_SMA)) %>%
  left_join(all_times_weather,
            by = c("collect_date_time")
            )
```

Check

Look at the difference between regular mass and SMI:

```
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_point(aes(x = mass_g,
                 y = SMI,
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = mass_g,
                  y = SMI,
                  ),
              formula = y ~ x,
              method = "lm",
              color = "gray",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Mass") +
  ylab("SMI") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0
)
```



Join Data

Add CEWL and morpho_blood_SMI data together.

```
CEWL_data_full <- CEWL %>%
  left_join(morpho_blood_SMI,
            by = c("date", "individual_ID")
            )
summary(CEWL_data_full)
```

```
##       date                Time           individual_ID  region
##  Min.   :2021-04-05   Length:699         01     :  5   dewl:139
##  1st Qu.:2021-04-19   Class :character   02     :  5   dors:141
##  Median :2021-04-26   Mode  :character   03     :  5   head:141
##  Mean   :2021-04-28                      04     :  5   mite:137
##  3rd Qu.:2021-05-10                      05     :  5   vent:141
##  Max.   :2021-05-17                      06     :  5
##                                          (Other):669
##    TEWL_g_m2h    ambient_temp_C  ambient_RH_percent
##  Min.   : 3.41   Min.   :22.30   Min.   :34.00
##  1st Qu.:17.09   1st Qu.:23.00   1st Qu.:41.30
##  Median :22.00   Median :23.20   Median :45.20
##  Mean   :25.87   Mean   :23.44   Mean   :43.56
##  3rd Qu.:32.59   3rd Qu.:23.80   3rd Qu.:46.30
##  Max.   :96.16   Max.   :25.30   Max.   :53.10
##
##  CEWL_date_time                 ambient_temp_K     e_s_kPa         e_a_kPa
##  Min.   :2021-04-05 13:24:15   Min.   :295.4   Min.   :2.760   Min.   :0.9779
##  1st Qu.:2021-04-19 14:07:34   1st Qu.:296.1   1st Qu.:2.882   1st Qu.:1.2086
##  Median :2021-04-26 17:10:23   Median :296.4   Median :2.918   Median :1.3315
##  Mean   :2021-04-28 23:39:45   Mean   :296.6   Mean   :2.964   Mean   :1.2910
##  3rd Qu.:2021-05-10 16:03:10   3rd Qu.:296.9   3rd Qu.:3.028   3rd Qu.:1.3948
##  Max.   :2021-05-17 17:22:31   Max.   :298.4   Max.   :3.318   Max.   :1.4956
##
##     VPD_kPa       collect_time                      SVL_mm          mass_g
##  Min.   :1.297   Min.   :2021-10-08 11:29:00   Min.   :46.00   Min.   : 3.20
##  1st Qu.:1.541   1st Qu.:2021-10-08 12:37:00   1st Qu.:64.00   1st Qu.: 9.70
##  Median :1.683   Median :2021-10-08 12:48:00   Median :67.00   Median :11.40
##  Mean   :1.673   Mean   :2021-10-08 12:55:48   Mean   :65.81   Mean   :10.88
##  3rd Qu.:1.779   3rd Qu.:2021-10-08 13:02:15   3rd Qu.:69.00   3rd Qu.:12.80
##  Max.   :2.055   Max.   :2021-10-08 15:44:00   Max.   :73.00   Max.   :15.00
##                  NA's   :59                    NA's   :44      NA's   :44
##  sex_M_F    gravid_Y_N blood_sample_eye hematocrit_percent osmolality_mmol_kg
##  F   :216   N   : 91   both: 10         Min.   :16.0       Min.   :293.0
##  M   :439   Y   :125   L   :  0         1st Qu.:33.0       1st Qu.:347.0
##  NA's: 44   NA's:483   R   :645         Median :36.0       Median :368.0
##                        NA's: 44         Mean   :35.4       Mean   :366.8
##                                         3rd Qu.:38.0       3rd Qu.:387.0
##                                         Max.   :54.0       Max.   :436.0
##                                         NA's   :119        NA's   :49
##  cloacal_temp_C processing_time                hemolyzed
##  Min.   :20.0   Min.   :2021-10-08 12:44:00   N   :368
##  1st Qu.:22.0   1st Qu.:2021-10-08 14:14:00   Y   :179
##  Median :23.0   Median :2021-10-08 15:25:00   NA's:152
##  Mean   :23.4   Mean   :2021-10-08 15:16:48
##  3rd Qu.:25.0   3rd Qu.:2021-10-08 16:18:00
##  Max.   :28.0   Max.   :2021-10-08 17:38:00
##  NA's   :49     NA's   :59
##  collect_date_time               hold_time          SMI          temp_C_interpol
##  Min.   :2021-04-05 13:25:00   Min.   : 21.0   Min.   : 6.450   Min.   :15.67
##  1st Qu.:2021-04-19 13:09:30   1st Qu.: 91.0   1st Qu.: 9.624   1st Qu.:16.66
##  Median :2021-05-03 12:40:00   Median :132.0   Median :10.505   Median :18.68
##  Mean   :2021-05-01 01:04:48   Mean   :140.5   Mean   :10.573   Mean   :18.77
##  3rd Qu.:2021-05-10 12:52:00   3rd Qu.:189.0   3rd Qu.:11.444   3rd Qu.:19.96
##  Max.   :2021-05-17 13:01:00   Max.   :268.0   Max.   :14.999   Max.   :23.61
##  NA's   :59                    NA's   :69      NA's   :44       NA's   :59
##  RH_percent_interpol  VPD_kPa_int     Wind_mph_interpol Solar_rad_Wm2_interpol
##  Min.   :44.29       Min.   :0.3424   Min.   :3.773     Min.   : 587.0
##  1st Qu.:57.51       1st Qu.:0.5533   1st Qu.:4.577     1st Qu.: 741.9
##  Median :68.10       Median :0.6986   Median :5.000     Median : 951.3
##  Mean   :66.53       Mean   :0.7837   Mean   :4.945     Mean   : 892.3
##  3rd Qu.:72.54       3rd Qu.:1.0127   3rd Qu.:5.233     3rd Qu.:1032.9
##  Max.   :81.10       Max.   :1.5691   Max.   :6.200     Max.   :1037.5
##  NA's   :59          NA's   :59       NA's   :59        NA's   :59
```

Final Formatting

redo the levels for body region:

```
CEWL_data_full$region <- factor(CEWL_data_full$region,
                                levels = c("dors", "vent",
                                           "head", "dewl", "mite"),
                                labels = c("Dorsum", "Ventrum", "Head",
                                           "Dewlap", "Mite Patch")
                                )
unique(CEWL_data_full$region)
```

```
## [1] Dorsum     Ventrum    Dewlap     Head       Mite Patch
## Levels: Dorsum Ventrum Head Dewlap Mite Patch
```

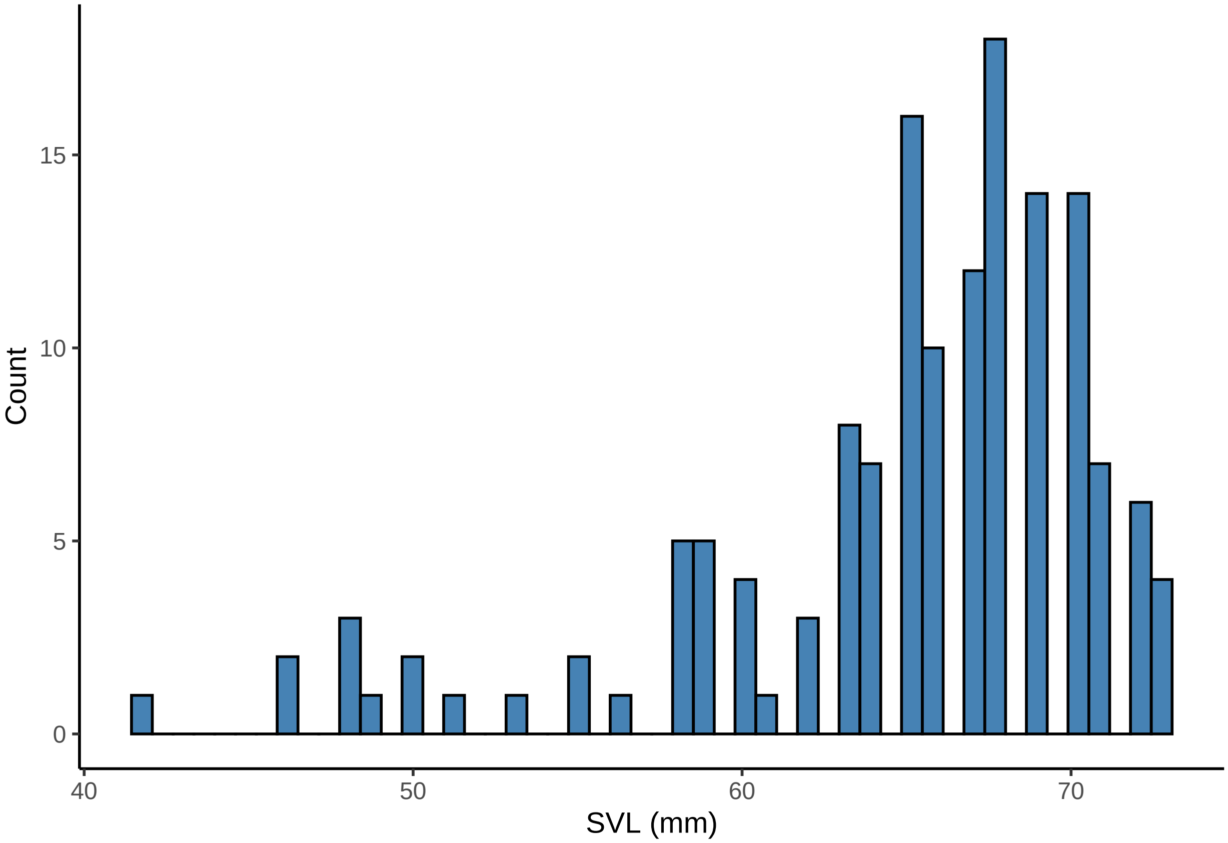
NOTE: running this more than once overrides things, so be careful

Check Data Distributions

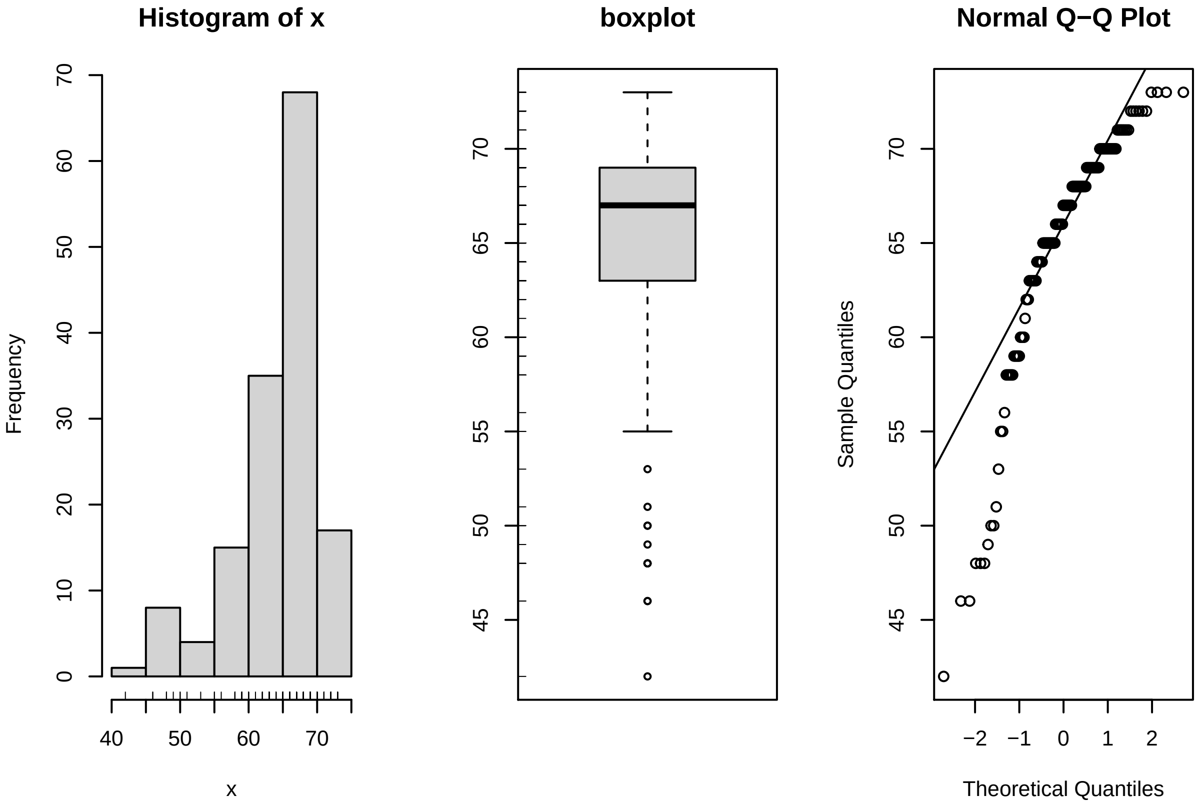
Histograms & Q-Q Plots

SVL

```
morpho_blood_SMI %>%
  ggplot(., aes(x = SVL_mm)) +
  geom_histogram(color = "black", fill="steelblue", bins=50) +
  theme_classic() +
  xlab("SVL (mm)") +
  ylab("Count")
```



```
simple.eda(morpho_blood_SMI$SVL_mm)
```

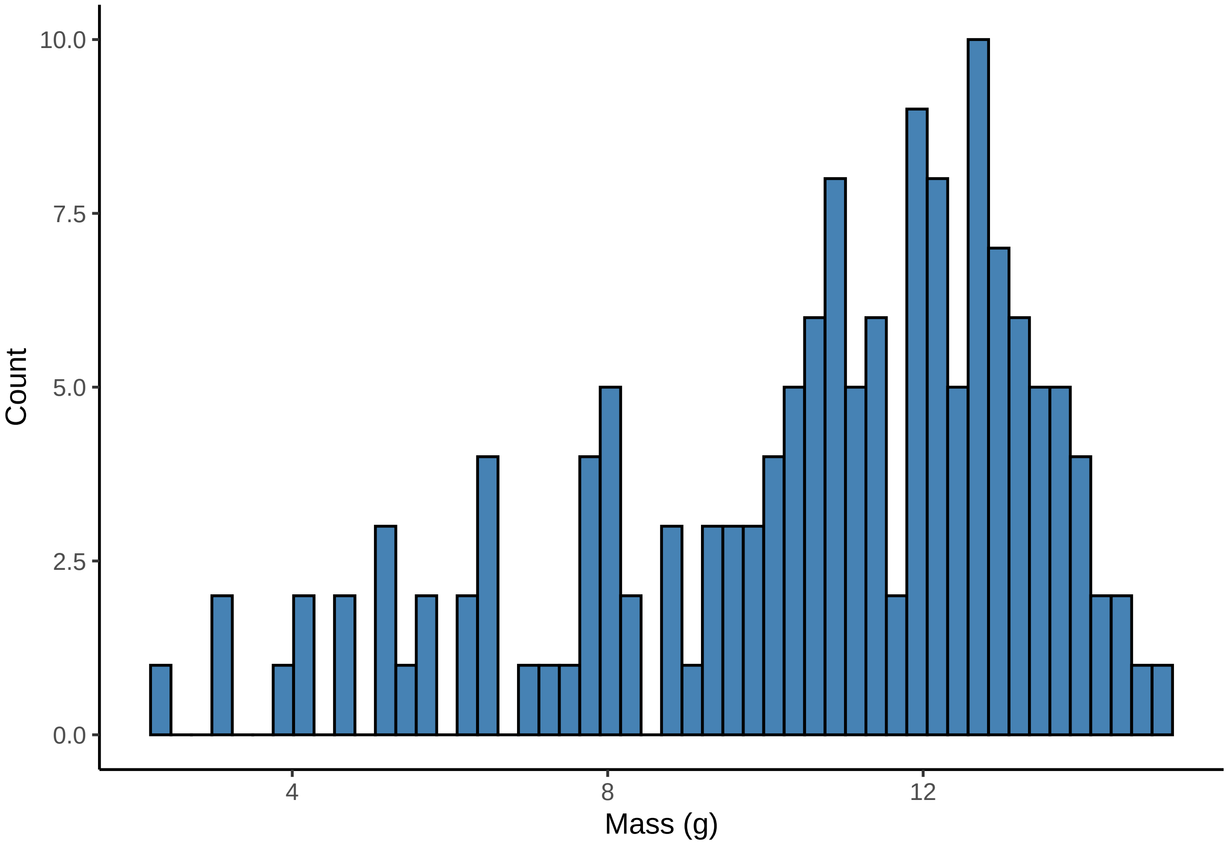


```
# Normality test if p > .05, data is normal. Data is not normal.
shapiro.test(morpho_blood_SMI$SVL_mm)
```

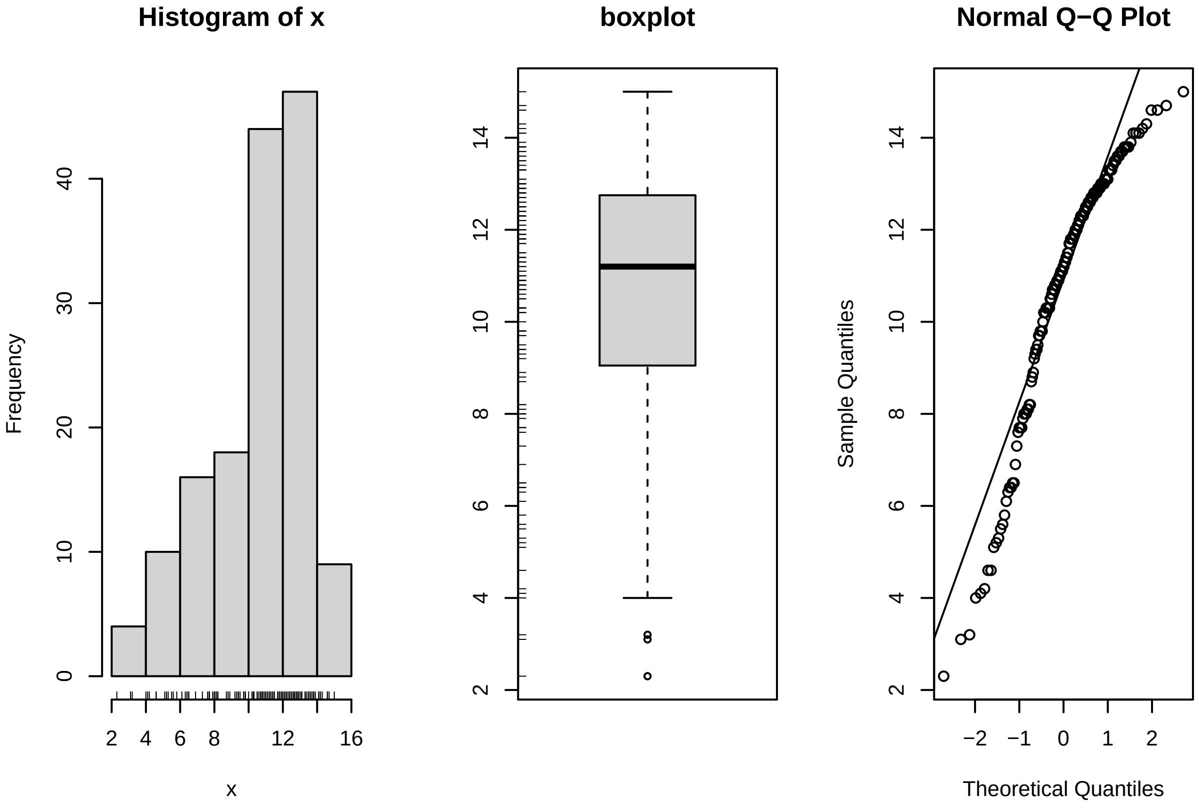
```
##
##  Shapiro-Wilk normality test
##
## data:  morpho_blood_SMI$SVL_mm
## W = 0.85469, p-value = 8.85e-11
```

Mass

```
morpho_blood_SMI %>%
  ggplot(., aes(x = mass_g)) +
  geom_histogram(color = "black", fill="steelblue", bins=50) +
  theme_classic() +
  xlab("Mass (g)") +
  ylab("Count")
```



```r
simple.eda(morpho_blood_SMI$mass_g)
```

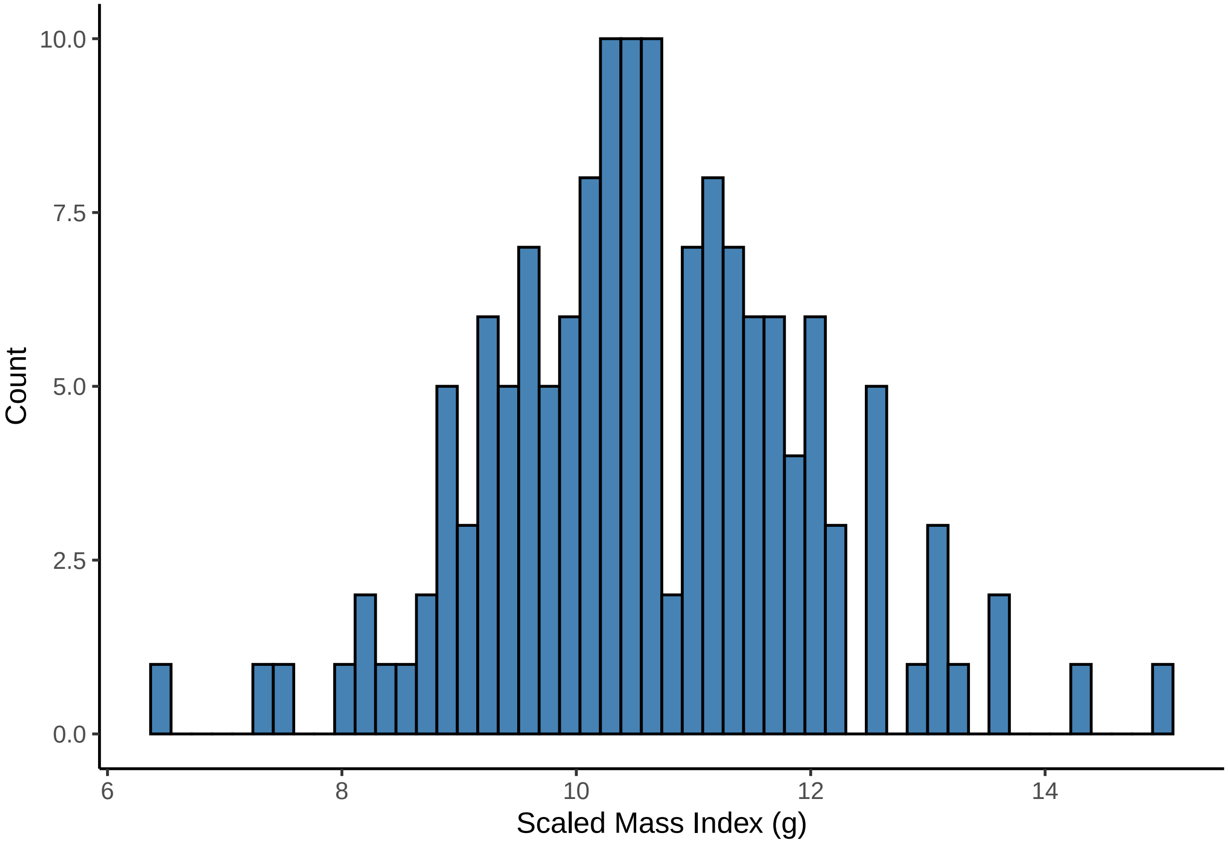


```
# Normality test if p > .05, data is normal. Data is not normal.
shapiro.test(morpho_blood_SMI$mass_g)
```

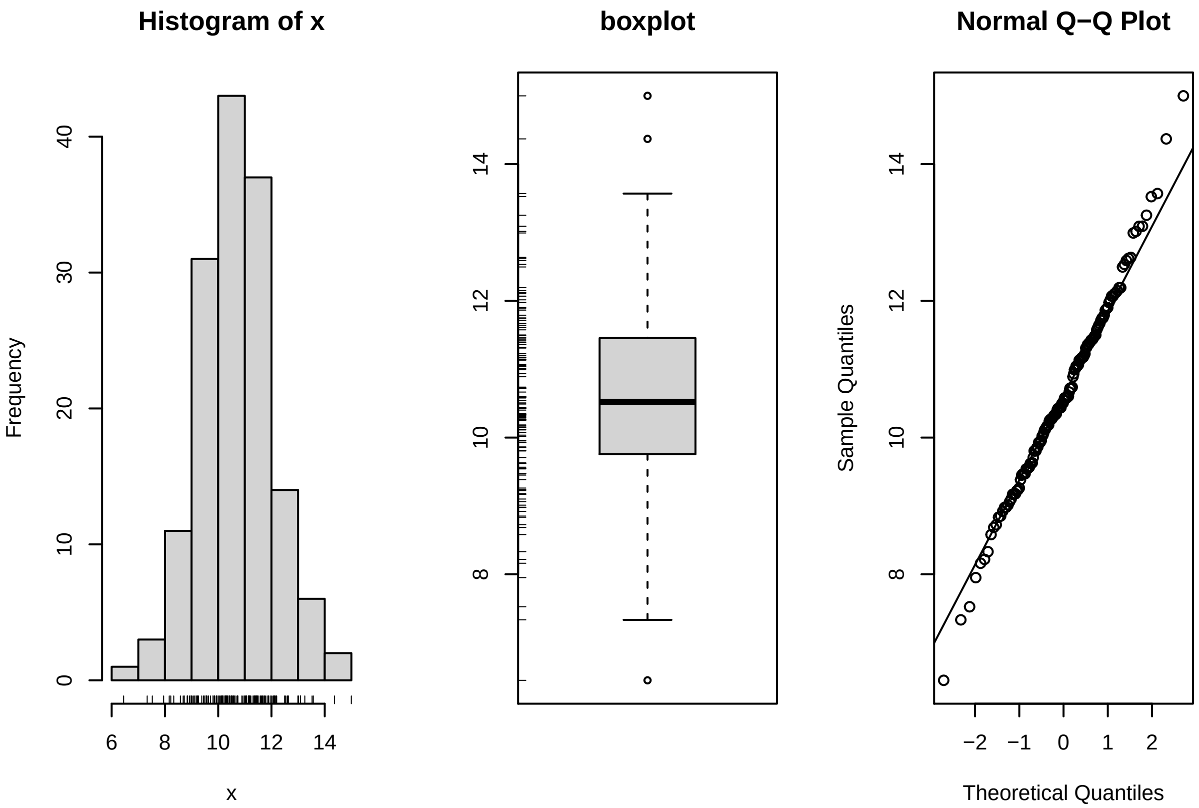
```
##
##  Shapiro-Wilk normality test
##
## data:  morpho_blood_SMI$mass_g
## W = 0.92517, p-value = 5.397e-07
```

SMI

```
morpho_blood_SMI %>%
  ggplot(., aes(x = SMI)) +
  geom_histogram(color = "black", fill="steelblue", bins=50) +
  theme_classic() +
  xlab("Scaled Mass Index (g)") +
  ylab("Count")
```



```
simple.eda(morpho_blood_SMI$SMI)
```



```
# Normality test if p > .05, data is normal. Data is not normal.
shapiro.test(morpho_blood_SMI$SMI)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  morpho_blood_SMI$SMI
## W = 0.9937, p-value = 0.7667
```

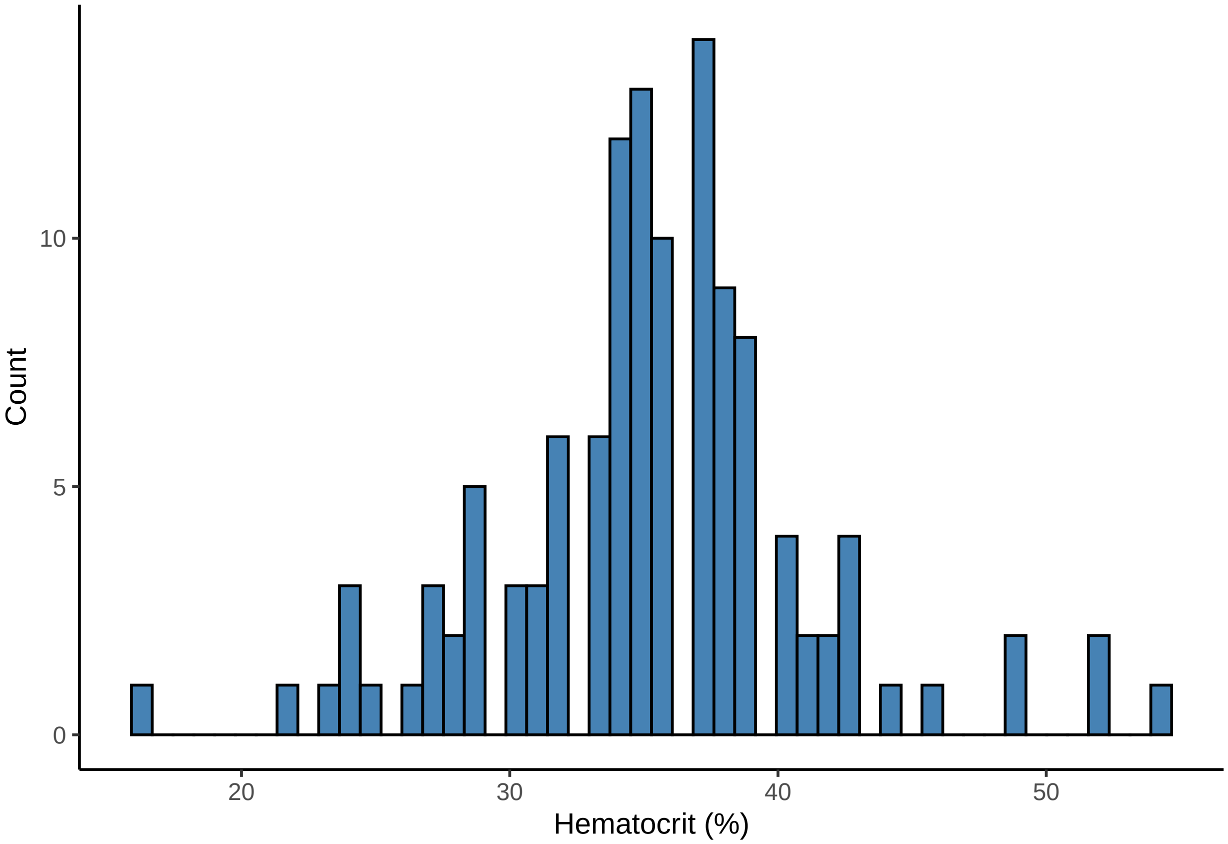
SMI is one of the few variables actually normally distributed!

Hematocrit

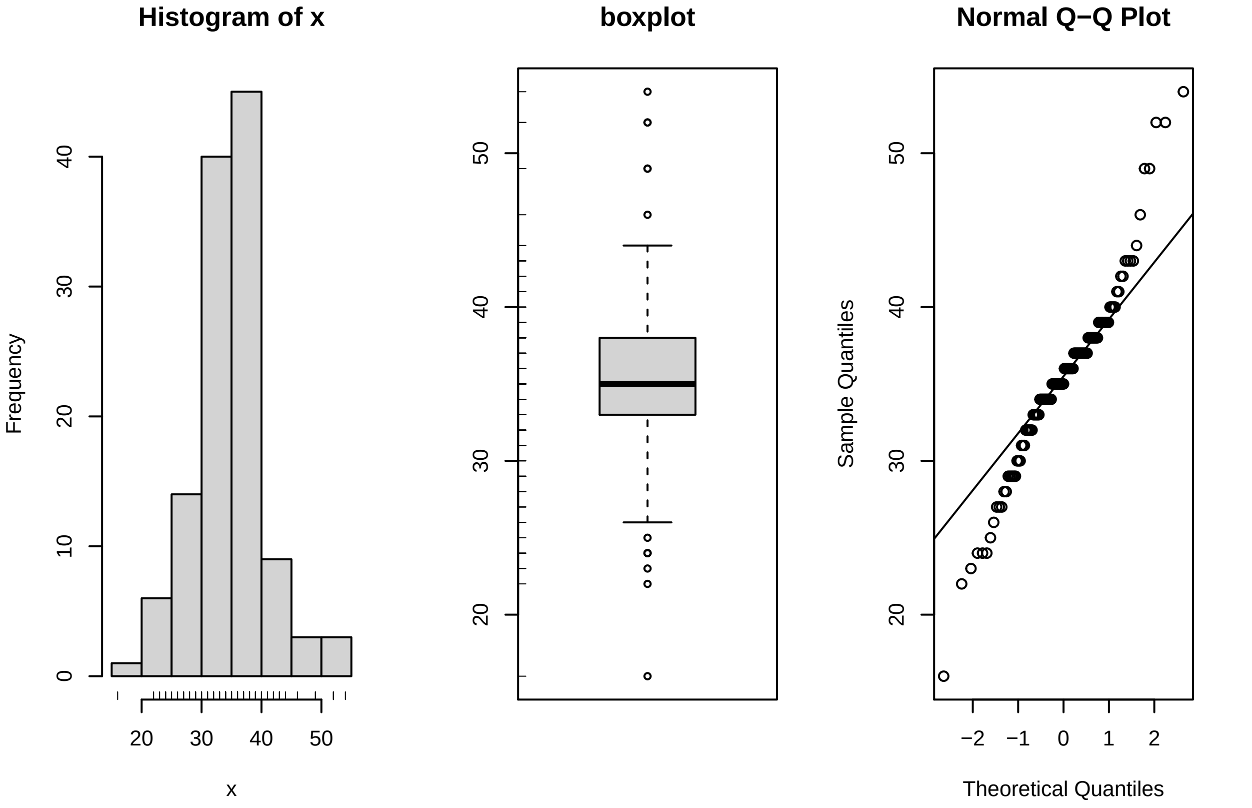
looks pretty normally distributed around ~35%, but not statistically normal

```
morpho_blood_SMI %>%
  ggplot(., aes(x = hematocrit_percent)) +
  geom_histogram(color = "black", fill="steelblue", bins=50) +
  theme_classic() +
  xlab("Hematocrit (%)") +
  ylab("Count")
```

```
## Warning: Removed 27 rows containing non-finite values (stat_bin).
```



```
simple.eda(morpho_blood_SMI$hematocrit_percent)
```



```
# Normality test if p > .05, data is normal. Data is not normal
shapiro.test(morpho_blood_SMI$hematocrit_percent)
```

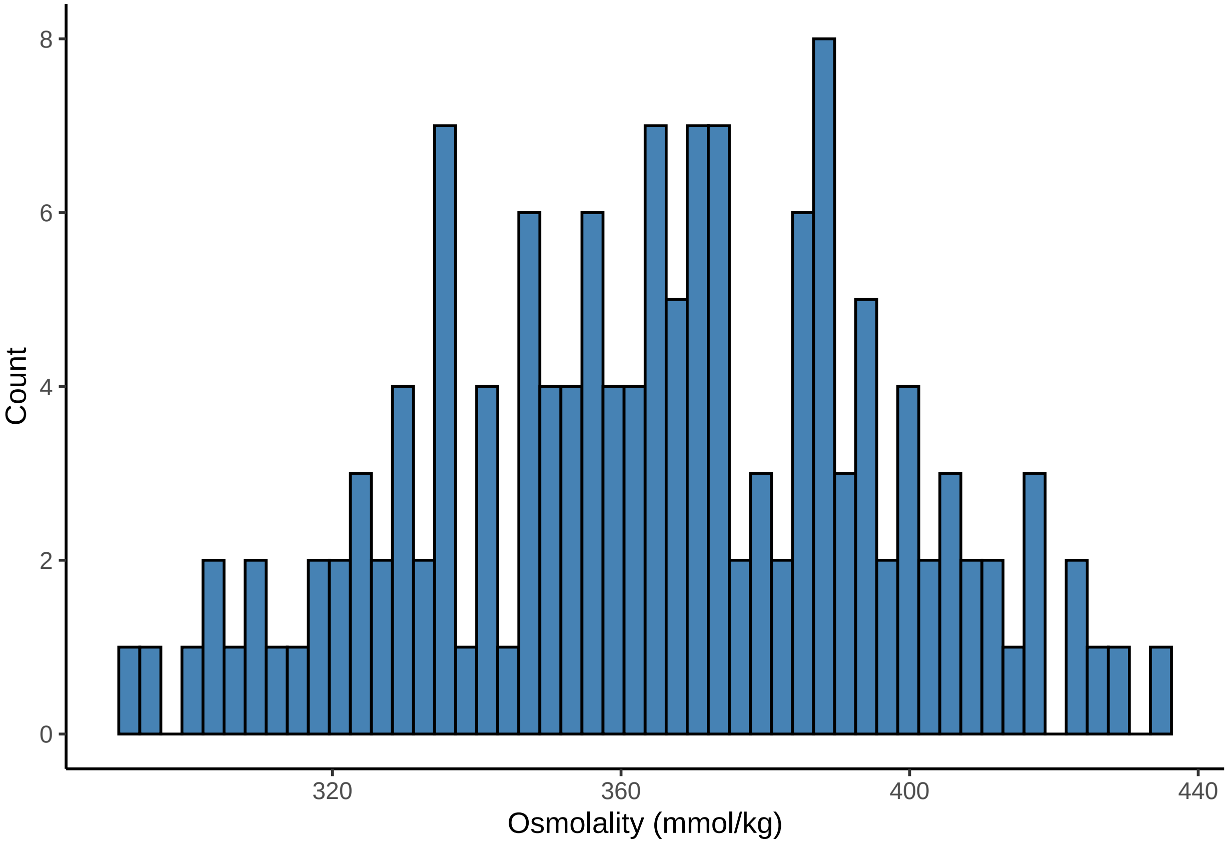
```
##
##  Shapiro-Wilk normality test
##
## data:  morpho_blood_SMI$hematocrit_percent
## W = 0.95933, p-value = 0.001056
```

Osmolality

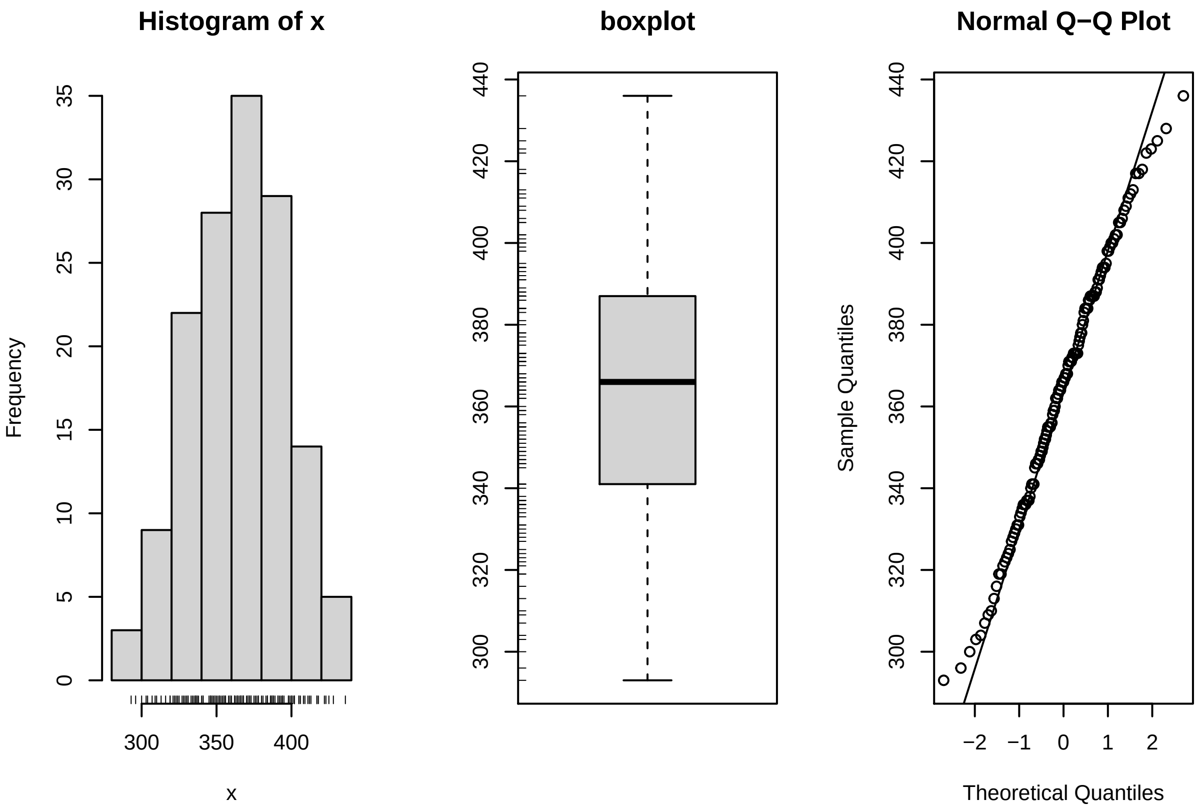
pretty normally distributed around ~370! :D

```
morpho_blood_SMI %>%
  ggplot(., aes(x = osmolality_mmol_kg)) +
  geom_histogram(color = "black", fill="steelblue", bins=50) +
  theme_classic() +
  xlab("Osmolality (mmol/kg)") +
  ylab("Count")
```

```
## Warning: Removed 3 rows containing non-finite values (stat_bin).
```



```
simple.eda(morpho_blood_SMI$osmolality_mmol_kg)
```



```
# Normality test if p > .05, data is normal. Data is normal
shapiro.test(morpho_blood_SMI$osmolality_mmol_kg)
```

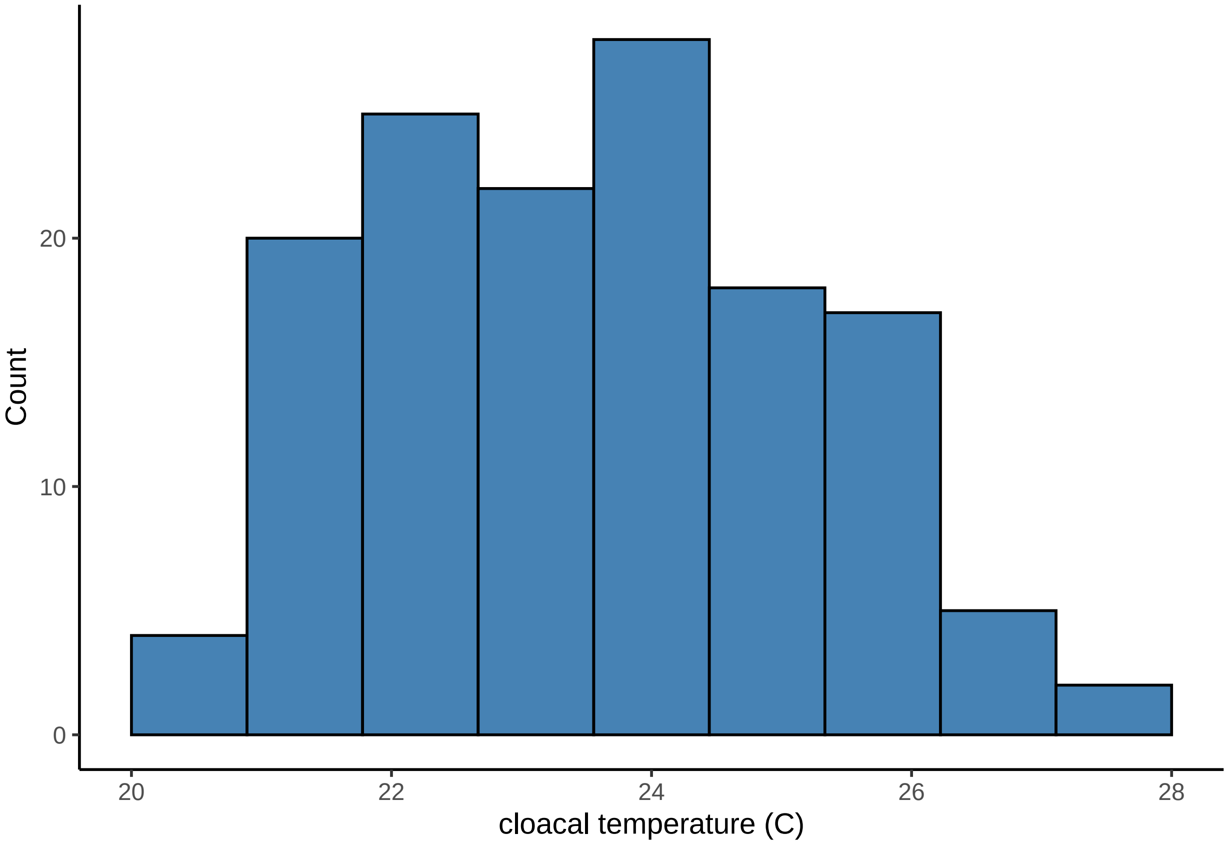
```
##
##  Shapiro-Wilk normality test
##
## data:  morpho_blood_SMI$osmolality_mmol_kg
## W = 0.99198, p-value = 0.5876
```

Cloacal Temperature

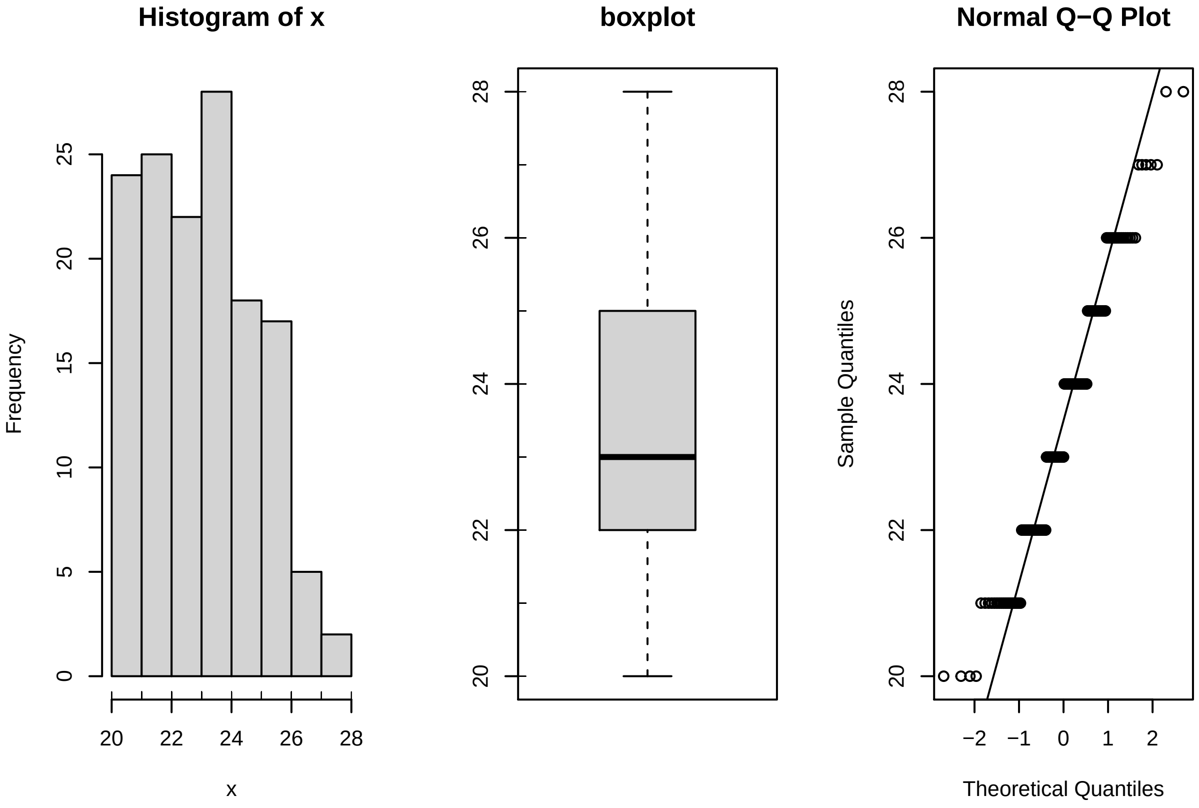
seems normally distributed, but not normal

```
morpho_blood_SMI %>%
  ggplot(., aes(x = cloacal_temp_C)) +
  geom_histogram(color = "black", fill="steelblue", bins=10) +
  theme_classic() +
  xlab("cloacal temperature (C)") +
  ylab("Count")
```

```
## Warning: Removed 7 rows containing non-finite values (stat_bin).
```



```r
simple.eda(morpho_blood_SMI$cloacal_temp_C)
```

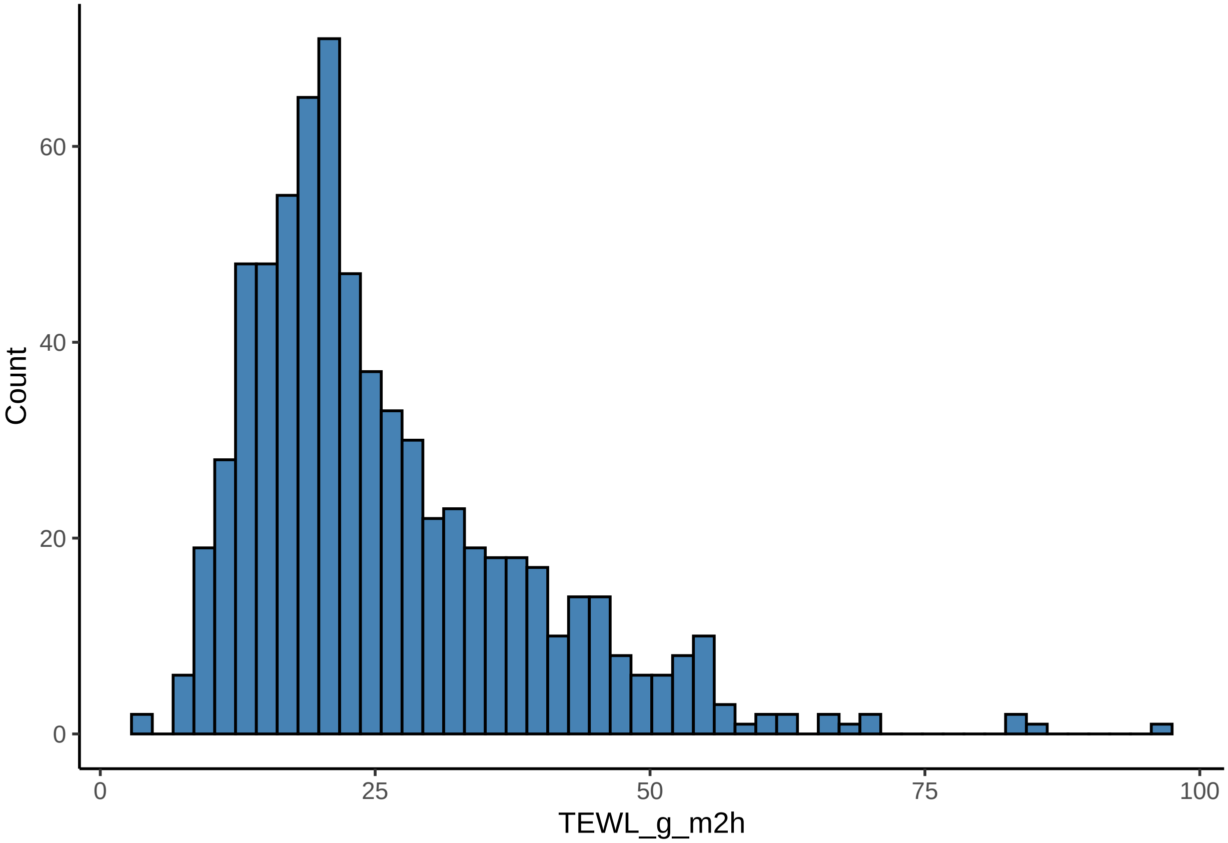


```
# Normality test if p > .05, data is normal. Data is not normal
shapiro.test(morpho_blood_SMI$cloacal_temp_C)
```

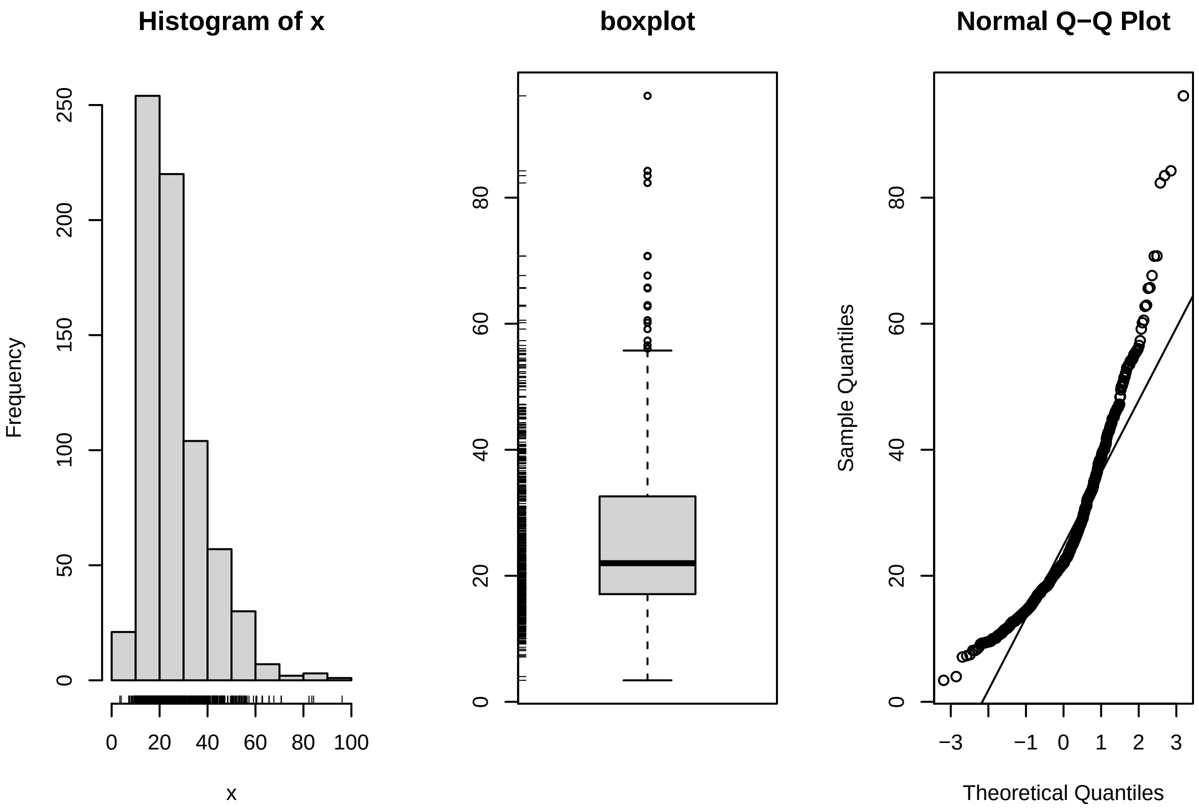
```
##
##  Shapiro-Wilk normality test
##
## data:  morpho_blood_SMI$cloacal_temp_C
## W = 0.95661, p-value = 0.0002007
```

CEWL

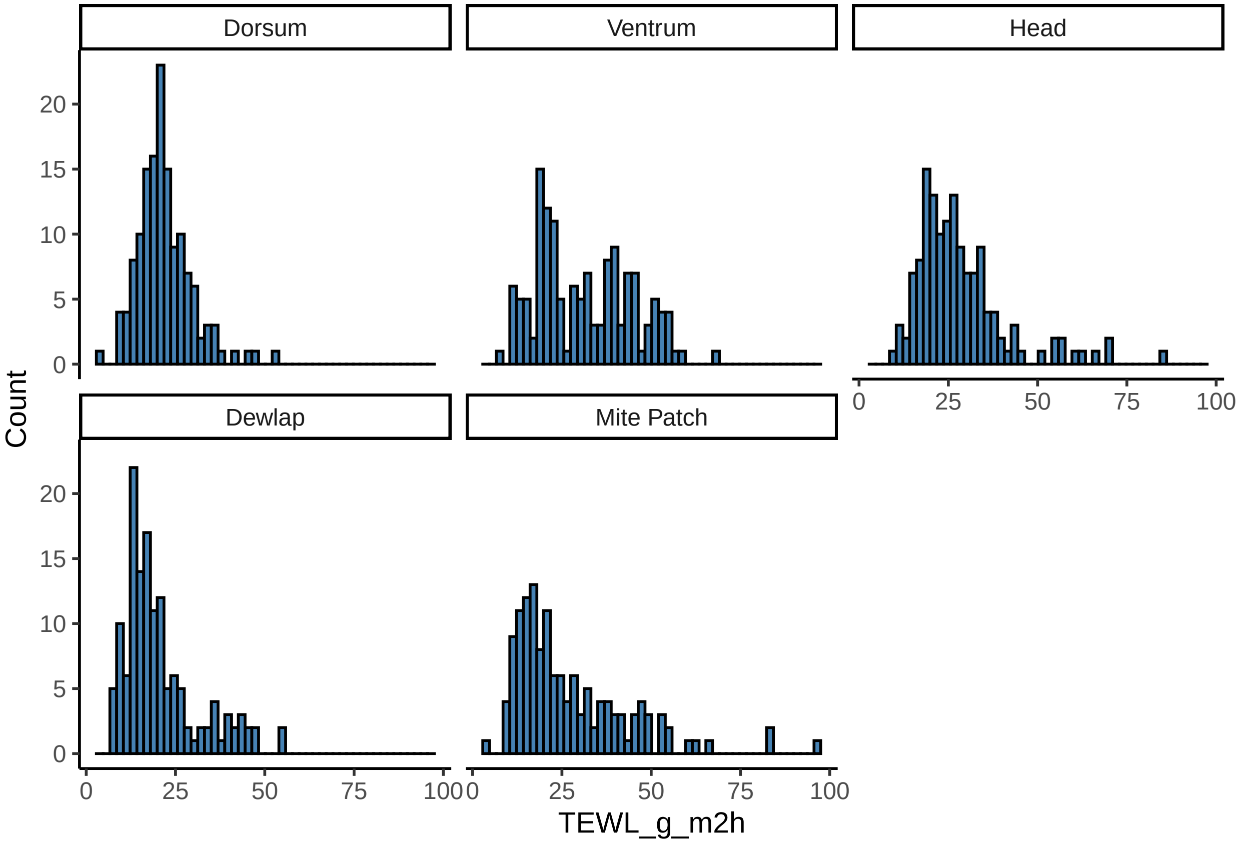
```
CEWL_data_full %>%
  ggplot(., aes(x = TEWL_g_m2h)) +
  geom_histogram(color = "black", fill="steelblue", bins=50) +
  theme_classic() +
  xlab("TEWL_g_m2h") +
  ylab("Count")
```



```
simple.eda(CEWL_data_full$TEWL_g_m2h)
```



```
CEWL_data_full %>%
  ggplot(., aes(x = TEWL_g_m2h)) +
  geom_histogram(color = "black", fill="steelblue", bins=50) +
  theme_classic() +
  xlab("TEWL_g_m2h") +
  ylab("Count") +
  facet_wrap(~region)
```



```
# Normality test if p > .05, data is normal. Data is not normal
shapiro.test(CEWL_data_full$TEWL_g_m2h)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  CEWL_data_full$TEWL_g_m2h
## W = 0.89548, p-value < 2.2e-16
```

```
# Log transformation
shapiro.test(log(CEWL_data_full$TEWL_g_m2h))
```

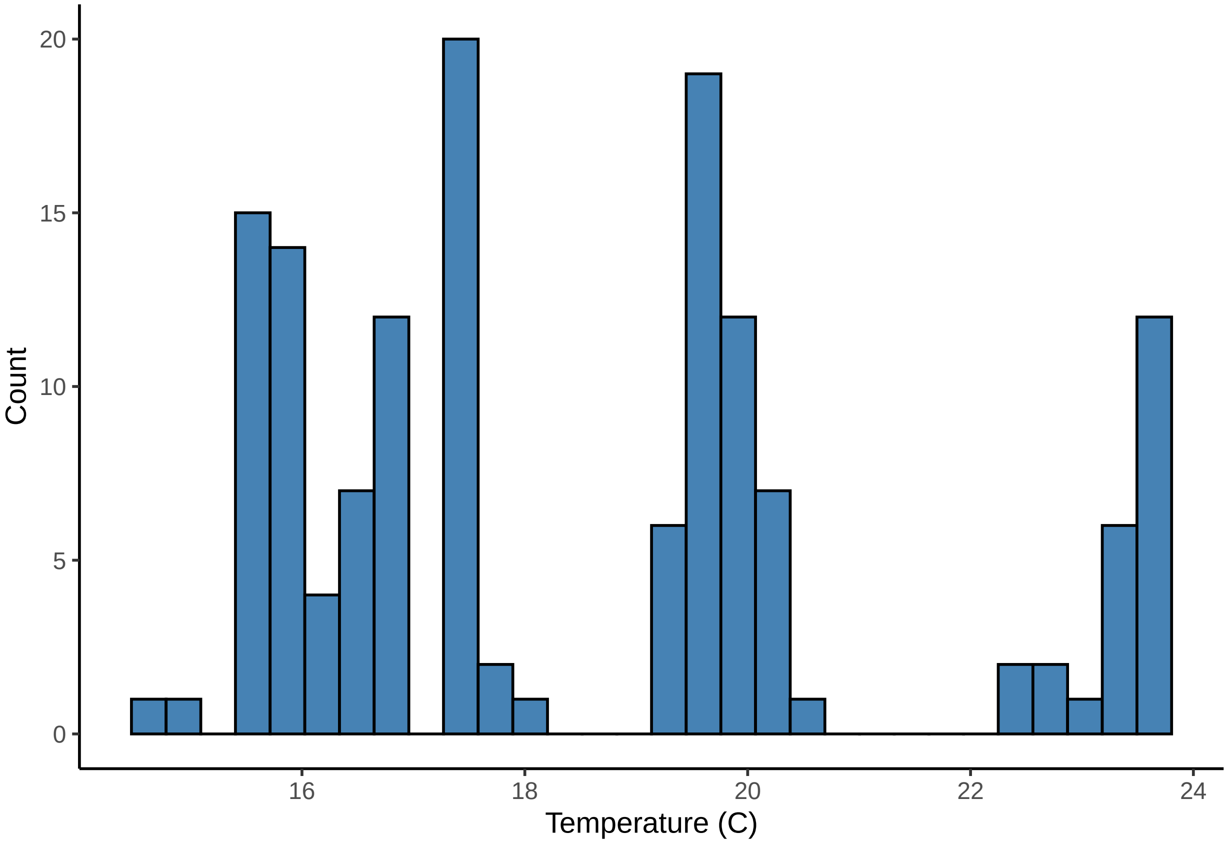
```
##
##  Shapiro-Wilk normality test
##
## data:  log(CEWL_data_full$TEWL_g_m2h)
## W = 0.99368, p-value = 0.004976
```

```
# p-value improves to  0.00548, but is still significant
```

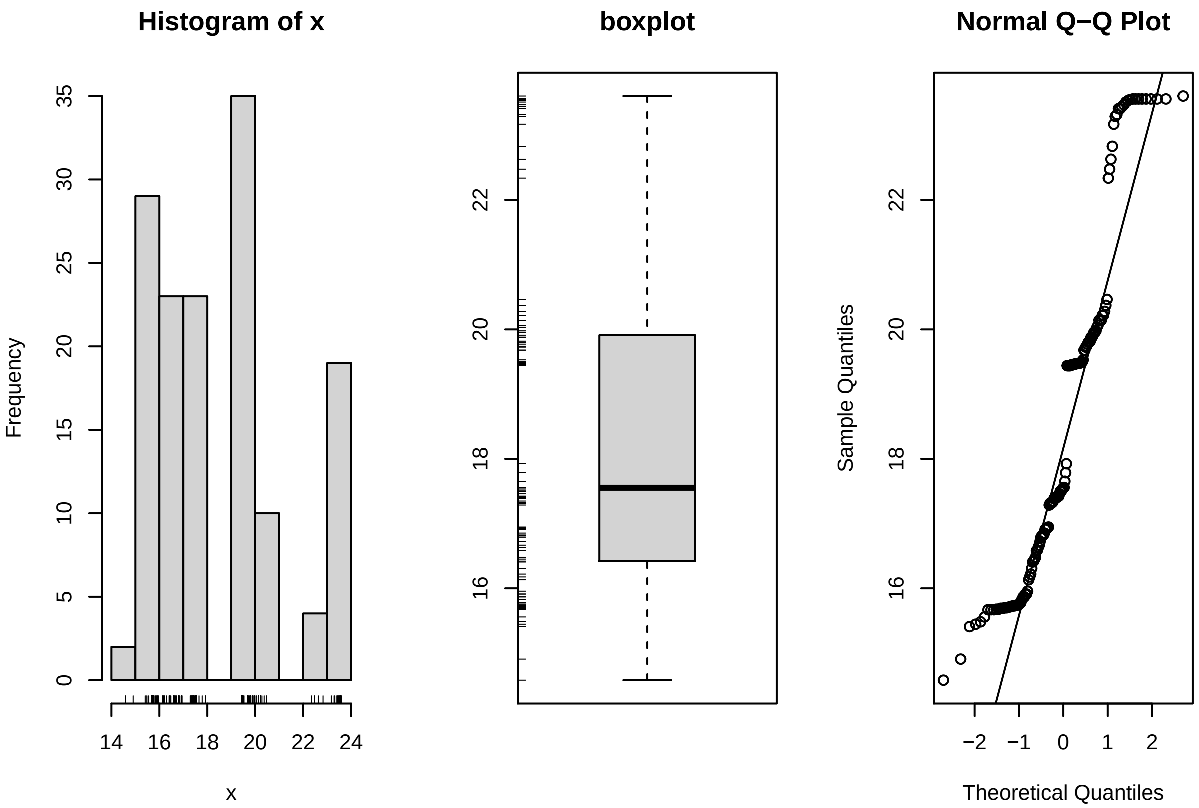
Capture Temperature

```
morpho_blood_SMI %>%
  ggplot(., aes(x = temp_C_interpol)) +
  geom_histogram(color = "black", fill="steelblue", bins=30) +
  theme_classic() +
  xlab("Temperature (C)") +
  ylab("Count")
```

```
## Warning: Removed 3 rows containing non-finite values (stat_bin).
```



```
simple.eda(morpho_blood_SMI$temp_C_interpol)
```



```
# Normality test if p > .05, data is normal. Data is not normal
shapiro.test(morpho_blood_SMI$temp_C_interpol)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  morpho_blood_SMI$temp_C_interpol
## W = 0.89272, p-value = 8.162e-09
```

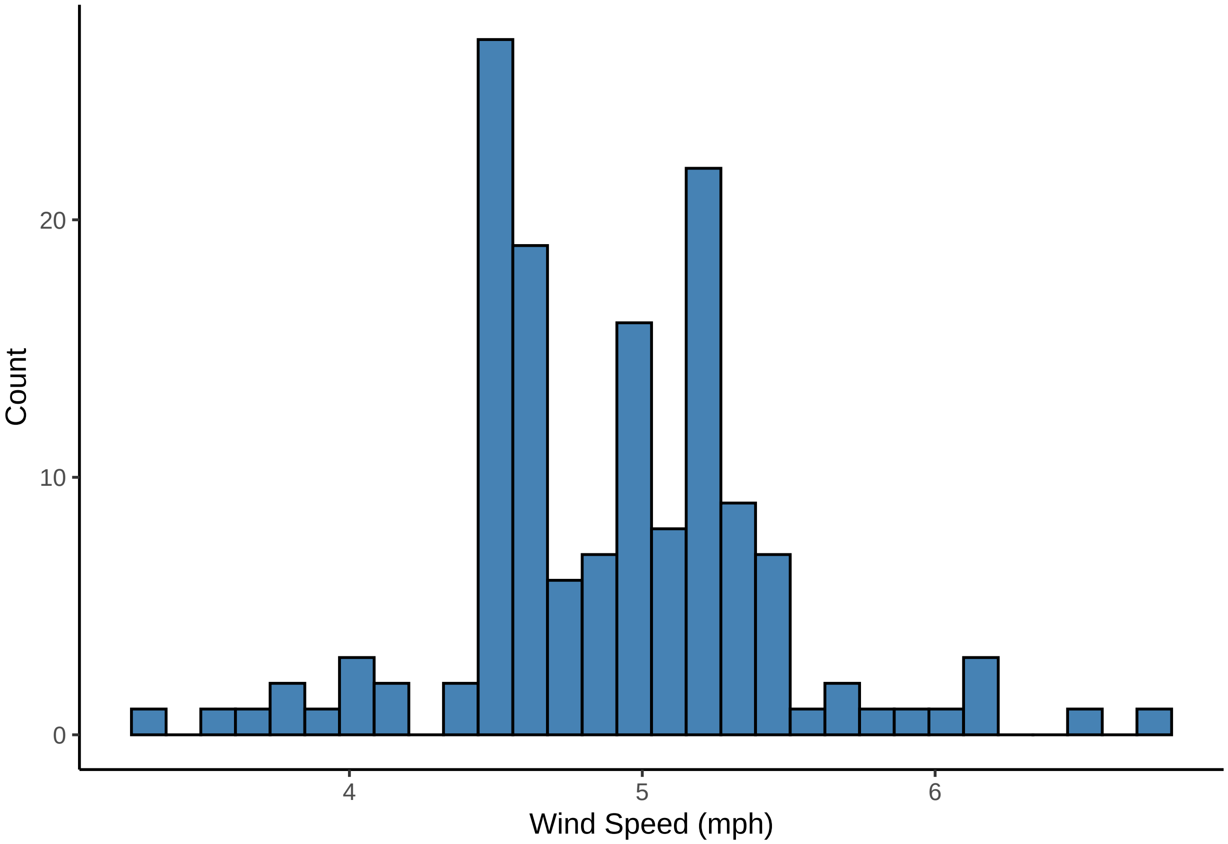
```
# log transform
shapiro.test(log(morpho_blood_SMI$temp_C_interpol))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  log(morpho_blood_SMI$temp_C_interpol)
## W = 0.91155, p-value = 9.439e-08
```

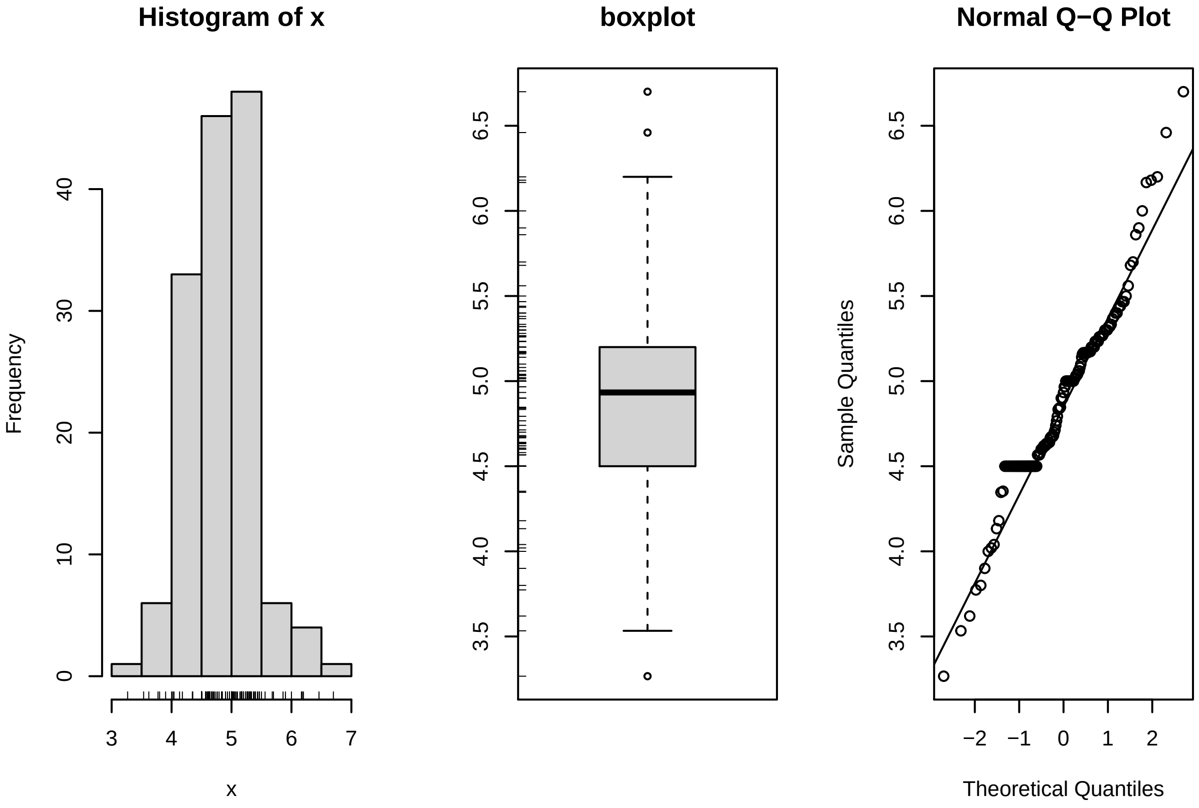
Wind Speed at Capture

```
morpho_blood_SMI %>%
  ggplot(., aes(x = Wind_mph_interpol)) +
  geom_histogram(color = "black", fill="steelblue", bins=30) +
  theme_classic() +
  xlab("Wind Speed (mph)") +
  ylab("Count")
```

```
## Warning: Removed 3 rows containing non-finite values (stat_bin).
```



```
simple.eda(morpho_blood_SMI$Wind_mph_interpol)
```



```
# Normality test if p > .05, data is normal. Data is not normal
shapiro.test(morpho_blood_SMI$Wind_mph_interpol)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  morpho_blood_SMI$Wind_mph_interpol
## W = 0.96149, p-value = 0.0004364
```

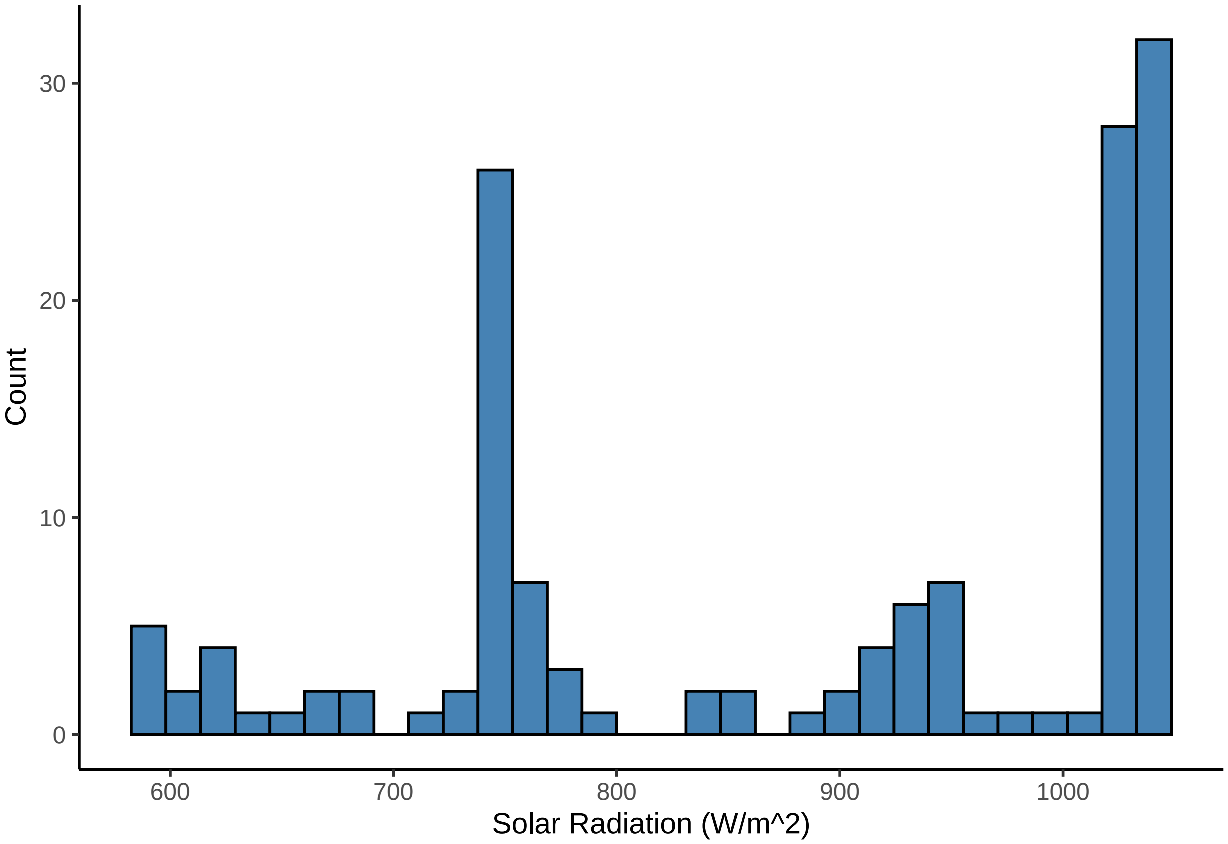
```
# ln transformation doesn't fix non normality
shapiro.test(log(morpho_blood_SMI$Wind_mph_interpol))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  log(morpho_blood_SMI$Wind_mph_interpol)
## W = 0.95827, p-value = 0.0002241
```

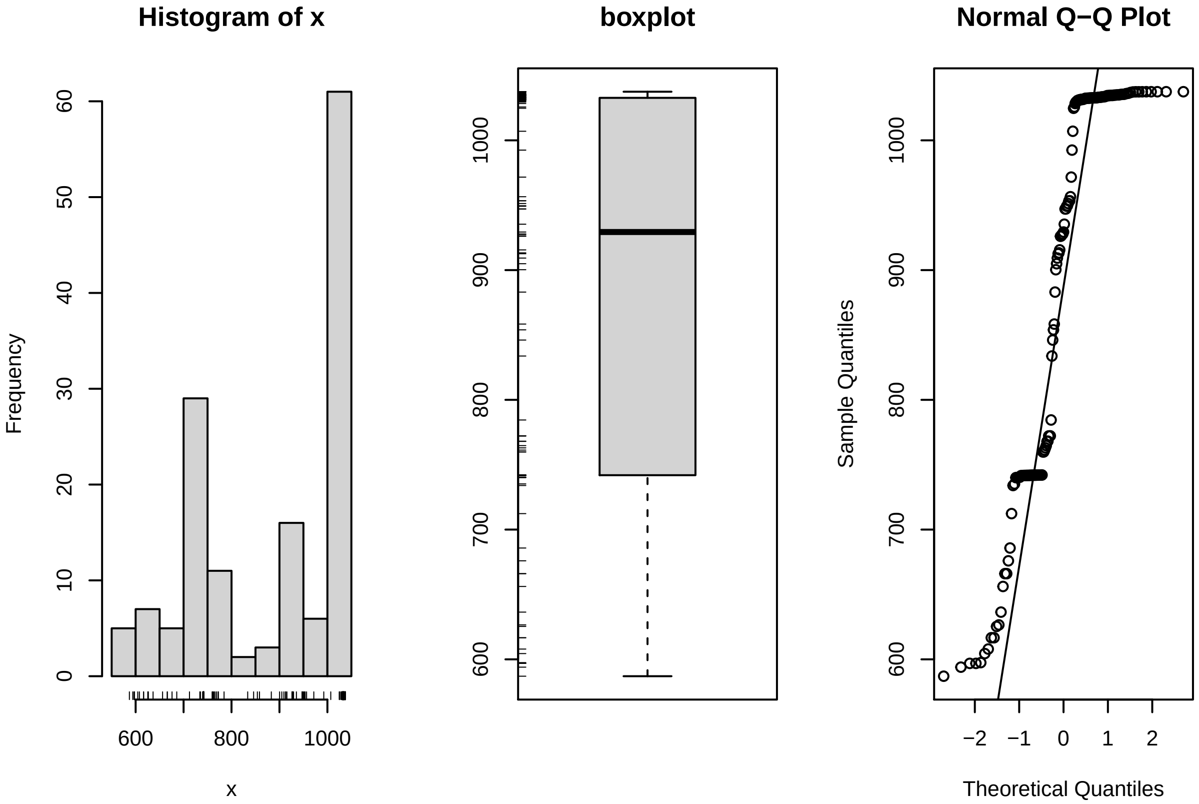
Solar Radiation at Capture

```
morpho_blood_SMI %>%
  ggplot(., aes(x = Solar_rad_Wm2_interpol)) +
  geom_histogram(color = "black", fill="steelblue", bins=30) +
  theme_classic() +
  xlab("Solar Radiation (W/m^2)") +
  ylab("Count")
```

```
## Warning: Removed 3 rows containing non-finite values (stat_bin).
```



```
simple.eda(morpho_blood_SMI$Solar_rad_Wm2_interpol)
```



```r
# Normality test if p > .05, data is normal. Data is not normal
shapiro.test(morpho_blood_SMI$Solar_rad_Wm2_interpol)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  morpho_blood_SMI$Solar_rad_Wm2_interpol
## W = 0.82946, p-value = 1.072e-11
```

```r
# Doesn't fix non normality
shapiro.test(log(morpho_blood_SMI$Solar_rad_Wm2_interpol))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  log(morpho_blood_SMI$Solar_rad_Wm2_interpol)
## W = 0.829, p-value = 1.028e-11
```

Conclusion

Osmolality and SMI were the only normally distributed variables.

The following variables each had non-normal distributions: - SVL (skewed left) - mass (skewed left) - hct (looks like a bell curve, but not very pretty) - cloacal temp (skewed right) - CEWL (skewed right, overall and for each region individually) - capture temp (multimodal) - wind speed (multimodal but sorta bell curve) - solar radiation (multimodal)

All of the non-normal variables are likely to need transformation if they're included in the final model.

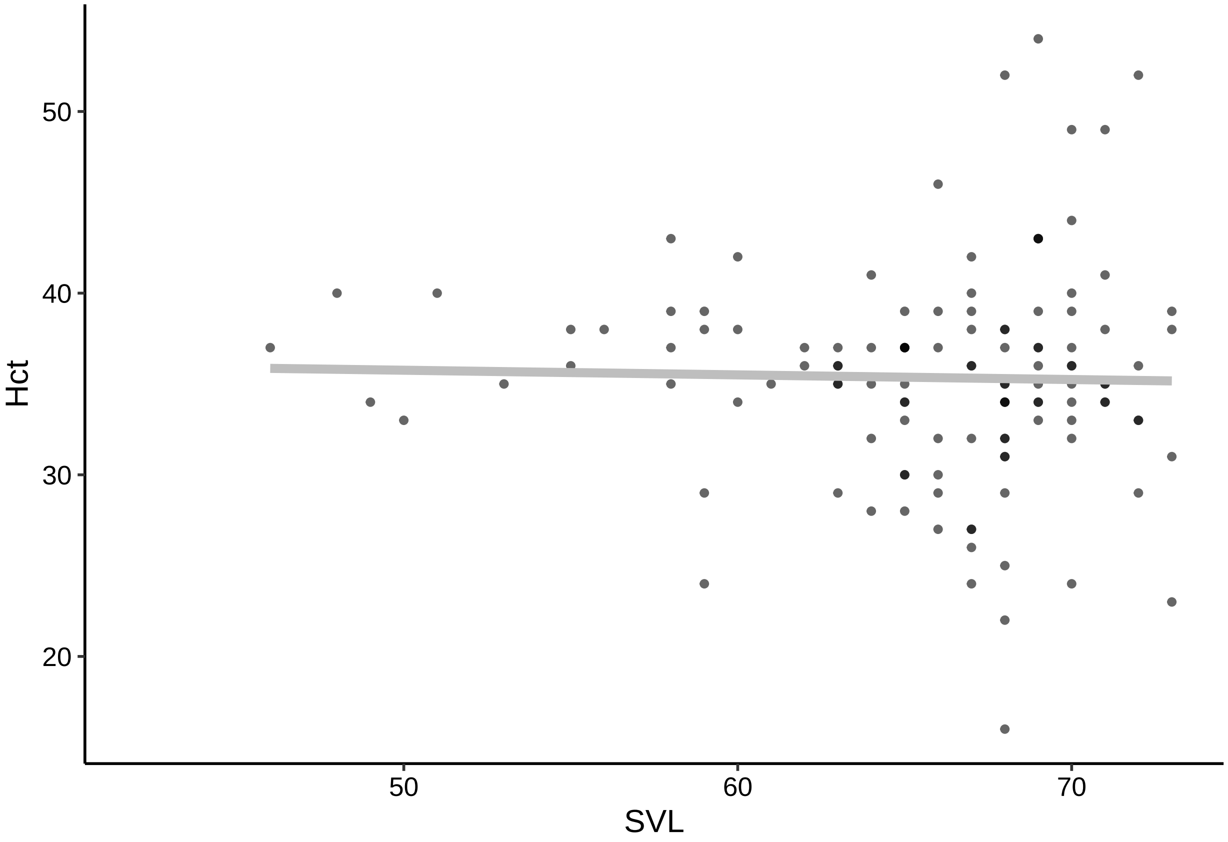
Basic Figs & LMs

What affects osmolality & hematocrit?

Hct ~ SVL

```r
# plot
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_point(aes(x = SVL_mm,
                 y = hematocrit_percent,
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = SVL_mm,
                  y = hematocrit_percent,
                  ),
              formula = y ~ x,
              method = "lm",
              color = "gray",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("SVL") +
  ylab("Hct") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 27 rows containing non-finite values (stat_smooth).

## Warning: Removed 27 rows containing missing values (geom_point).
```



```
# lm
lm_hct_SVL <- lm(hematocrit_percent ~ SVL_mm,
           data = morpho_blood_SMI)
summary(lm_hct_SVL)
```

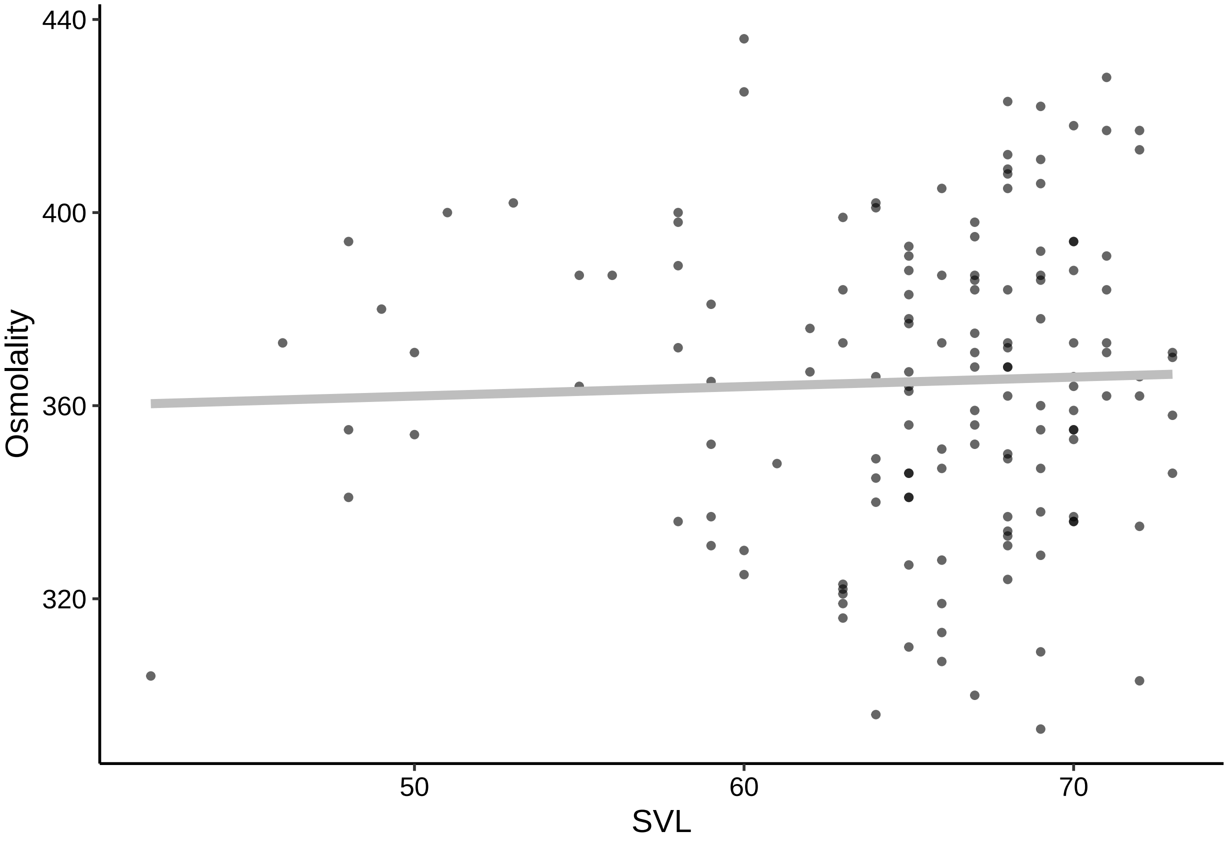
```
##
## Call:
## lm(formula = hematocrit_percent ~ SVL_mm, data = morpho_blood_SMI)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.2957  -2.7565  -0.2189   2.7043  18.7299
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 37.03634    6.51586   5.684 9.54e-08 ***
## SVL_mm      -0.02560    0.09888  -0.259    0.796
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.953 on 119 degrees of freedom
##   (27 observations deleted due to missingness)
## Multiple R-squared:  0.0005628,  Adjusted R-squared:  -0.007836
## F-statistic: 0.06702 on 1 and 119 DF,  p-value: 0.7962
```

Osml ~ SVL

```
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_point(aes(x = SVL_mm,
                 y = osmolality_mmol_kg,
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = SVL_mm,
                  y = osmolality_mmol_kg,
                  ),
              formula = y ~ x,
              method = "lm",
              color = "gray",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("SVL") +
  ylab("Osmolality") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 3 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 3 rows containing missing values (geom_point).
```



```
# lm
lm_osml_SVL <- lm(osmolality_mmol_kg ~ SVL_mm,
           data = morpho_blood_SMI)
summary(lm_osml_SVL)
```

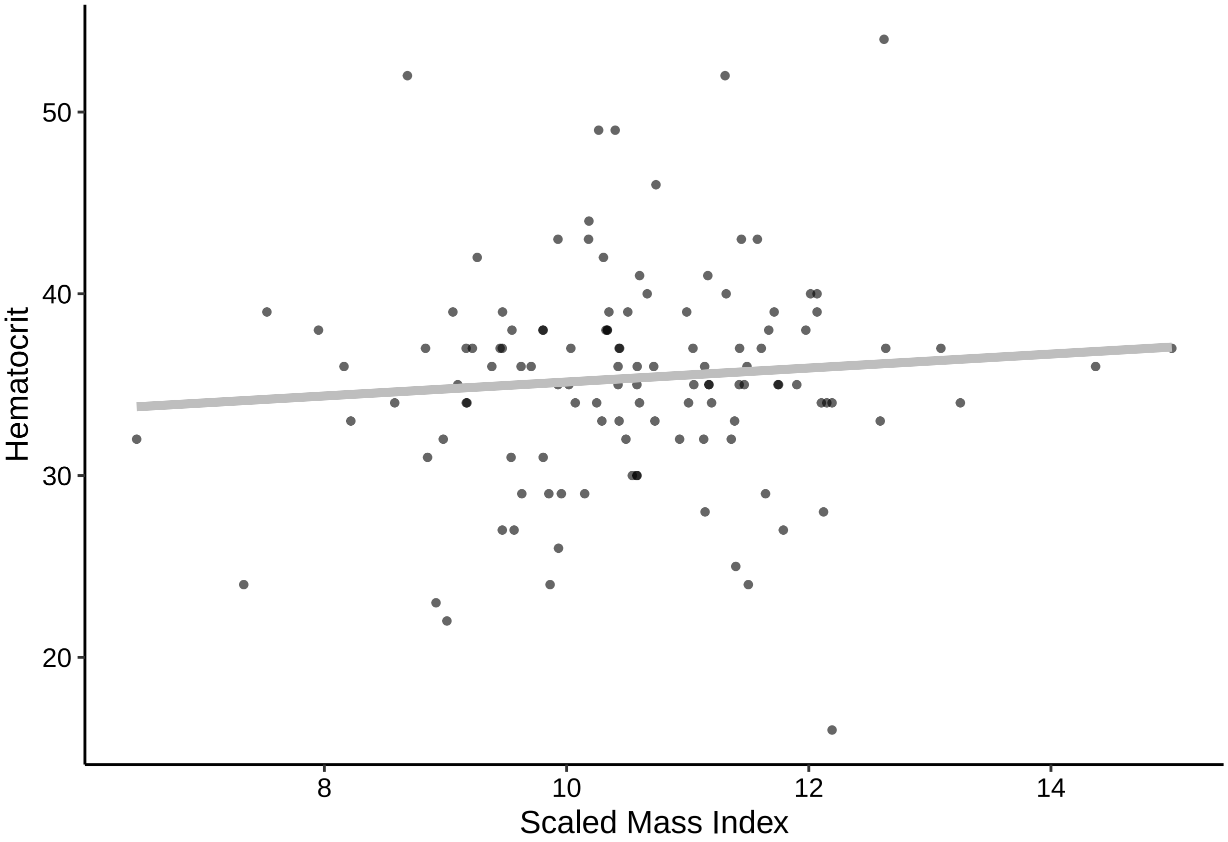
```
##
## Call:
## lm(formula = osmolality_mmol_kg ~ SVL_mm, data = morpho_blood_SMI)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -72.716 -20.584   1.268  22.088  72.055
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 352.1404    28.1557  12.507   <2e-16 ***
## SVL_mm        0.1967     0.4305   0.457    0.648
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 31.27 on 143 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.001458,   Adjusted R-squared:  -0.005525
## F-statistic: 0.2088 on 1 and 143 DF,  p-value: 0.6484
```

Hct ~ SMI

```
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_point(aes(x = SMI,
                 y = hematocrit_percent,
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = SMI,
                  y = hematocrit_percent,
                  ),
              formula = y ~ x,
              method = "lm",
              color = "gray",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Scaled Mass Index") +
  ylab("Hematocrit") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 27 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 27 rows containing missing values (geom_point).
```



```r
# lm
lm_hct_mass <- lm(hematocrit_percent ~ SMI,
           data = morpho_blood_SMI)
summary(lm_hct_mass)
```

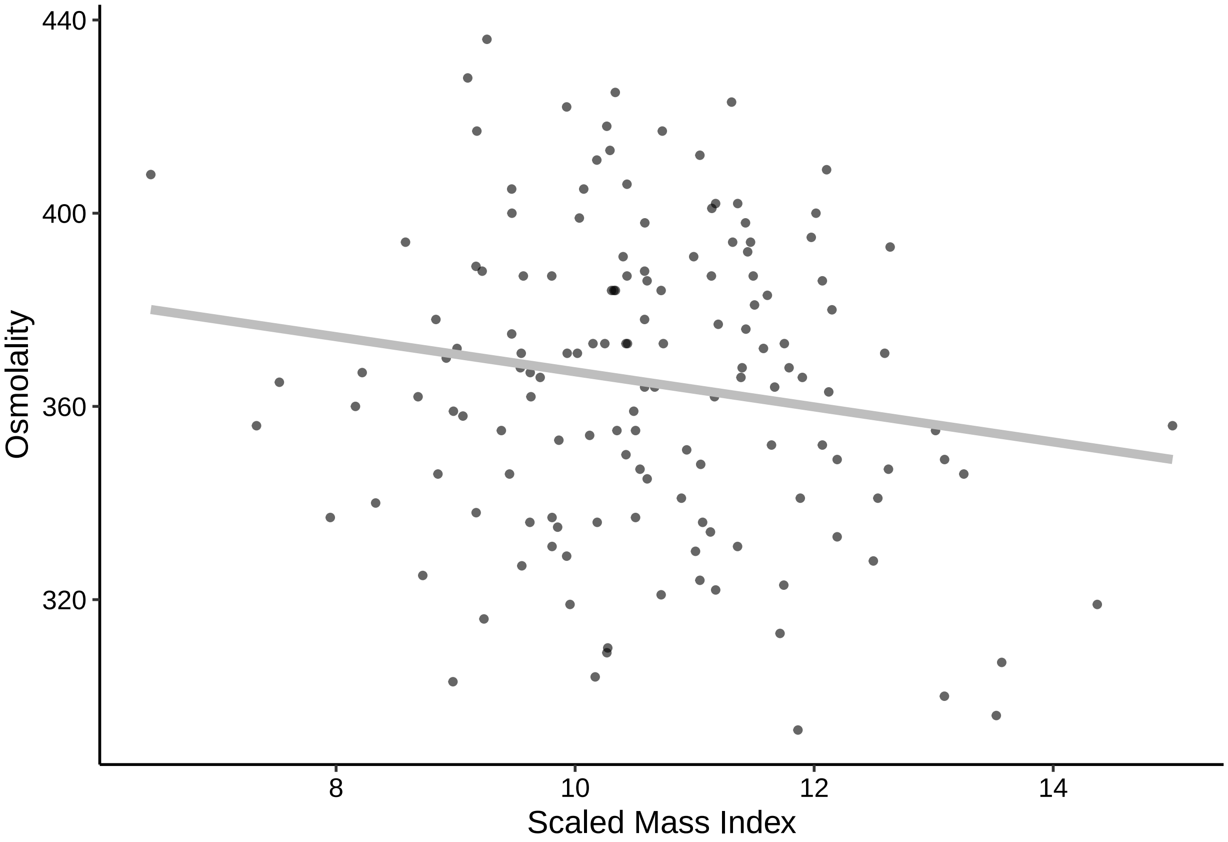
```
##
## Call:
## lm(formula = hematocrit_percent ~ SMI, data = morpho_blood_SMI)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.990  -2.680  -0.120   2.927  17.845
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  31.3051     4.3249   7.238 4.84e-11 ***
## SMI           0.3842     0.4071   0.944    0.347
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.933 on 119 degrees of freedom
##   (27 observations deleted due to missingness)
## Multiple R-squared:  0.007431,   Adjusted R-squared:  -0.0009099
## F-statistic: 0.8909 on 1 and 119 DF,  p-value: 0.3471
```

Osml ~ SMI

```
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_point(aes(x = SMI,
                 y = osmolality_mmol_kg,
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = SMI,
                  y = osmolality_mmol_kg),
              formula = y ~ x,
              method = "lm",
              se = F,
              color = "gray",
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Scaled Mass Index") +
  ylab("Osmolality") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 3 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 3 rows containing missing values (geom_point).
```



```r
# lm
lm_osml_mass <- lm(osmolality_mmol_kg ~ SMI,
           data = morpho_blood_SMI)
summary(lm_osml_mass)
```

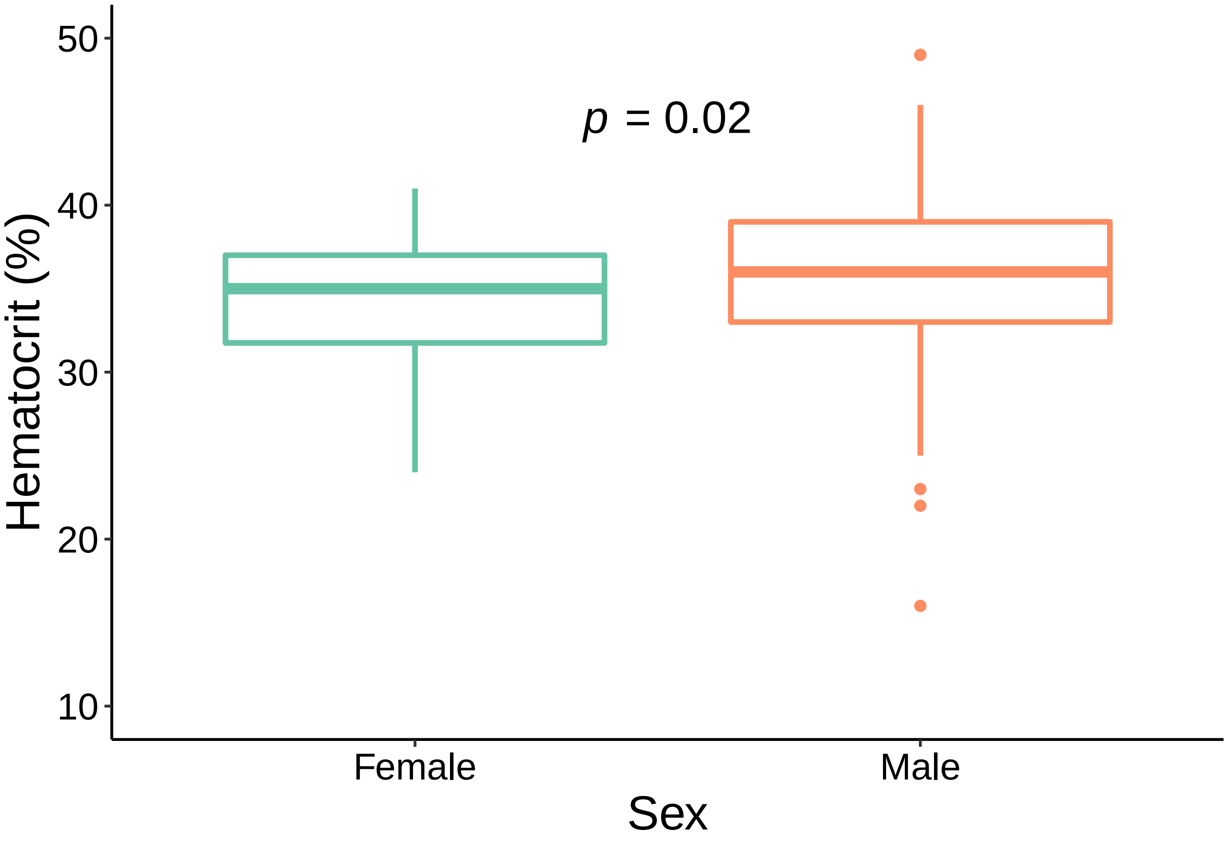
```
##
## Call:
## lm(formula = osmolality_mmol_kg ~ SMI, data = morpho_blood_SMI)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -67.879 -22.935   1.245  21.677  66.154
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  403.490     20.121  20.053   <2e-16 ***
## SMI           -3.632      1.881  -1.931   0.0555 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 30.89 on 143 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.02541,    Adjusted R-squared:  0.0186
## F-statistic: 3.729 on 1 and 143 DF,  p-value: 0.05545
```

Hct ~ Sex

males have significantly higher hematocrit %

```
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = sex_M_F,
                   y = hematocrit_percent,
                   color = sex_M_F
                   ),
               size = 1,
               alpha = 1) +
  theme_classic() +
  xlab("Sex") +
  ylab("Hematocrit (%)") +
  annotate("text", x = 1.5, y = 45,
           label = "paste(italic(p), \" = 0.02\")",
           parse = TRUE,
           size = 6) +
  ylim(10, 50) +
  scale_x_discrete(labels = c("F" = "Female",
                              "M" = "Male")) +
  scale_color_brewer(palette = "Set2") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 18),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 14),
        legend.text.align = 0,
        legend.position = "none"
) -> hct_sex_fig
hct_sex_fig
```

```
## Warning: Removed 30 rows containing non-finite values (stat_boxplot).
```



```
# export figure
#ggsave(filename = "hct_sex_fig.jpeg",
 #      plot = hct_sex_fig,
  #     path = "./final_figures",
   #    device = "jpeg",
    #   dpi = 1200,
     #  width = 6, height = 4)

# lms
lm_hct_sex_mass <- lm(hematocrit_percent ~ sex_M_F*mass_g - mass_g,
           data = morpho_blood_SMI)
summary(lm_hct_sex_mass)
```

```
##
## Call:
## lm(formula = hematocrit_percent ~ sex_M_F * mass_g - mass_g,
##     data = morpho_blood_SMI)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20.6306  -2.6187   0.6422   3.1481  17.2271
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      35.9163     3.7236   9.646   <2e-16 ***
## sex_M_FM         -0.9224     4.7553  -0.194    0.847
## sex_M_FF:mass_g  -0.2244     0.3596  -0.624    0.534
## sex_M_FM:mass_g   0.1186     0.2568   0.462    0.645
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.847 on 117 degrees of freedom
##   (27 observations deleted due to missingness)
## Multiple R-squared:  0.05203,    Adjusted R-squared:  0.02772
## F-statistic: 2.141 on 3 and 117 DF,  p-value: 0.09884
```

```
lm_hct_sex <- lm(hematocrit_percent ~ sex_M_F,
           data = morpho_blood_SMI)
summary(lm_hct_sex)
```

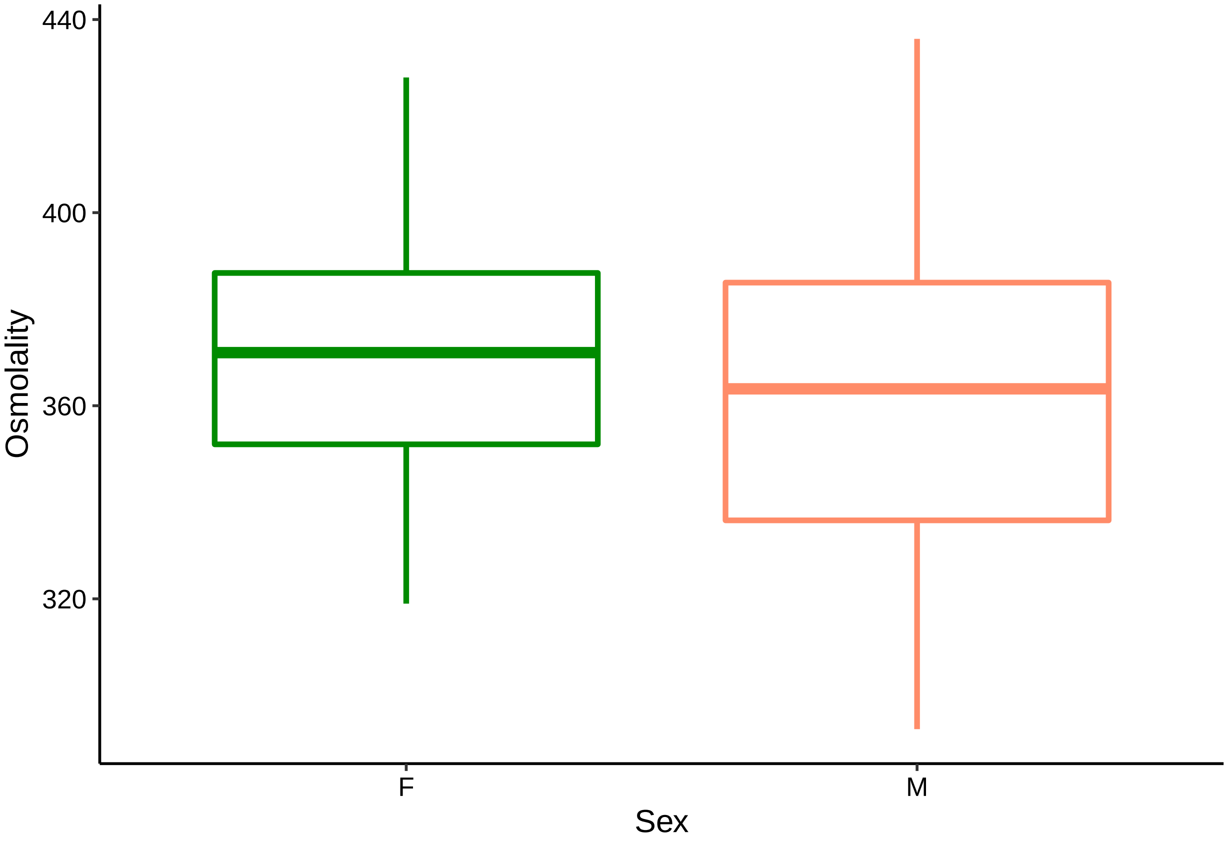
```
##
## Call:
## lm(formula = hematocrit_percent ~ sex_M_F, data = morpho_blood_SMI)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20.3247  -3.3247   0.6753   3.3409  17.6753
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  33.6591     0.8763  38.410   <2e-16 ***
## sex_M_FM      2.6656     1.0985   2.427   0.0167 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.813 on 119 degrees of freedom
##   (27 observations deleted due to missingness)
## Multiple R-squared:  0.04715,    Adjusted R-squared:  0.03914
## F-statistic: 5.888 on 1 and 119 DF,  p-value: 0.01674
```

Hematocrit is significantly predicted by sex, but the interaction between sex and mass is ~nonexistent.

Osml ~ Sex

```
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = sex_M_F,
                   y = osmolality_mmol_kg,
                   color = sex_M_F
                   ),
               size = 1,
               alpha = 0.6) +
  scale_colour_manual(name = "Sex",
                      values = c("green4", "salmon1") ) +
  theme_classic() +
  xlab("Sex") +
  ylab("Osmolality") +
  theme(text = element_text(color = "black", family = "sans", size = 12),
        axis.text = element_text(color = "black", family = "sans", size = 10),
        legend.text.align = 0,
        legend.position = "none"
)
```

```
## Warning: Removed 3 rows containing non-finite values (stat_boxplot).
```



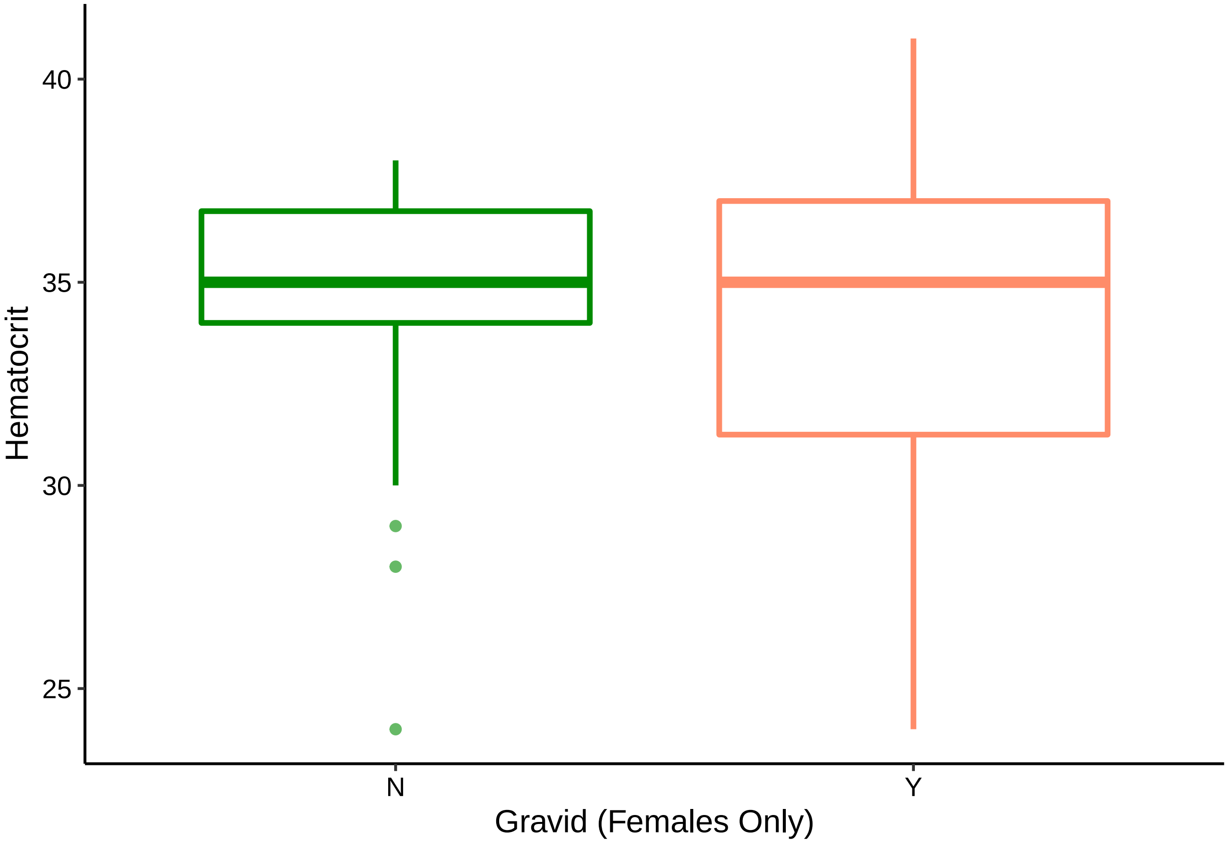
```
# lm
lm_osml_sex <- lm(osmolality_mmol_kg ~ sex_M_F,
           data = morpho_blood_SMI)
summary(lm_osml_sex)
```

```
##
## Call:
## lm(formula = osmolality_mmol_kg ~ sex_M_F, data = morpho_blood_SMI)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -68.612 -22.915   1.085  22.388  74.388
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  371.915      4.509  82.479   <2e-16 ***
## sex_M_FM     -10.303      5.485  -1.878   0.0624 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 30.91 on 143 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.02408,    Adjusted R-squared:  0.01725
## F-statistic: 3.528 on 1 and 143 DF,  p-value: 0.06237
```

Hct ~ Gravidity

```
morpho_blood_SMI %>%
  dplyr::filter(sex_M_F == 'F') %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = gravid_Y_N,
                   y = hematocrit_percent,
                   color = gravid_Y_N
                   ),
               size = 1,
               alpha = 0.6) +
  scale_colour_manual(name = "Gravid",
                      values = c("green4", "salmon1") ) +
  theme_classic() +
  xlab("Gravid (Females Only)") +
  ylab("Hematocrit") +
  theme(text = element_text(color = "black", family = "sans", size = 12),
        axis.text = element_text(color = "black", family = "sans", size = 10),
        legend.text.align = 0,
        legend.position = "none"
)
```

```
## Warning: Removed 4 rows containing non-finite values (stat_boxplot).
```



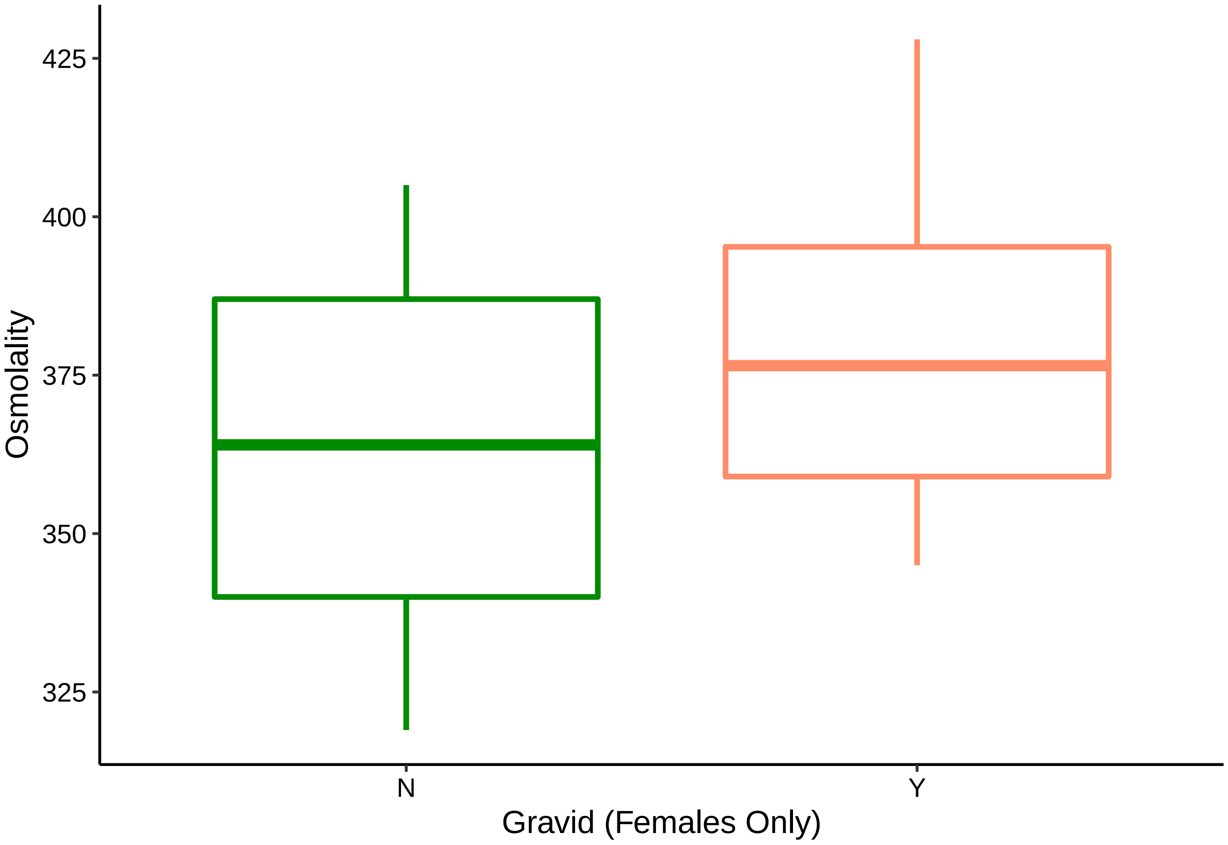
```
# lm
lm_hct_gravid <- lm(hematocrit_percent ~ gravid_Y_N,
           data = morpho_blood_SMI)
summary(lm_hct_gravid)
```

```
##
## Call:
## lm(formula = hematocrit_percent ~ gravid_Y_N, data = morpho_blood_SMI)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.111  -1.596   1.271   3.080   7.654
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   34.111      1.017  33.546   <2e-16 ***
## gravid_Y_NY   -0.765      1.323  -0.578    0.566
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.314 on 42 degrees of freedom
##   (104 observations deleted due to missingness)
## Multiple R-squared:  0.0079, Adjusted R-squared:  -0.01572
## F-statistic: 0.3344 on 1 and 42 DF,  p-value: 0.5662
```

Osml ~ Gravidity

```
morpho_blood_SMI %>%
  dplyr::filter(sex_M_F == 'F') %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = gravid_Y_N,
                   y = osmolality_mmol_kg,
                   color = gravid_Y_N
                   ),
               size = 1,
               alpha = 0.6) +
  scale_colour_manual(name = "Gravid",
                      values = c("green4", "salmon1") ) +
  theme_classic() +
  xlab("Gravid (Females Only)") +
  ylab("Osmolality") +
  theme(text = element_text(color = "black", family = "sans", size = 12),
        axis.text = element_text(color = "black", family = "sans", size = 10),
        legend.text.align = 0,
        legend.position = "none"
)
```

```
## Warning: Removed 1 rows containing non-finite values (stat_boxplot).
```



```
# lm
lm_osml_gravid <- lm(osmolality_mmol_kg ~ gravid_Y_N,
           data = morpho_blood_SMI)
summary(lm_osml_gravid)
```

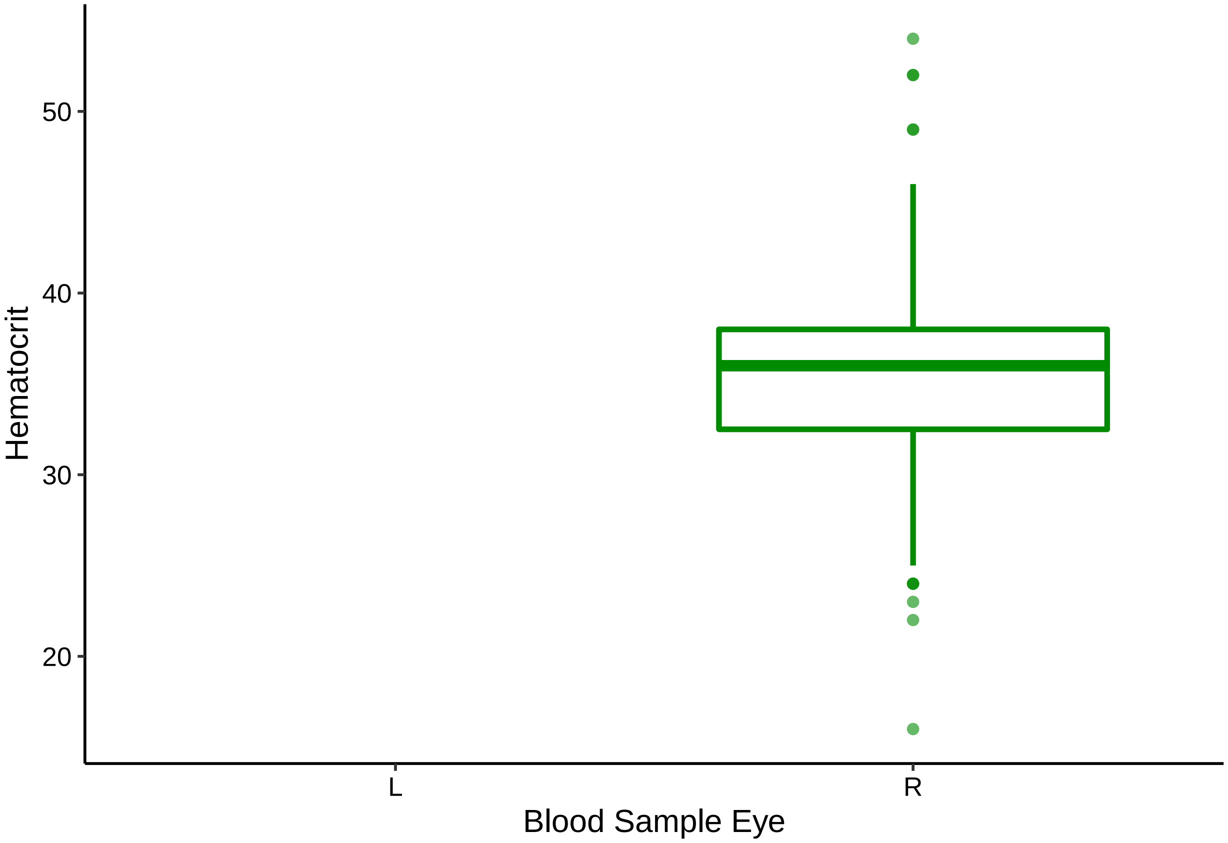
```
##
## Call:
## lm(formula = osmolality_mmol_kg ~ gravid_Y_N, data = morpho_blood_SMI)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -45.333 -22.186  -0.333  22.667  49.962
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  364.333      5.604  65.011   <2e-16 ***
## gravid_Y_NY   13.705      7.535   1.819   0.0756 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.68 on 45 degrees of freedom
##   (101 observations deleted due to missingness)
## Multiple R-squared:  0.06848,    Adjusted R-squared:  0.04778
## F-statistic: 3.308 on 1 and 45 DF,  p-value: 0.07559
```

Hct ~ Sample Eye

Actually, any blood samples not taken from the right eye ended up being excluded or didn't have hematocrit values, so we can't test this difference.

```
morpho_blood_SMI %>%
  dplyr::filter(blood_sample_eye %in% c("R", "L")) %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = blood_sample_eye,
                   y = hematocrit_percent,
                   color = blood_sample_eye
                   ),
               size = 1,
               alpha = 0.6) +
  scale_colour_manual(name = "Blood Sample Eye",
                      values = c("green4", "salmon1") ) +
  theme_classic() +
  xlab("Blood Sample Eye") +
  ylab("Hematocrit") +
  theme(text = element_text(color = "black", family = "sans", size = 12),
        axis.text = element_text(color = "black", family = "sans", size = 10),
        legend.text.align = 0,
        legend.position = "none"
)
```

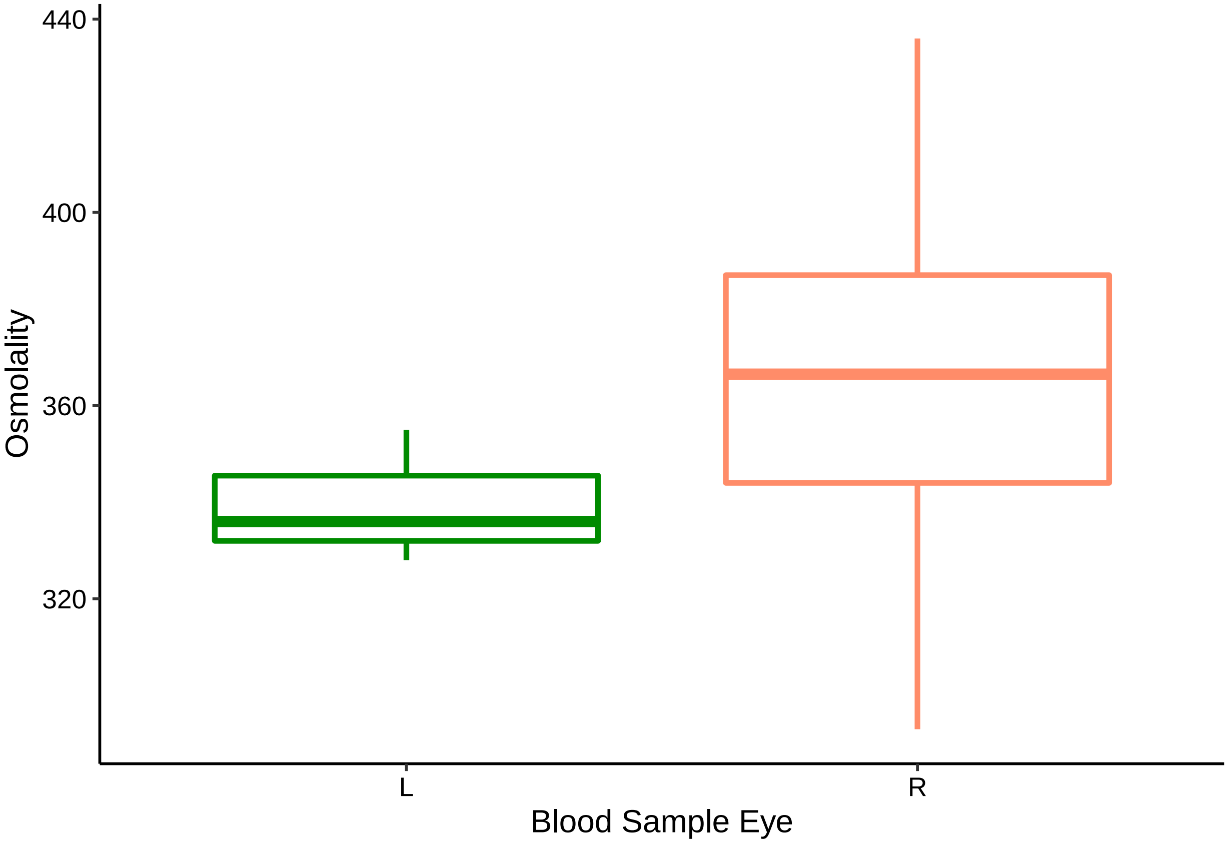
```
## Warning: Removed 27 rows containing non-finite values (stat_boxplot).
```



Osml ~ Sample Eye

```
morpho_blood_SMI %>%
  dplyr::filter(blood_sample_eye %in% c("R", "L")) %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = blood_sample_eye,
                   y = osmolality_mmol_kg,
                   color = blood_sample_eye
                   ),
               size = 1,
               alpha = 0.6) +
  scale_colour_manual(name = "Blood Sample Eye",
                      values = c("green4", "salmon1") ) +
  theme_classic() +
  xlab("Blood Sample Eye") +
  ylab("Osmolality") +
  theme(text = element_text(color = "black", family = "sans", size = 12),
        axis.text = element_text(color = "black", family = "sans", size = 10),
        legend.text.align = 0,
        legend.position = "none"
)
```

```
## Warning: Removed 3 rows containing non-finite values (stat_boxplot).
```



```
# lm
lm_osml_eye <- lm(osmolality_mmol_kg ~ blood_sample_eye,
           data = morpho_blood_SMI)
summary(lm_osml_eye)
```

```
##
## Call:
## lm(formula = osmolality_mmol_kg ~ blood_sample_eye, data = morpho_blood_SMI)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -71.986 -19.986   1.014  22.014  71.014
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          400.50      21.85  18.330   <2e-16 ***
## blood_sample_eyeL    -60.83      28.21  -2.157   0.0327 *
## blood_sample_eyeR    -35.51      22.01  -1.614   0.1088
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 30.9 on 142 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.03175,    Adjusted R-squared:  0.01811
## F-statistic: 2.328 on 2 and 142 DF,  p-value: 0.1012
```

```r
summary(morpho_blood_SMI$blood_sample_eye)
```

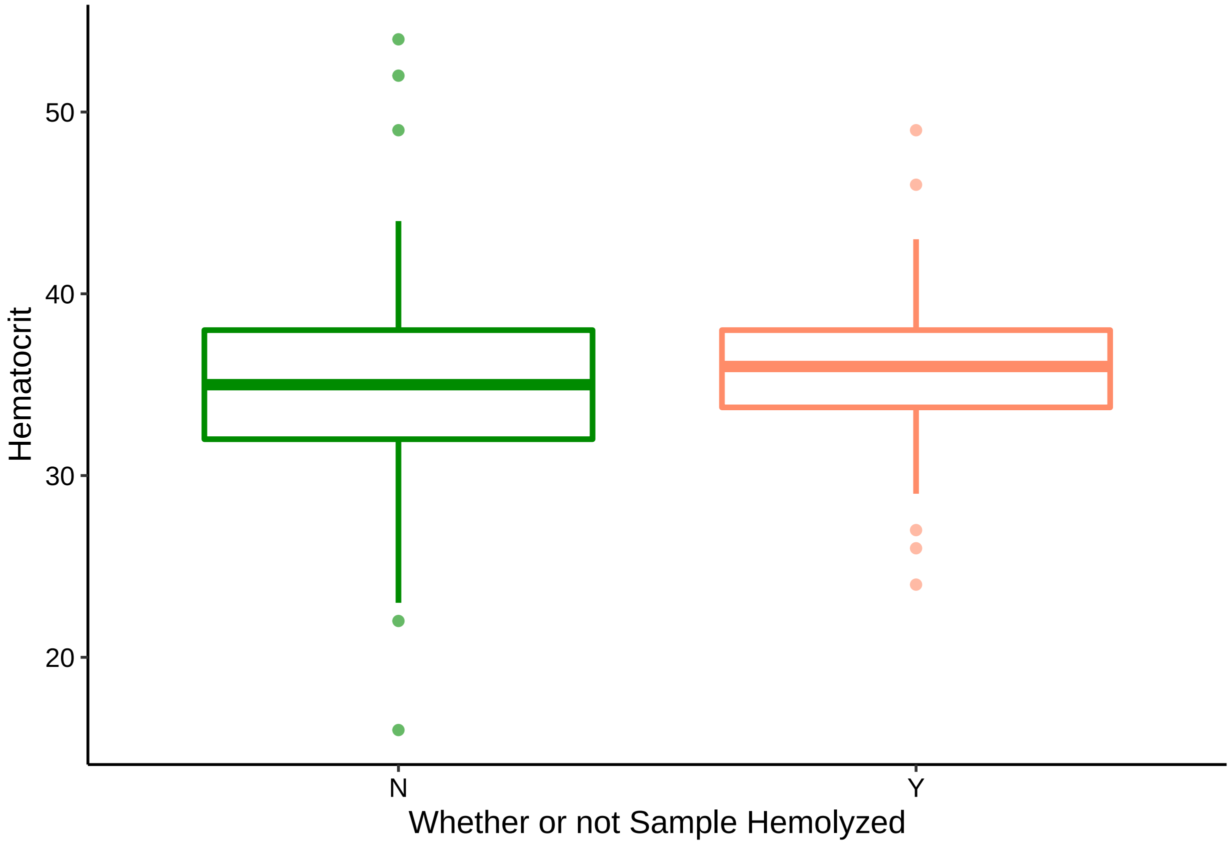
```
## both    L    R
##    2    4  142
```

There IS a significant difference, but upon further inspection, the number of blood draws taken from the right postorbital sinus vs left or both is so uneven, this would be unhelpful in the actual model.

Hct ~ Hemolyzed/Not

```r
morpho_blood_SMI %>%
  dplyr::filter(hemolyzed %in% c("Y", "N")) %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = hemolyzed,
                   y = hematocrit_percent,
                   color = hemolyzed
                   ),
               size = 1,
               alpha = 0.6) +
  scale_colour_manual(name = "Blood Sample Eye",
                      values = c("green4", "salmon1", "green4", "salmon1") ) +
  theme_classic() +
  xlab("Whether or not Sample Hemolyzed") +
  ylab("Hematocrit") +
  theme(text = element_text(color = "black", family = "sans", size = 12),
        axis.text = element_text(color = "black", family = "sans", size = 10),
        legend.text.align = 0,
        legend.position = "none"
)
```

```
## Warning: Removed 25 rows containing non-finite values (stat_boxplot).
```



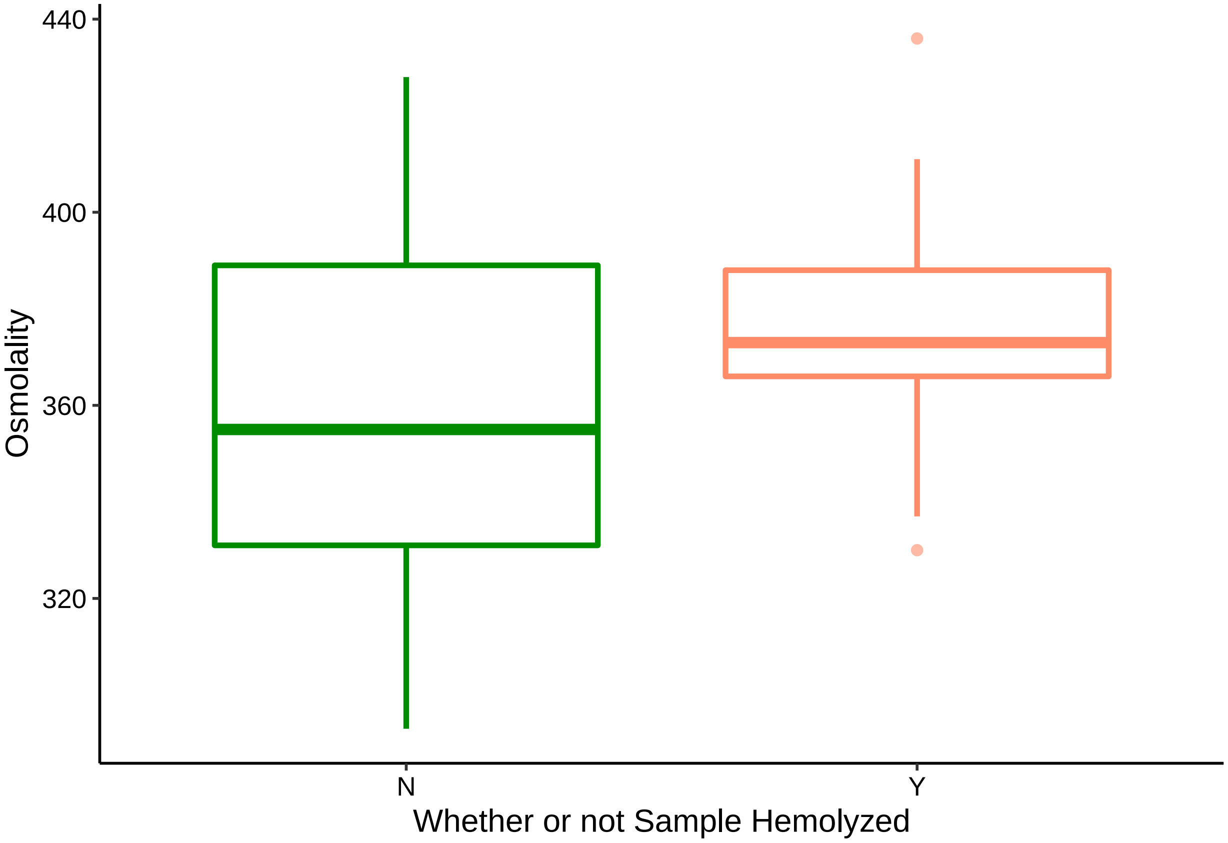
```r
# lm
lm_hct_hem <- lm(hematocrit_percent ~ hemolyzed,
           data = morpho_blood_SMI)
summary(lm_hct_hem)
```

```
##
## Call:
## lm(formula = hematocrit_percent ~ hemolyzed, data = morpho_blood_SMI)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.0952  -2.7778  -0.0952   2.2222  18.9048
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  35.0952     0.7636  45.958   <2e-16 ***
## hemolyzedY    0.6825     1.2664   0.539    0.591
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.061 on 97 degrees of freedom
##   (49 observations deleted due to missingness)
## Multiple R-squared:  0.002986,   Adjusted R-squared:  -0.007293
## F-statistic: 0.2905 on 1 and 97 DF,  p-value: 0.5911
```

Osml ~ Hemolyzed/Not

```
morpho_blood_SMI %>%
  dplyr::filter(hemolyzed %in% c("Y", "N")) %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = hemolyzed,
                   y = osmolality_mmol_kg,
                   color = hemolyzed
                   ),
               size = 1,
               alpha = 0.6) +
  scale_colour_manual(name = "Blood Sample Eye",
                      values = c("green4", "salmon1", "green4", "salmon1") ) +
  theme_classic() +
  xlab("Whether or not Sample Hemolyzed") +
  ylab("Osmolality") +
  theme(text = element_text(color = "black", family = "sans", size = 12),
        axis.text = element_text(color = "black", family = "sans", size = 10),
        legend.text.align = 0,
        legend.position = "none"
)
```

```
## Warning: Removed 1 rows containing non-finite values (stat_boxplot).
```



```
# lm
lm_osml_hem <- lm(osmolality_mmol_kg ~ hemolyzed,
           data = morpho_blood_SMI)
summary(lm_osml_hem)
```

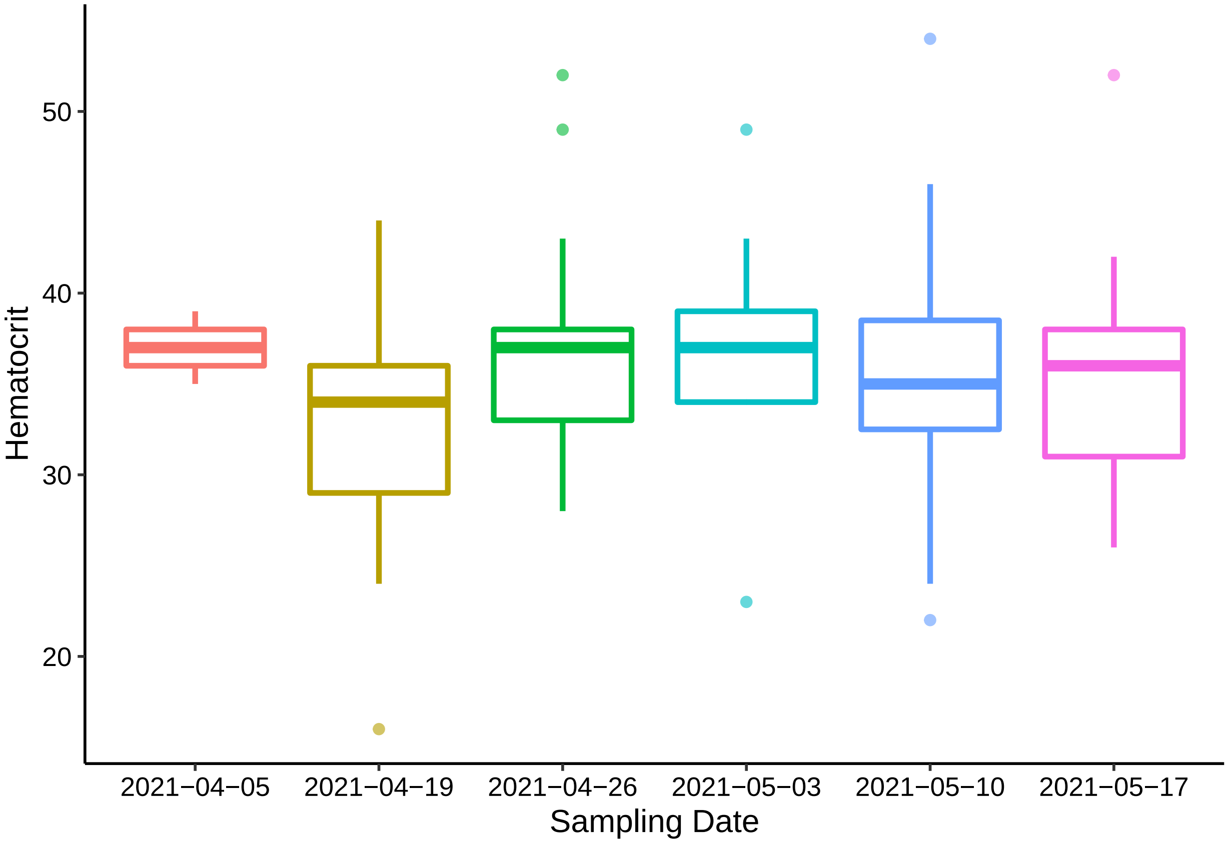
```
##
## Call:
## lm(formula = osmolality_mmol_kg ~ hemolyzed, data = morpho_blood_SMI)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -66.690 -23.858  -2.026  24.310  68.310
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  359.690      3.523 102.109   <2e-16 ***
## hemolyzedY    15.335      6.256   2.451   0.0157 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.29 on 121 degrees of freedom
##   (25 observations deleted due to missingness)
## Multiple R-squared:  0.04731,    Adjusted R-squared:  0.03944
## F-statistic: 6.009 on 1 and 121 DF,  p-value: 0.01566
```

Osmolality in hemolyzed samples versus not are significantly different! This is an unfortunate trait of the dataset, and not something we are specifically interested in testing, so we will include it as a random factor in the LMM.

Hct ~ Week

```
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = as.factor(date),
                   y = hematocrit_percent,
                   color = as.factor(date)
                   ),
               size = 1,
               alpha = 0.6) +
  theme_classic() +
  xlab("Sampling Date") +
  ylab("Hematocrit") +
  theme(text = element_text(color = "black", family = "sans", size = 12),
        axis.text = element_text(color = "black", family = "sans", size = 10),
        legend.text.align = 0,
        legend.position = "none"
)
```

```
## Warning: Removed 27 rows containing non-finite values (stat_boxplot).
```



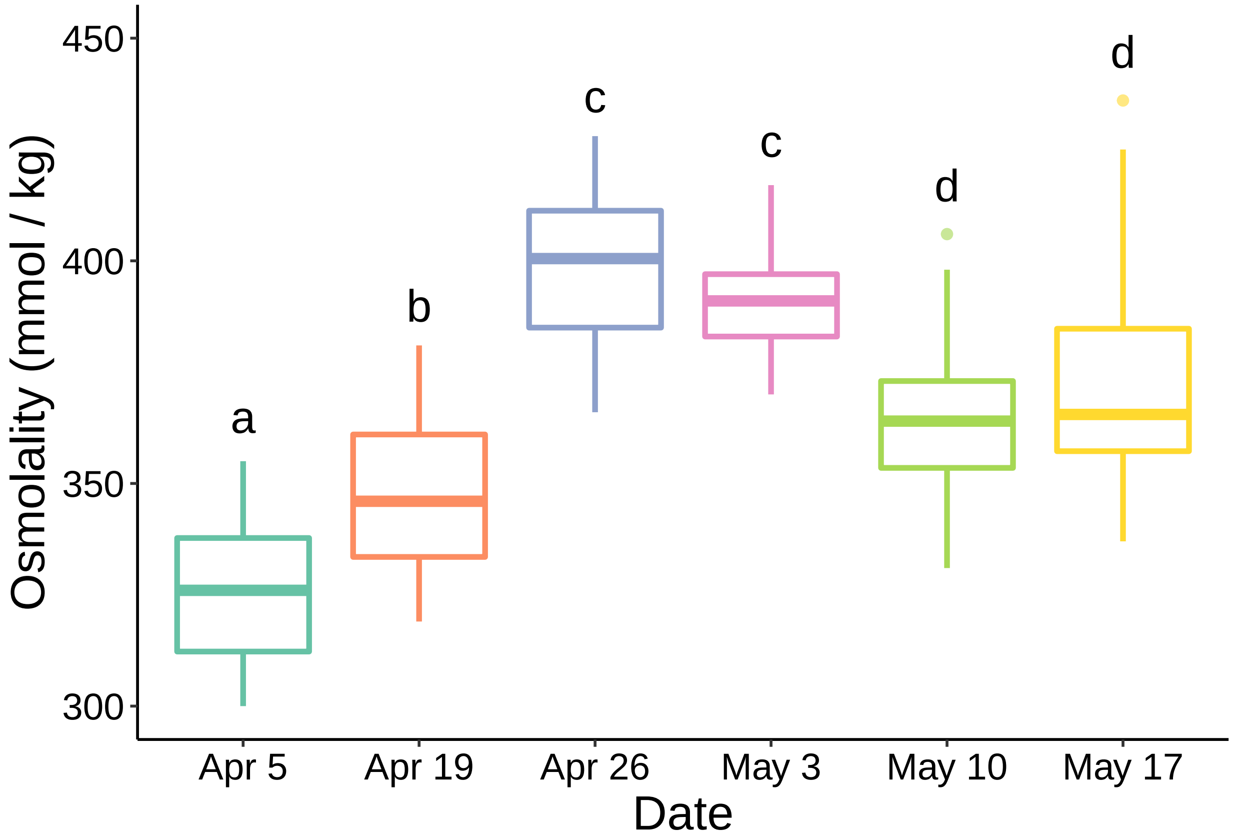
```
# lm
lm_hct_wk <- lm(hematocrit_percent ~ date,
           data = morpho_blood_SMI)
summary(lm_hct_wk)
```

```
##
## Call:
## lm(formula = hematocrit_percent ~ date, data = morpho_blood_SMI)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.8497  -2.8497   0.1348   2.6425  18.3886
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -644.69394  942.77212  -0.684    0.495
## date           0.03627    0.05028   0.721    0.472
##
## Residual standard error: 5.942 on 119 degrees of freedom
##   (27 observations deleted due to missingness)
## Multiple R-squared:  0.004353,  Adjusted R-squared:  -0.004013
## F-statistic: 0.5203 on 1 and 119 DF,  p-value: 0.4721
```

Osml ~ Week

```
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = as.factor(date),
                   y = osmolality_mmol_kg,
                   color = as.factor(date)
                   ),
               size = 1,
               alpha = 0.6) +
  theme_classic() +
  xlab("Date") +
  ylab("Osmolality (mmol / kg)") +
  annotate("text", x = 1, y = 365, label = "a", size = 6) +
  annotate("text", x = 2, y = 390, label = "b", size = 6) +
  annotate("text", x = 3, y = 437, label = "c", size = 6) +
  annotate("text", x = 4, y = 427, label = "c", size = 6) +
  annotate("text", x = 5, y = 417, label = "d", size = 6) +
  annotate("text", x = 6, y = 447, label = "d", size = 6) +
  scale_x_discrete(labels = c("2021-04-05" = "Apr 5",
                              "2021-04-19" = "Apr 19",
                              "2021-04-26" = "Apr 26",
                              "2021-05-03" = "May 3",
                              "2021-05-10" = "May 10",
                              "2021-05-17" = "May 17")) +
  scale_color_brewer(palette = "Set2") +
  ylim(300, 450) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 18),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 14),
        legend.text.align = 0,
        legend.position = "none"
) -> osml_date_fig
osml_date_fig
```

```
## Warning: Removed 5 rows containing non-finite values (stat_boxplot).
```



```
# export figure
#ggsave(filename = "osml_date_fig.tiff",
 #      plot = osml_date_fig,
  #     path = "./final_figures",
   #    device = "tiff",
    #   dpi = 1200,
     #  width = 6, height = 4)

# lm
lm_osml_wk <- glm(osmolality_mmol_kg ~ date,
           data = morpho_blood_SMI)
summary(lm_osml_wk)
```

```
##
## Call:
## glm(formula = osmolality_mmol_kg ~ date, data = morpho_blood_SMI)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -48.236  -21.636   -2.635   20.364   65.564
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.999e+04  2.954e+03  -6.766 3.15e-10 ***
## date         1.086e+00  1.576e-01   6.889 1.65e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for gaussian family taken to be 735.2172)
##
##     Null deviance: 140031  on 144  degrees of freedom
## Residual deviance: 105136  on 143  degrees of freedom
##   (3 observations deleted due to missingness)
## AIC: 1372.5
##
## Number of Fisher Scoring iterations: 2
```

Osmolality was VERY different by week, and that was not easily attributable to climate factors (see below). We also experienced technical difficulties with the osmometer between some weeks, so some of the variation may be attributable to technical error. Thus date/week will be a good random factor to include in the LMM.

distinguish pairwise differences using an ANOVA:

```
osml_date_aov <- aov(osmolality_mmol_kg ~ as.factor(date),
                     data = morpho_blood_SMI)
TukeyHSD(osml_date_aov)
```

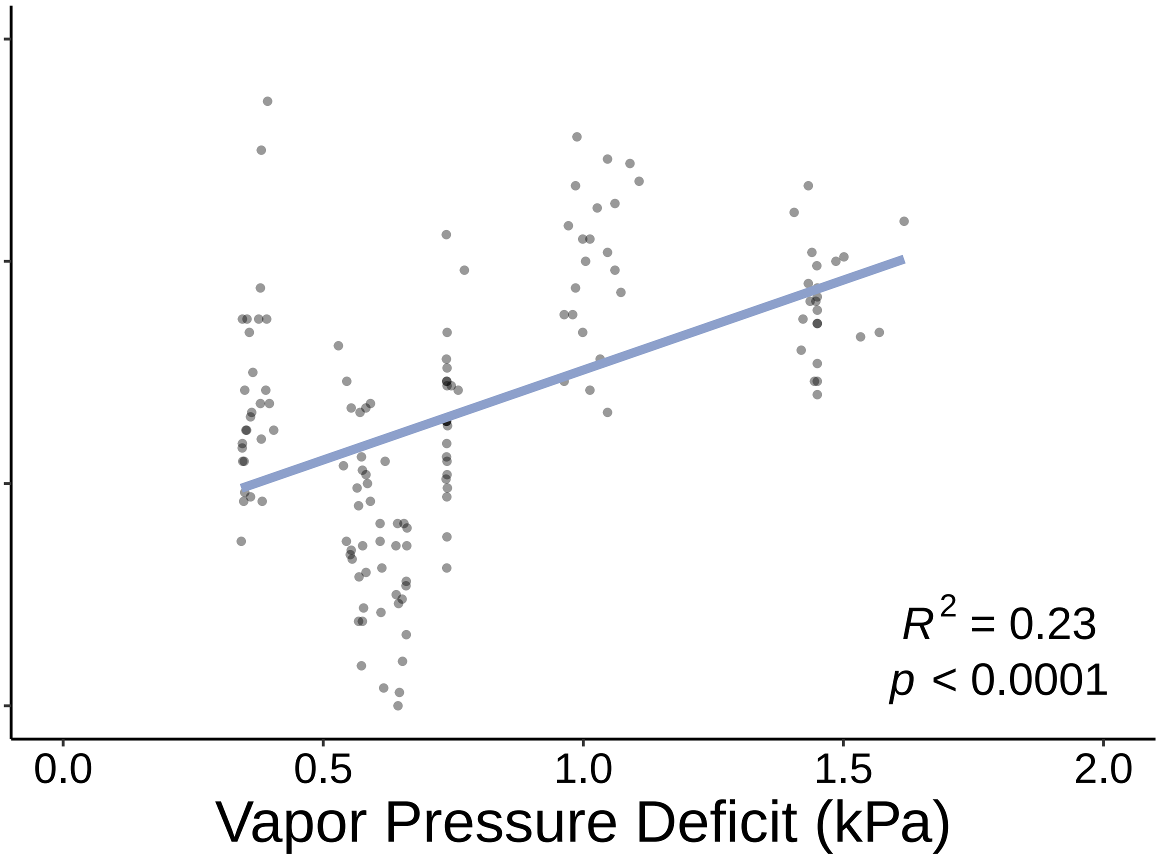
```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = osmolality_mmol_kg ~ as.factor(date), data = morpho_blood_SMI)
##
## $`as.factor(date)`
##                              diff        lwr        upr     p adj
## 2021-04-19-2021-04-05  22.891304   7.979417  37.803191 0.0002634
## 2021-04-26-2021-04-05  75.076923  60.628732  89.525114 0.0000000
## 2021-05-03-2021-04-05  66.657895  50.935165  82.380624 0.0000000
## 2021-05-10-2021-04-05  41.543478  26.631591  56.455365 0.0000000
## 2021-05-17-2021-04-05  46.785714  32.597872  60.973557 0.0000000
## 2021-04-26-2021-04-19  52.185619  37.273732  67.097506 0.0000000
## 2021-05-03-2021-04-19  43.766590  27.616718  59.916462 0.0000000
## 2021-05-10-2021-04-19  18.652174   3.290581  34.013766 0.0078361
## 2021-05-17-2021-04-19  23.894410   9.234635  38.554185 0.0000857
## 2021-05-03-2021-04-26  -8.419028 -24.141758   7.303701 0.6342961
## 2021-05-10-2021-04-26 -33.533445 -48.445332 -18.621558 0.0000000
## 2021-05-17-2021-04-26 -28.291209 -42.479051 -14.103367 0.0000008
## 2021-05-10-2021-05-03 -25.114416 -41.264289  -8.964544 0.0002088
## 2021-05-17-2021-05-03 -19.872180 -35.356006  -4.388355 0.0039942
## 2021-05-17-2021-05-10   5.242236  -9.417539  19.902011 0.9059063
```

Osml ~ VPD

```
my_blue <- RColorBrewer::brewer.pal(8, "Set2")[3]
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_point(aes(x = VPD_kPa_int,
                 y = osmolality_mmol_kg,
                 ),
             size = 1,
             alpha = 0.4) +
  stat_smooth(aes(x = VPD_kPa_int,
                  y = osmolality_mmol_kg),
              formula = y ~ x,
              method = "lm",
              se = F,
              color = my_blue,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Vapor Pressure Deficit (kPa)") +
  #ylab("Osmolality (mmol/kg)") +
  ylab("") +
  xlim(0, 2) +
  ylim(300, 450) +
  annotate("text", x = 1.8, y = 320,
           label = "paste(italic(R) ^ 2, \" = 0.23\")",
           parse = TRUE,
           size = 6) +
  annotate("text", x = 1.8, y = 305,
           label = "paste(italic(p), \" < 0.0001\")",
           parse = TRUE,
           size = 6) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 22),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 16),
        axis.text.y = element_blank(),
        legend.text.align = 0,
        plot.margin = unit(c(0.1,0,0.1,0.45), "cm"),
) -> osml_vpd_fig
osml_vpd_fig
```

```
## Warning: Removed 7 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 7 rows containing missing values (geom_point).
```



```
# export figure
#ggsave(filename = "osml_abhum_fig.jpeg",
 #      plot = osml_abhum_fig,
  #     path = "./final_figures",
   #    device = "jpeg",
    #   dpi = 1200,
     #  width = 6, height = 4)

# lm
lm_osml_vpd <- lm(osmolality_mmol_kg ~ VPD_kPa_int,
           data = morpho_blood_SMI)
summary(lm_osml_vpd)
```

```
##
## Call:
## lm(formula = osmolality_mmol_kg ~ VPD_kPa_int, data = morpho_blood_SMI)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -67.02 -18.29  -0.29  15.94  86.33
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  333.383      5.376  62.008  < 2e-16 ***
## VPD_kPa_int   41.434      6.256   6.623 6.87e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

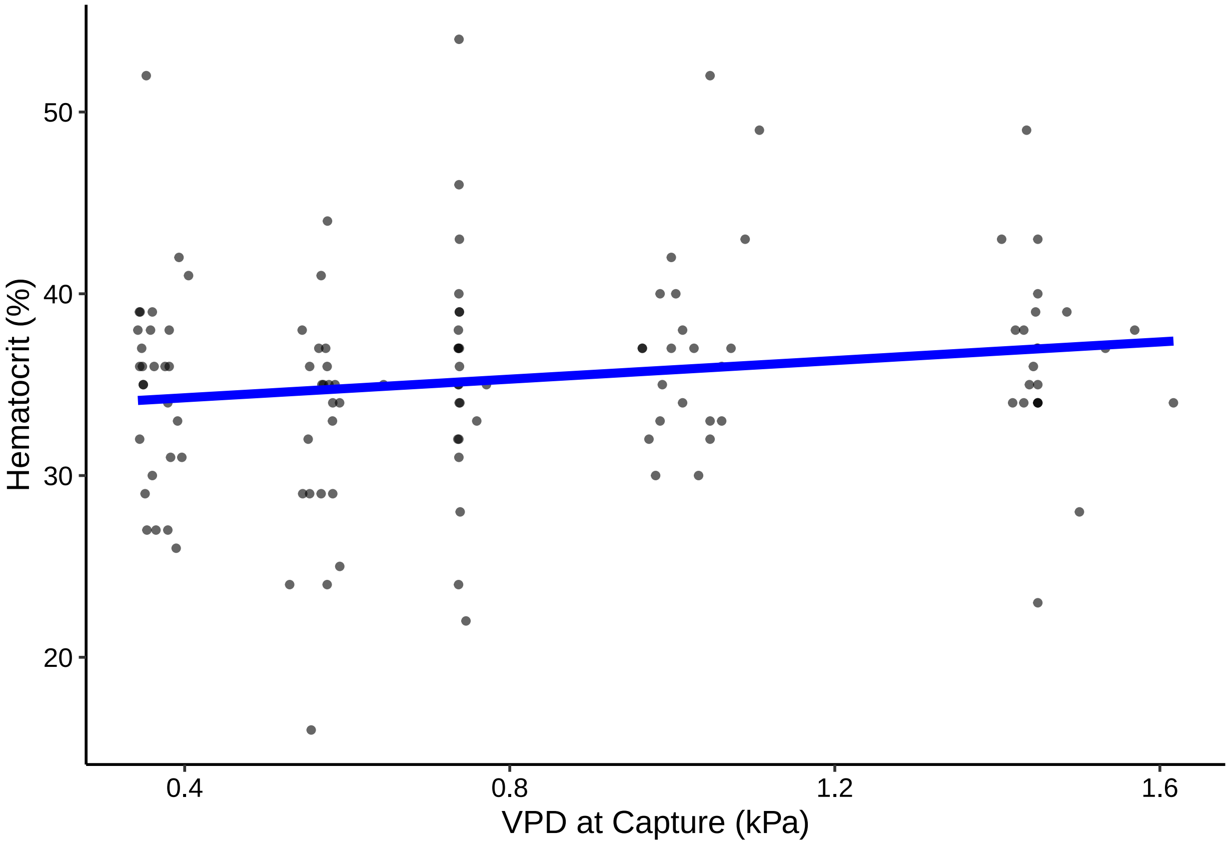
```
##
## Residual standard error: 26.91 on 141 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.2373, Adjusted R-squared:  0.2319
## F-statistic: 43.86 on 1 and 141 DF,  p-value: 6.87e-10
```

Hct ~ VPD

```
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_point(aes(x = VPD_kPa_int,
                 y = hematocrit_percent,
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = VPD_kPa_int,
                  y = hematocrit_percent),
              formula = y ~ x,
              method = "lm",
              se = F,
              color = "blue",
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("VPD at Capture (kPa)") +
  ylab("Hematocrit (%)") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 28 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 28 rows containing missing values (geom_point).
```



```
# lm
lm_hct_VPD <- lm(hematocrit_percent ~ VPD_kPa_int,
           data = morpho_blood_SMI)
summary(lm_hct_VPD)
```

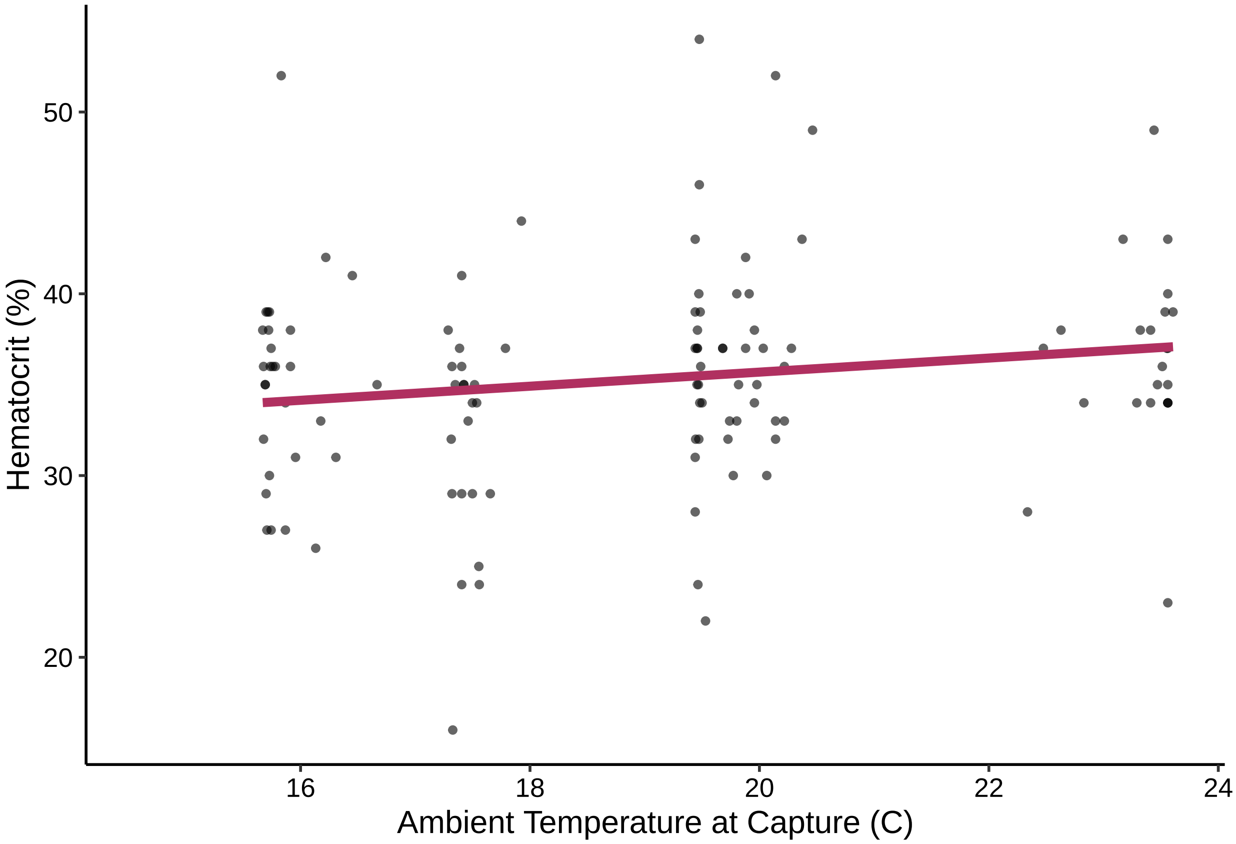
```
##
## Call:
## lm(formula = hematocrit_percent ~ VPD_kPa_int, data = morpho_blood_SMI)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.6767  -2.9680   0.1716   2.4389  18.8573
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   33.253      1.247   26.66   <2e-16 ***
## VPD_kPa_int    2.562      1.392    1.84   0.0682 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.887 on 118 degrees of freedom
##   (28 observations deleted due to missingness)
## Multiple R-squared:  0.0279, Adjusted R-squared:  0.01967
## F-statistic: 3.387 on 1 and 118 DF,  p-value: 0.06822
```

Hct ~ Temperature

```r
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_point(aes(x = temp_C_interpol,
                 y = hematocrit_percent,
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = temp_C_interpol,
                  y = hematocrit_percent),
              formula = y ~ x,
              method = "lm",
              se = F,
              color = "maroon",
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Ambient Temperature at Capture (C)") +
  ylab("Hematocrit (%)") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 28 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 28 rows containing missing values (geom_point).
```



```
# lm
lm_hct_temp <- lm(hematocrit_percent ~ temp_C_interpol,
           data = morpho_blood_SMI)
summary(lm_hct_temp)
```

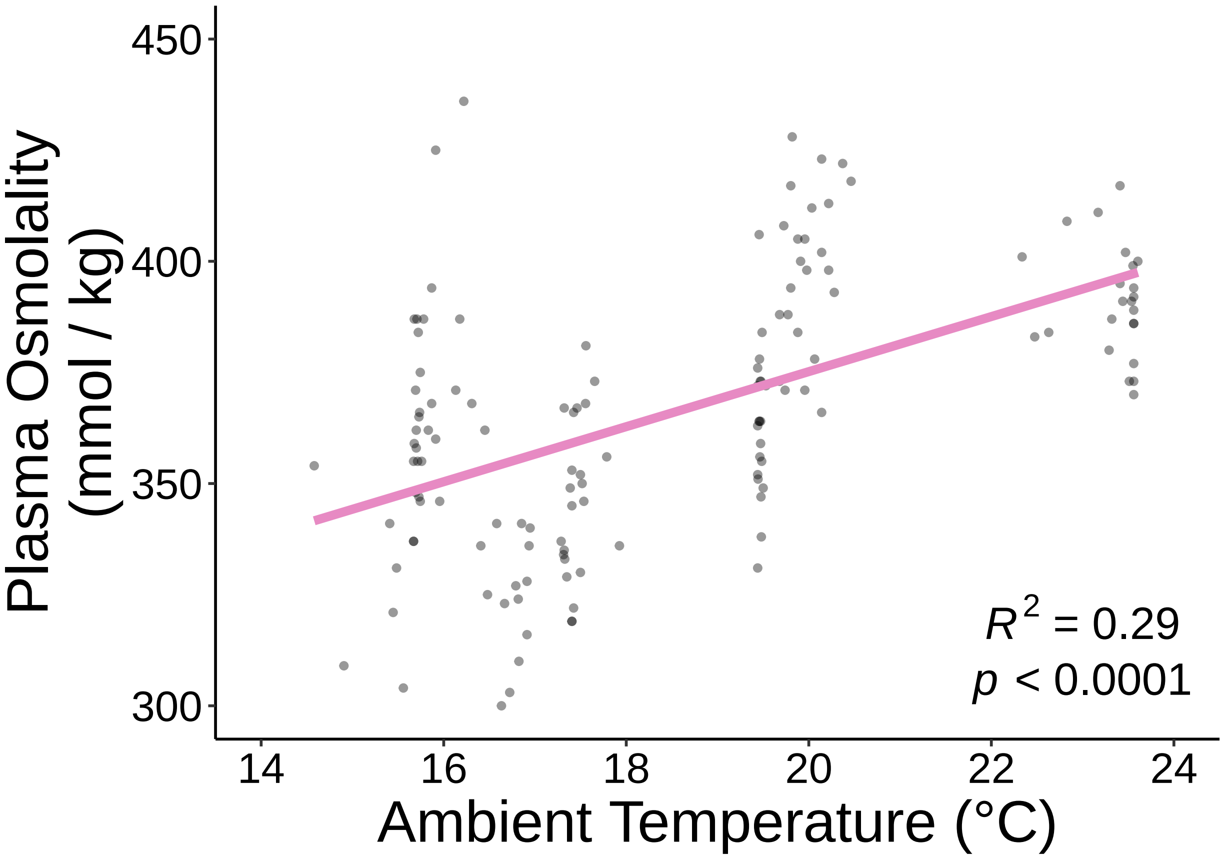
```
##
## Call:
## lm(formula = hematocrit_percent ~ temp_C_interpol, data = morpho_blood_SMI)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.6541  -3.0678   0.3085   2.3757  18.5137
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      27.9447     3.9993   6.987 1.78e-10 ***
## temp_C_interpol   0.3872     0.2079   1.862   0.0651 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.885 on 118 degrees of freedom
##   (28 observations deleted due to missingness)
## Multiple R-squared:  0.02855,    Adjusted R-squared:  0.02032
## F-statistic: 3.468 on 1 and 118 DF,  p-value: 0.06505
```

Osml ~ Temperature

```r
my_red <- RColorBrewer::brewer.pal(8, "Set2")[4]

morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_point(aes(x = temp_C_interpol,
                 y = osmolality_mmol_kg,
                 ),
             size = 1,
             alpha = 0.4) +
  stat_smooth(aes(x = temp_C_interpol,
                  y = osmolality_mmol_kg),
              formula = y ~ x,
              method = "lm",
              se = F,
              color = my_red,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Ambient Temperature (°C)") +
  ylab("Plasma Osmolality\n(mmol / kg)") +
  annotate("text", x = 23, y = 320,
           label = "paste(italic(R) ^ 2, \" = 0.29\")",
           parse = TRUE,
           size = 6) +
  annotate("text", x = 23, y = 305,
           label = "paste(italic(p), \" < 0.0001\")",
           parse = TRUE,
           size = 6) +
  xlim(14, 24) +
  ylim(300,450) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 22),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 16),
        legend.text.align = 0,
        plot.margin = unit(c(0.1,0,0.1,0.45), "cm"),
) -> osml_temp_fig
osml_temp_fig
```

```
## Warning: Removed 7 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 7 rows containing missing values (geom_point).
```



```
# export figure
#ggsave(filename = "osml_temp_fig.jpeg",
 #      plot = osml_temp_fig,
  #     path = "./final_figures",
   #    device = "jpeg",
    #   dpi = 1200,
     #  width = 6, height = 4)

# lm
lm_osml_temp <- lm(osmolality_mmol_kg ~ temp_C_interpol,
           data = morpho_blood_SMI)
summary(lm_osml_temp)
```

```
##
## Call:
## lm(formula = osmolality_mmol_kg ~ temp_C_interpol, data = morpho_blood_SMI)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -59.723 -16.089  -1.996  15.667  85.643
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     246.0362    15.6239  15.747  < 2e-16 ***
## temp_C_interpol   6.4316     0.8315   7.735  1.8e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

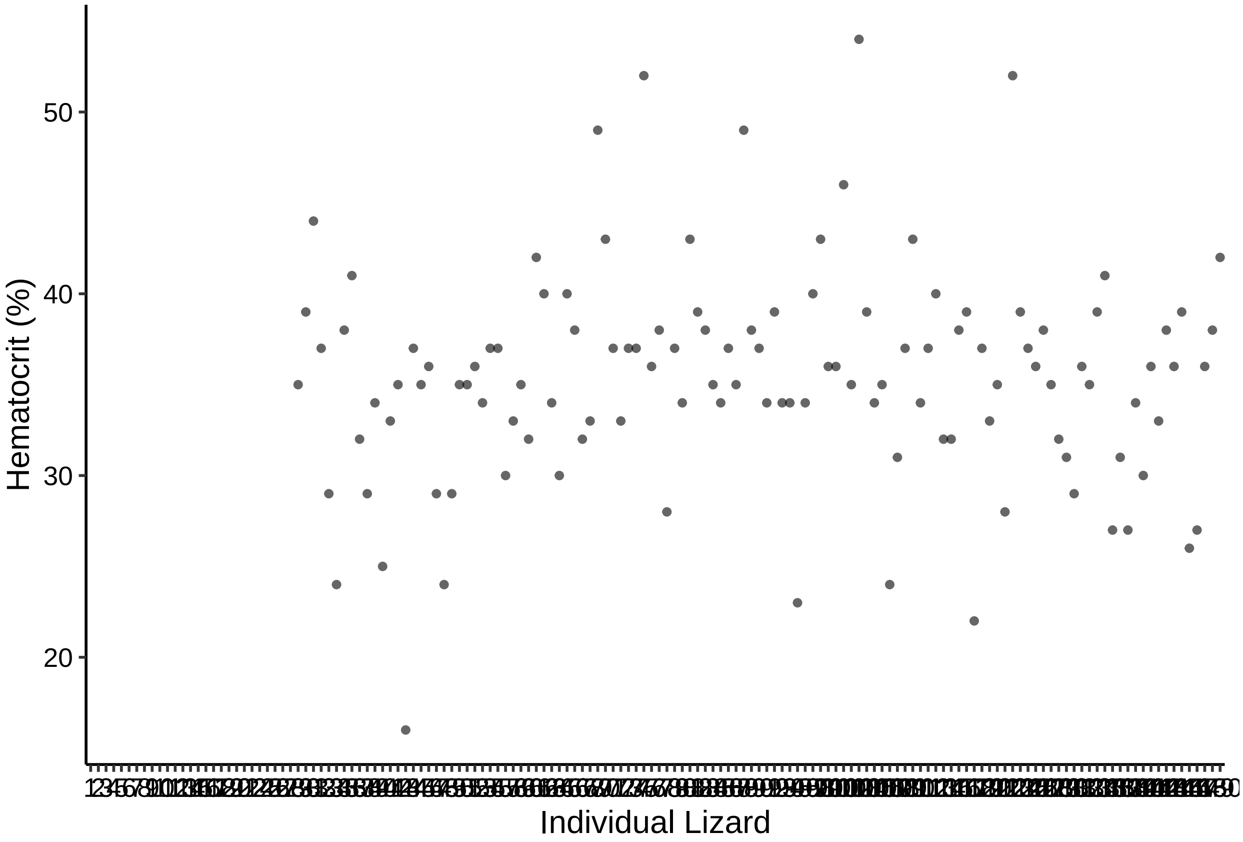
```
##
## Residual standard error: 25.82 on 141 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.2979, Adjusted R-squared:  0.2929
## F-statistic: 59.82 on 1 and 141 DF,  p-value: 1.796e-12
```

Hct ~ Individual

```
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_point(aes(x = individual_ID,
                 y = hematocrit_percent,
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = individual_ID,
                  y = hematocrit_percent,
                  ),
              formula = y ~ x,
              method = "lm",
              color = "gray",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Individual Lizard") +
  ylab("Hematocrit (%)") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 27 rows containing non-finite values (stat_smooth).
```

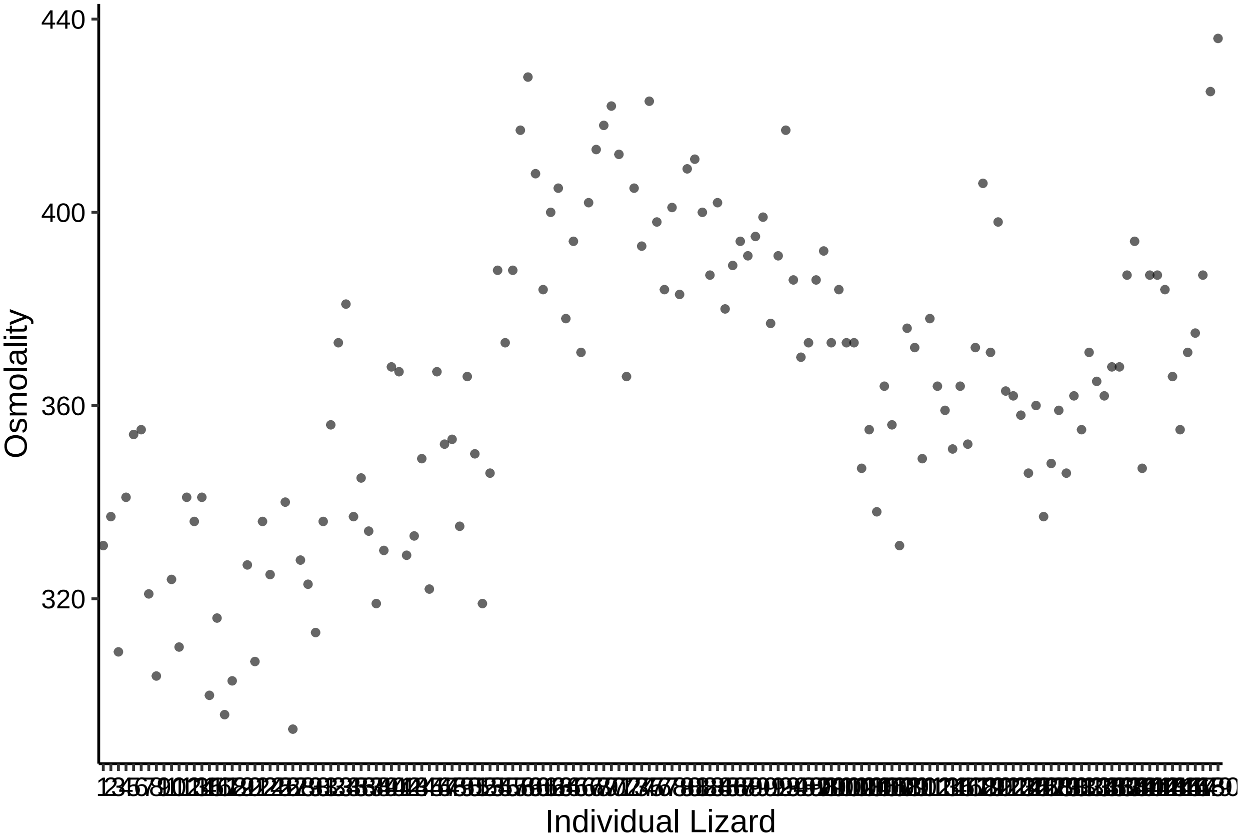
```
## Warning: Removed 27 rows containing missing values (geom_point).
```



Osml ~ Individual

```
morpho_blood_SMI %>%
  ggplot(data = .) +
  geom_point(aes(x = individual_ID,
                 y = osmolality_mmol_kg,
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = individual_ID,
                  y = osmolality_mmol_kg,
                  ),
              formula = y ~ x,
              method = "lm",
              color = "gray",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Individual Lizard") +
  ylab("Osmolality") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 3 rows containing non-finite values (stat_smooth).

## Warning: Removed 3 rows containing missing values (geom_point).
```



Conclusion

Osmolality seems to be affected by: - SMI (large effect, although not statistically significant) - sex (NS) - whether or not the sample was hemolyzed *() ! - week/date of sampling!! (*) **- capture temp & VPD (both** *)

So, for the LMM to predict osmolality, we will start with SMI, hemolyzed/not, week/date, and capture temp and VPD as our predictor variables in the model.

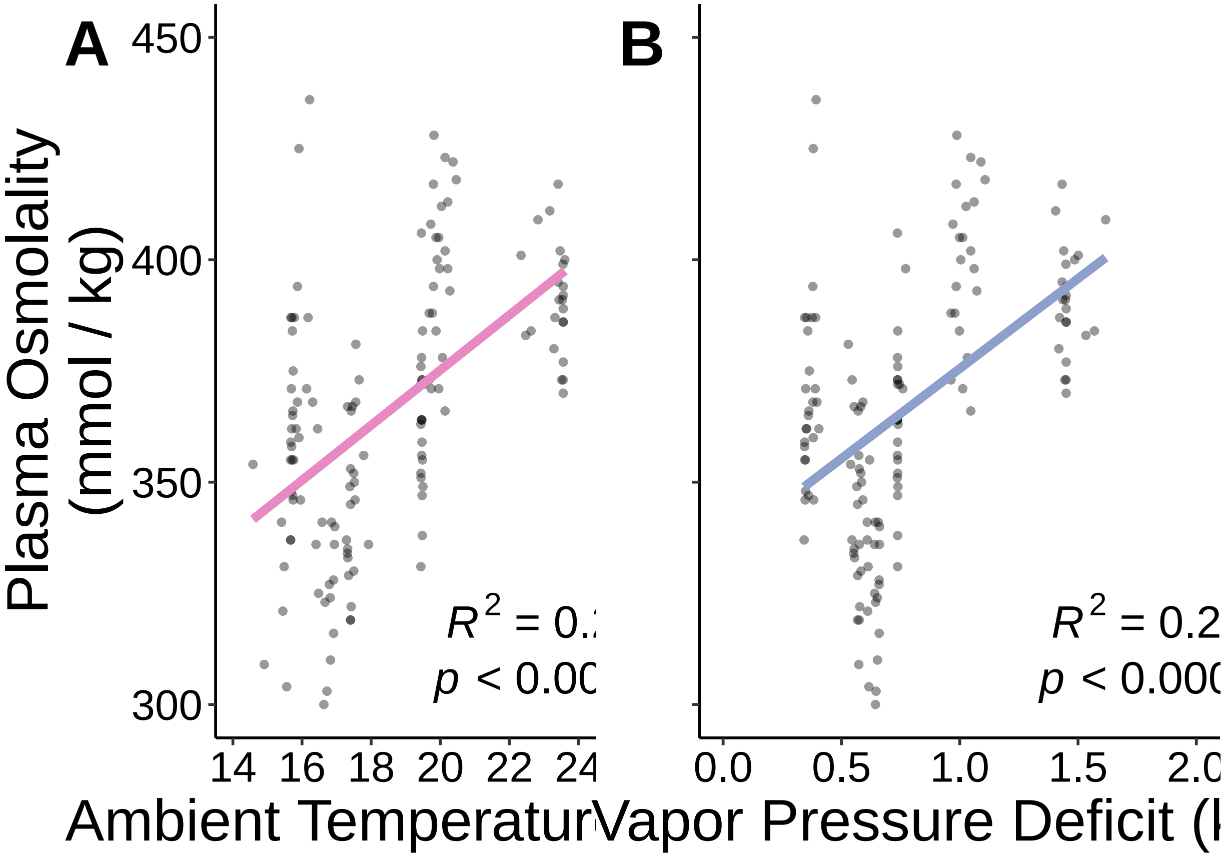
Hematocrit was only meaningfully different based on sex, so that’s the only statistic we’ll include in the paper for what variables affect hematocrit.

Osmolality Multi-Figure

```
ggarrange(osml_temp_fig, osml_vpd_fig,
          ncol = 2, nrow = 1,
          labels = c("A", "B", "C", "D", "E", "F"),
          hjust = c(-2, -0.5),
          font.label = list(size = 24, face = "bold", color ="black")
          ) -> osml_multi_fig
```

```
## Warning: Removed 7 rows containing non-finite values (stat_smooth).

## Warning: Removed 7 rows containing missing values (geom_point).

## Warning: Removed 7 rows containing non-finite values (stat_smooth).

## Warning: Removed 7 rows containing missing values (geom_point).
```

```
osml_multi_fig
```



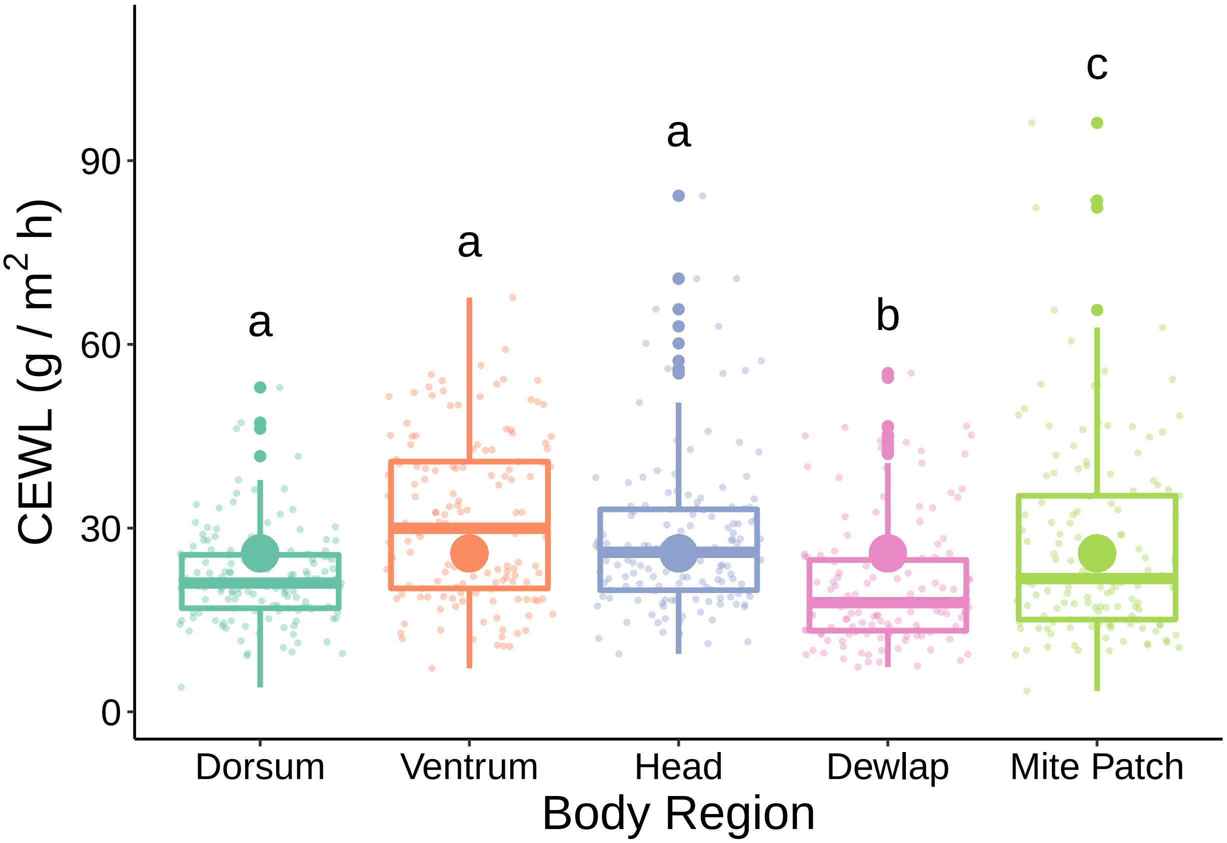
```
# export figure
ggsave(filename = "osml_multi_fig.jpeg",
       plot = osml_multi_fig,
       path = "./final_figures",
       device = "jpeg",
       dpi = 1200,
       width = 12, height = 4)
```

What affects evaporative water loss?

Potential relationships: - CEWL ~ date/week, individual, SVL, SMI, gravidity, hct, osml, cloacal temp, ambient temp, ambient RH, measurement temp, measurement RH, **body region**

CEWL ~ Body Region

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = (region),
                   y = (TEWL_g_m2h),
                   color = region
                   ),
               size = 1,
               alpha = 1) +
  geom_jitter(aes(x = (region),
                   y = TEWL_g_m2h,
                   color = region
                   ),
               size = 0.6,
               alpha = 0.4) +
  geom_point(aes(x = (region),
                   y = mean(TEWL_g_m2h),
                   color = region,
                   ),
               size = 6,
               alpha = 1) +
  theme_classic() +
  xlab("Body Region") +
  ylab(bquote('CEWL (g / '*m^2~h*')')) +
  annotate("text", x = 1, y = 64, label = "a", size = 6) +
  annotate("text", x = 2, y = 77, label = "a", size = 6) +
  annotate("text", x = 3, y = 95, label = "a", size = 6) +
  annotate("text", x = 4, y = 65, label = "b", size = 6) +
  annotate("text", x = 5, y = 106, label = "c", size = 6) +
  scale_color_brewer(palette = "Set2") +
  ylim(1, 110) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 18),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 14),
        legend.text.align = 0,
        legend.position = "none"
) -> CEWL_region_fig
CEWL_region_fig
```



```
# export figure
ggsave(filename = "CEWL_region_fig.jpeg",
       plot = CEWL_region_fig,
       path = "./final_figures",
       device = "jpeg",
       dpi = 1200,
       width = 6, height = 4)
```

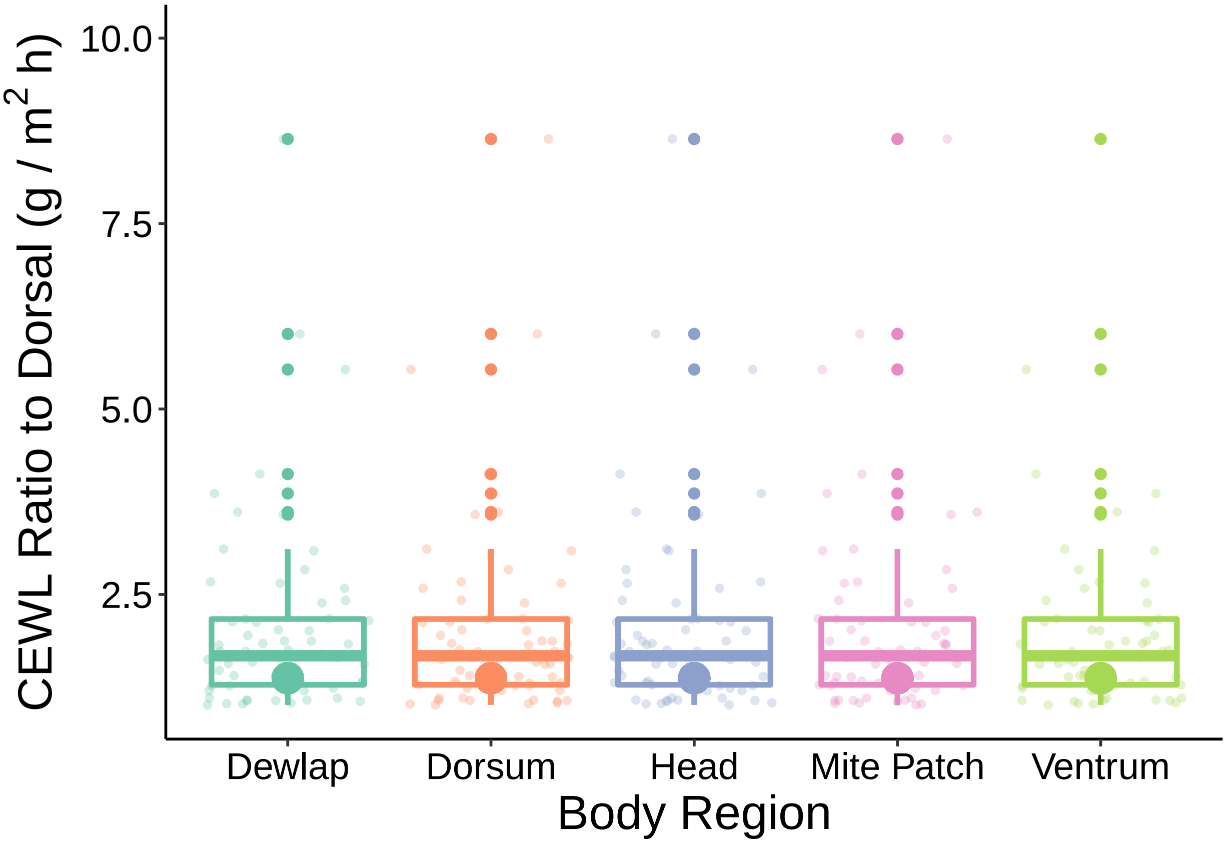
Body Region Ratios

```
CEWL_mite <- CEWL_data_full %>%
  dplyr::filter(region == "Mite Patch") %>%
  dplyr::select(mite_CEWL = TEWL_g_m2h, individual_ID)
CEWL_dewl <- CEWL_data_full %>%
  dplyr::filter(region == "Dewlap")%>%
  dplyr::select(dewl_CEWL = TEWL_g_m2h, individual_ID)
CEWL_head <- CEWL_data_full %>%
  dplyr::filter(region == "Head") %>%
  dplyr::select(head_CEWL = TEWL_g_m2h, individual_ID)
CEWL_vent <- CEWL_data_full %>%
  dplyr::filter(region == "Ventrum") %>%
  dplyr::select(vent_CEWL = TEWL_g_m2h, individual_ID)
CEWL_dors <- CEWL_data_full %>%
  dplyr::filter(region == "Dorsum") %>%
  dplyr::select(dors_CEWL = TEWL_g_m2h, individual_ID)

by_region_CEWL <- CEWL_mite %>%
  left_join(CEWL_dewl, by = "individual_ID") %>%
  left_join(CEWL_head, by = "individual_ID") %>%
  left_join(CEWL_vent, by = "individual_ID") %>%
  left_join(CEWL_dors, by = "individual_ID") %>%
  mutate(mite_dors = mite_CEWL/dors_CEWL,
         dewl_dors = mite_CEWL/dors_CEWL,
         head_dors = mite_CEWL/dors_CEWL,
         vent_dors = mite_CEWL/dors_CEWL,
         dors_dors = mite_CEWL/dors_CEWL,
         )

ratio_mite <- by_region_CEWL %>%
  dplyr::select(individual_ID, ratio_to_dors = mite_dors) %>%
  mutate(region = "Mite Patch")
ratio_dewl <- by_region_CEWL %>%
  dplyr::select(individual_ID, ratio_to_dors = dewl_dors) %>%
  mutate(region = "Dewlap")
ratio_head <- by_region_CEWL %>%
  dplyr::select(individual_ID, ratio_to_dors = head_dors) %>%
  mutate(region = "Head")
ratio_vent <- by_region_CEWL %>%
  dplyr::select(individual_ID, ratio_to_dors = vent_dors) %>%
  mutate(region = "Ventrum")
ratio_dors <- by_region_CEWL %>%
  dplyr::select(individual_ID, ratio_to_dors = dors_dors) %>%
  mutate(region = "Dorsum")

vert_ratios <- ratio_mite %>%
  rbind(ratio_dewl) %>%
  rbind(ratio_head) %>%
  rbind(ratio_vent) %>%
  rbind(ratio_dors)
```

```
vert_ratios %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = (region),
                   y = ratio_to_dors,
                   color = region
                   ),
               size = 1,
               alpha = 1) +
  geom_jitter(aes(x = (region),
                   y = ratio_to_dors,
                   color = region
                   ),
               size = 1,
               alpha = 0.3) +
  geom_point(aes(x = (region),
                   y = mean(ratio_to_dors),
                   color = region
                   ),
               size = 5,
               alpha = 1) +
  theme_classic() +
```

```r
  xlab("Body Region") +
  ylab(bquote('CEWL Ratio to Dorsal (g / '*m^2~h*')')) +
  annotate("text", x = 1, y = 64, label = "a", size = 6) +
  annotate("text", x = 2, y = 77, label = "b", size = 6) +
  annotate("text", x = 3, y = 95, label = "b", size = 6) +
  annotate("text", x = 4, y = 65, label = "a", size = 6) +
  annotate("text", x = 5, y = 106, label = "b", size = 6) +
  scale_color_brewer(palette = "Set2") +
  ylim(1, 10) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 18),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 14),
        legend.text.align = 0,
        legend.position = "none"
) -> CEWL_region_fig3
CEWL_region_fig3
```

```
## Warning: Removed 360 rows containing non-finite values (stat_boxplot).

## Warning: Removed 360 rows containing missing values (geom_point).

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

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

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

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

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



```
# export figure
ggsave(filename = "CEWL_region_fig3.jpeg",
       plot = CEWL_region_fig3,
       path = "./final_figures",
       device = "jpeg",
       dpi = 1200,
       width = 6, height = 4)
```

```
## Warning: Removed 360 rows containing non-finite values (stat_boxplot).

## Warning: Removed 360 rows containing missing values (geom_point).

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

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

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

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

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

```
CEWL_lm1 <- lm(TEWL_g_m2h ~ region, data = CEWL_data_full)
summary(CEWL_lm1)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region, data = CEWL_data_full)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -24.153  -8.364  -2.366   5.668  68.934 
## 
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)    
## (Intercept)        21.903      1.044  20.986  < 2e-16 ***
## regionVentrum       9.361      1.476   6.342 4.09e-10 ***
## regionHead          6.391      1.476   4.330 1.71e-05 ***
## regionDewlap       -1.255      1.481  -0.847 0.397223    
## regionMite Patch    5.324      1.487   3.581 0.000367 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 12.39 on 694 degrees of freedom
## Multiple R-squared:  0.09502,    Adjusted R-squared:  0.0898 
## F-statistic: 18.22 on 4 and 694 DF,  p-value: 3.059e-14
```

Also get pairwise statistics:

```
# one-way ANOVA
CEWL_region_aov <- aov(data = CEWL_data_full,
                       TEWL_g_m2h ~ region)
# post-hoc pairwise analysis
TukeyHSD(CEWL_region_aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
## 
## Fit: aov(formula = TEWL_g_m2h ~ region, data = CEWL_data_full)
## 
## $region
##                          diff        lwr         upr     p adj
## Ventrum-Dorsum       9.360638   5.323763 13.39751386 0.0000000
## Head-Dorsum          6.390887   2.354011 10.42776209 0.0001662
## Dewlap-Dorsum       -1.254852  -5.306223  2.79651839 0.9156911
## Mite Patch-Dorsum    5.323511   1.257276  9.38974646 0.0033630
## Head-Ventrum        -2.969752  -7.006627  1.06712379 0.2612293
## Dewlap-Ventrum     -10.615491 -14.666861 -6.56411991 0.0000000
## Mite Patch-Ventrum  -4.037127  -8.103362  0.02910817 0.0527256
## Dewlap-Head         -7.645739 -11.697109 -3.59436813 0.0000032
## Mite Patch-Head     -1.067375  -5.133610  2.99885994 0.9523881
## Mite Patch-Dewlap    6.578364   2.497738 10.65898957 0.0001172
```

CEWL Heat Map

get means:

```
mean_CEWL_region <- CEWL_data_full %>%
  group_by(region) %>%
  summarise(CEWL = mean(TEWL_g_m2h)) %>%
  arrange(CEWL)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
mean_CEWL_region
```

```
## # A tibble: 5 x 2
##   region      CEWL
##   <fct>      <dbl>
## 1 Dewlap      20.6
## 2 Dorsum      21.9
## 3 Mite Patch  27.2
## 4 Head        28.3
## 5 Ventrum     31.3
```

get colors:

```
my_CEWL_colors <- RColorBrewer::brewer.pal(9, "Blues")[1:9]
```

assign colors:

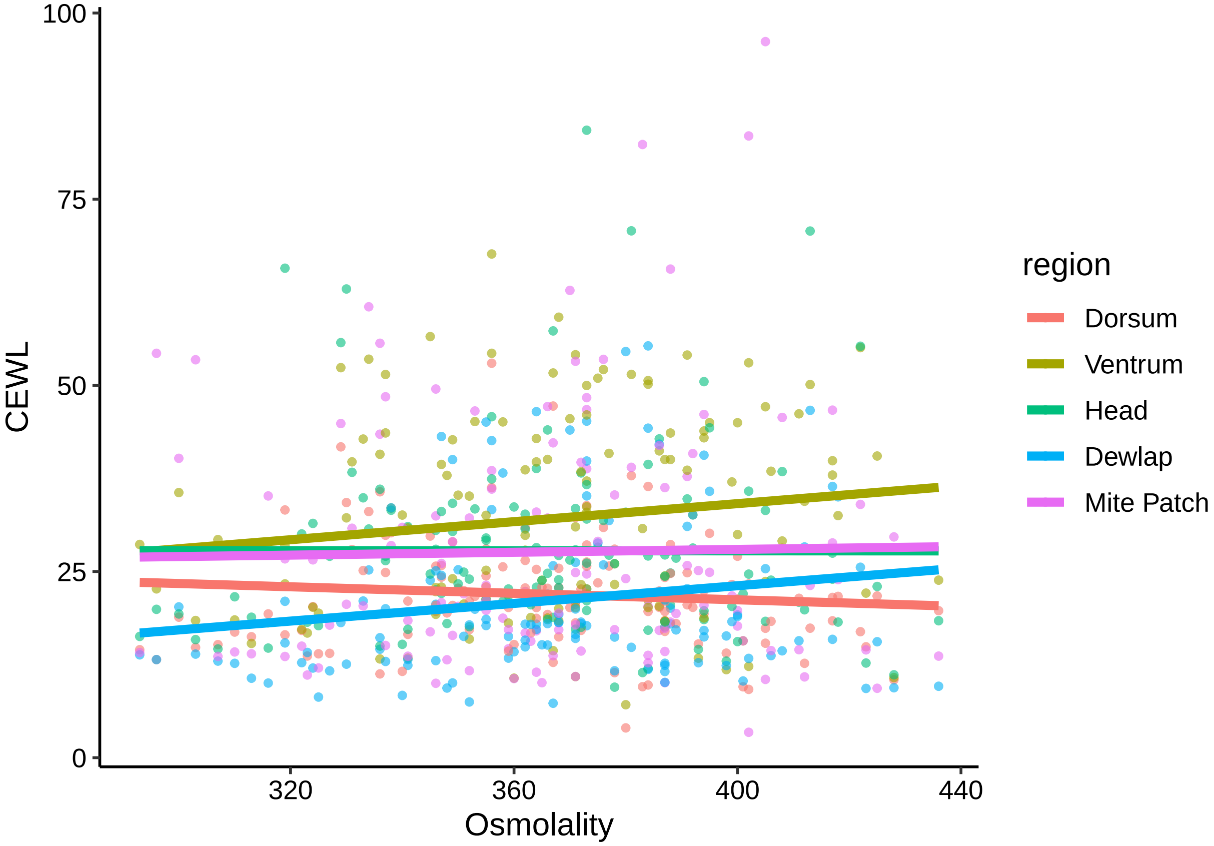
```
CEWL_heatmap_colors <- data.frame(region = c("Dorsum",
                                             "Ventrum",
                                             "Head",
                                             "Dewlap",
                                             "Mite Patch"),
                                  color = c(my_CEWL_colors[1], #dorsum
                                            my_CEWL_colors[2], #ventrum
                                            my_CEWL_colors[7], #head
                                            my_CEWL_colors[8], #dewlap
                                            my_CEWL_colors[9])) #mite patch

#write.csv(CEWL_heatmap_colors, "./CEWL_heatmap_colors.csv")
```

CEWL ~ Osmolality

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = osmolality_mmol_kg,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = osmolality_mmol_kg,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Osmolality") +
  ylab("CEWL") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                        size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 49 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 49 rows containing missing values (geom_point).
```

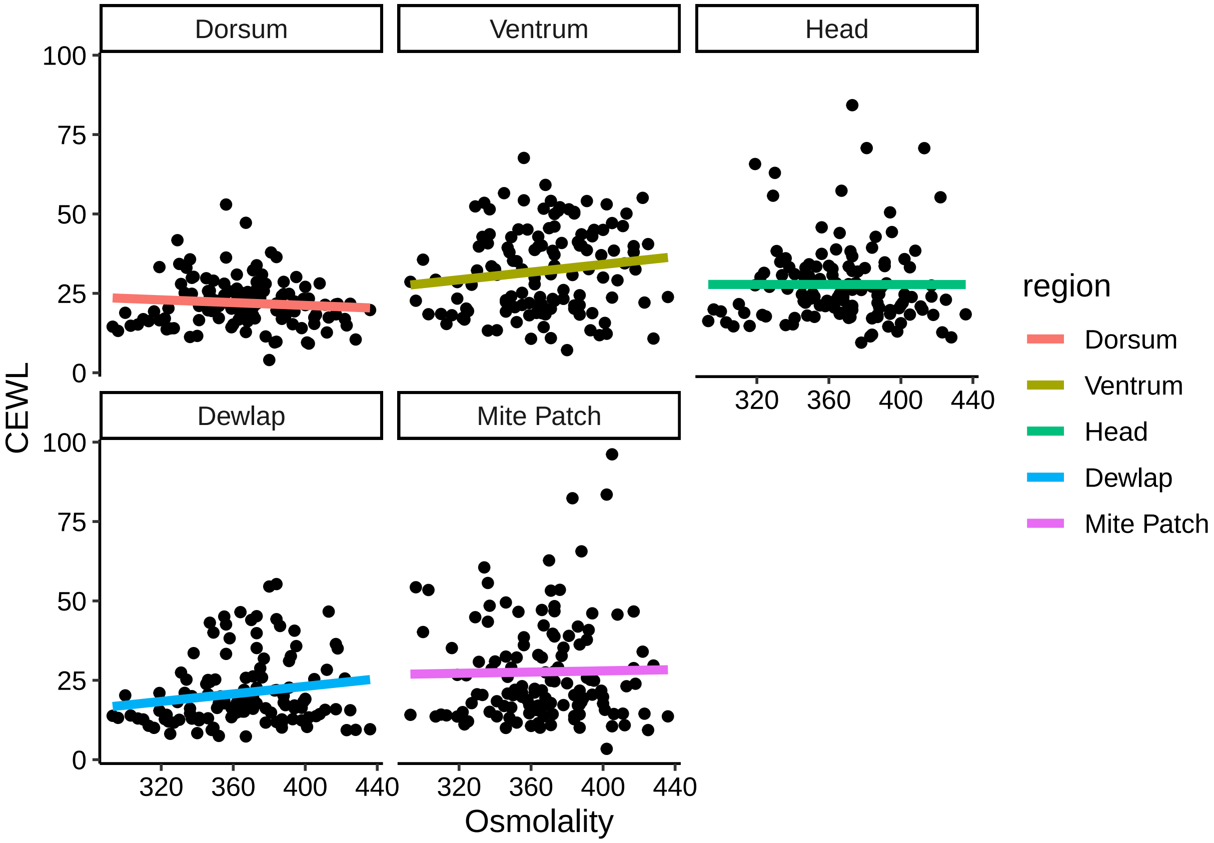


```
# lm
CEWL_lm2 <- lm(TEWL_g_m2h ~ region + osmolality_mmol_kg,
           data = CEWL_data_full)
summary(CEWL_lm2)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region + osmolality_mmol_kg, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -25.283  -8.587  -2.447   5.805  67.689
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         14.06871    5.97323   2.355 0.018807 *
## regionVentrum       10.18895    1.53611   6.633 6.98e-11 ***
## regionHead           5.86072    1.53611   3.815 0.000149 ***
## regionDewlap        -0.80956    1.54206  -0.525 0.599774
## regionMite Patch     5.73228    1.54510   3.710 0.000225 ***
## osmolality_mmol_kg  0.02141    0.01601   1.337 0.181597
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.43 on 644 degrees of freedom
##   (49 observations deleted due to missingness)
## Multiple R-squared:  0.1011, Adjusted R-squared:  0.0941
## F-statistic: 14.48 on 5 and 644 DF,  p-value: 1.836e-13
```

```
# Facet ggplot
ggplot(aes(osmolality_mmol_kg, TEWL_g_m2h), data = CEWL_data_full) +
  geom_point() +
  stat_smooth(aes(x = osmolality_mmol_kg,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 )+
    theme_classic() +
  xlab("Osmolality") +
  ylab("CEWL") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)+
  facet_wrap(~ region) # create a facet for each body region
```

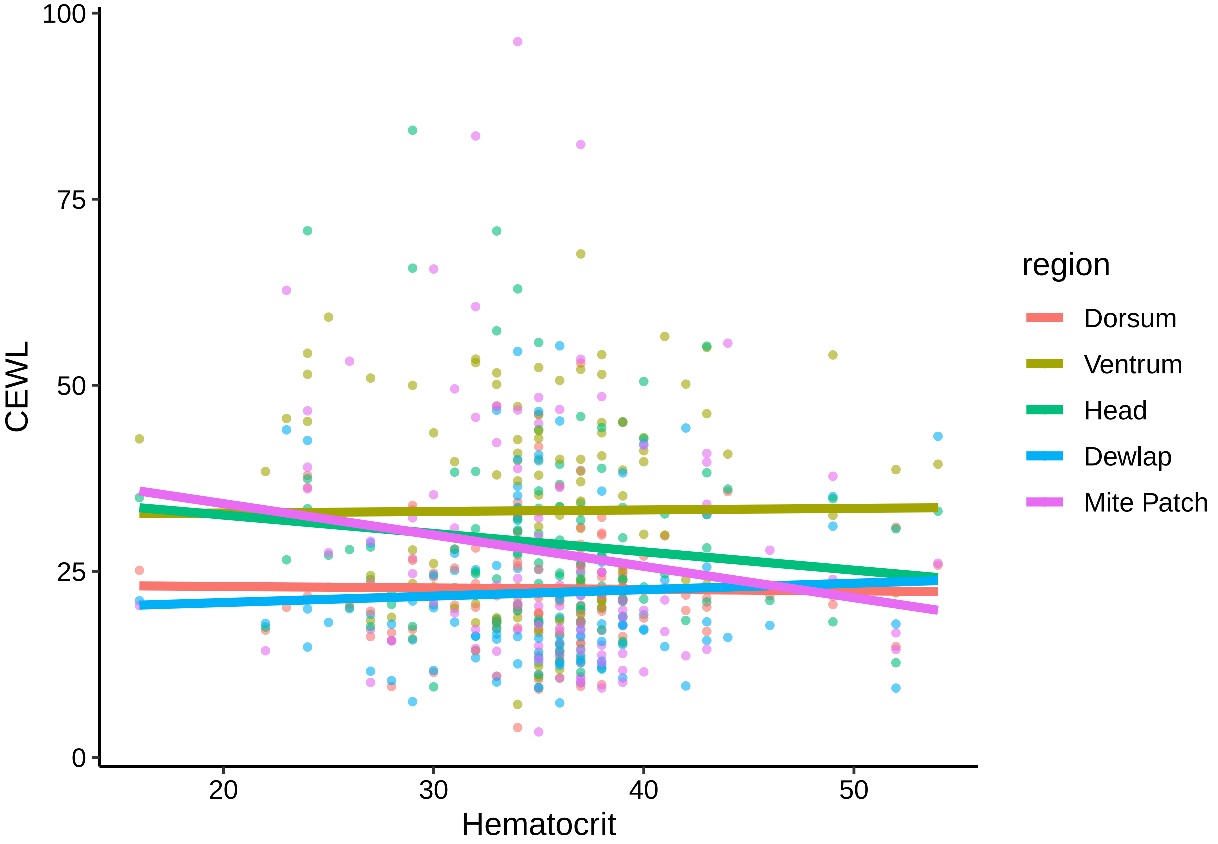
```
## Warning: Removed 49 rows containing non-finite values (stat_smooth).

## Warning: Removed 49 rows containing missing values (geom_point).
```



CEWL ~ Hematocrit

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = hematocrit_percent,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = hematocrit_percent,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Hematocrit") +
  ylab("CEWL") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                  family = "sans",
                                  size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 119 rows containing non-finite values (stat_smooth).

## Warning: Removed 119 rows containing missing values (geom_point).
```



```
# lm
CEWL_lm3 <- lm(TEWL_g_m2h ~ region * hematocrit_percent,
           data = CEWL_data_full)
summary(CEWL_lm3)
```

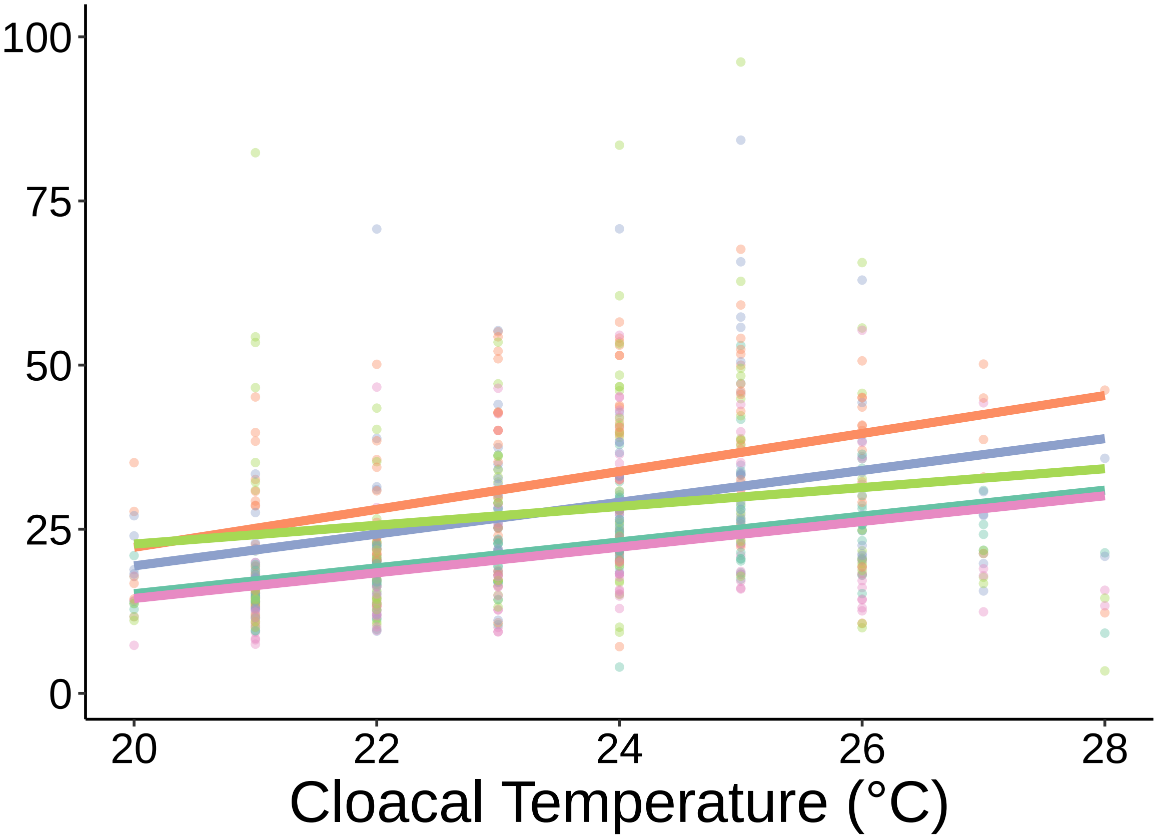
```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region * hematocrit_percent, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -26.008  -8.565  -2.311   5.873  67.969
##
## Coefficients:
##                                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           23.34895    7.01415   3.329 0.000928 ***
## regionVentrum                          9.06299    9.91223   0.914 0.360933
## regionHead                            14.13898    9.91545   1.426 0.154429
## regionDewlap                          -4.28061    9.92716  -0.431 0.666485
## regionMite Patch                    19.17144    9.93845   1.929 0.054225 .
## hematocrit_percent                  -0.01962    0.19547  -0.100 0.920093
## regionVentrum:hematocrit_percent     0.04039    0.27613   0.146 0.883764
## regionHead:hematocrit_percent       -0.22718    0.27612  -0.823 0.411007
## regionDewlap:hematocrit_percent      0.10618    0.27640   0.384 0.701002
## regionMite Patch:hematocrit_percent -0.40184    0.27724  -1.449 0.147759
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.6 on 570 degrees of freedom
##   (119 observations deleted due to missingness)
## Multiple R-squared:  0.1063, Adjusted R-squared:  0.09216
## F-statistic: 7.531 on 9 and 570 DF,  p-value: 1.86e-10
```

CEWL ~ Cloacal Temperature

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = cloacal_temp_C,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.4) +
  stat_smooth(aes(x = cloacal_temp_C,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1) +
  theme_classic() +
  xlab("Cloacal Temperature (°C)") +
  ylab("") +
  #ylab(bquote('CEWL (g/'*m^2*'h)')) +
  #annotate("text", x = 1, y = 65, label = "a", size = 6) +
  scale_color_brewer(palette = "Set2",
                     name = "") +
  ylim(1, 100) +
  xlim(20, 28) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 22),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 16),
        legend.text = element_text(color = "black",
                                   family = "sans",
                                   size = 26),
        plot.margin = unit(c(0.1,0,0.35,0.45), "cm"),
        legend.text.align = 0,
        legend.position = "none"
        #legend.position = c(0.9, 0.85)
  #) +
  #guides(color = guide_legend(nrow = 2, byrow = TRUE)
         ) -> CEWL_ctemp_fig
CEWL_ctemp_fig
```

```
## Warning: Removed 49 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 49 rows containing missing values (geom_point).
```



```
# export figure
#ggsave(filename = "CEWL_ctemp_fig.tiff",
 #      plot = CEWL_ctemp_fig,
  #     path = "./final_figures",
   #    device = "tiff",
    #   dpi = 1200,
     #  width = 6, height = 4)
```

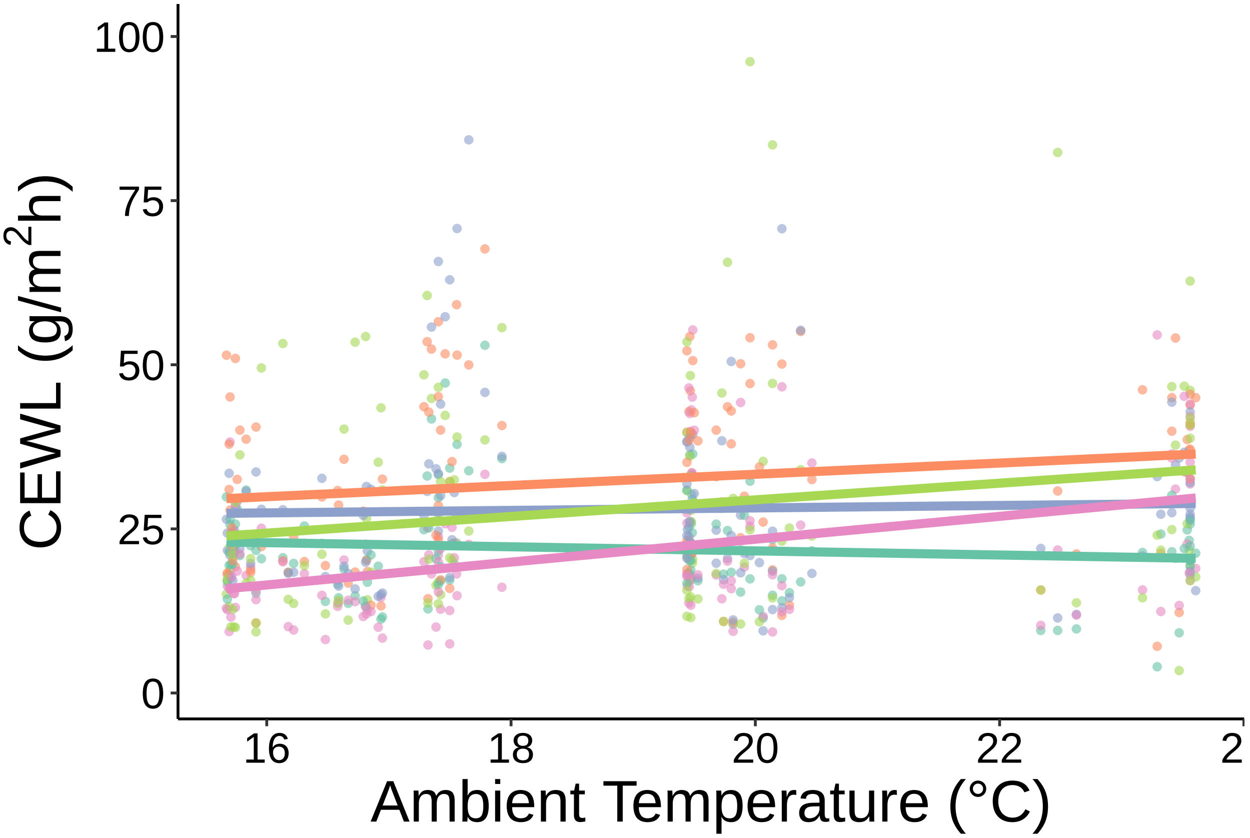
```
CEWL_lm4 <- lm(TEWL_g_m2h ~ region + cloacal_temp_C,
           data = CEWL_data_full)
summary(CEWL_lm4)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region + cloacal_temp_C, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -34.079  -7.557  -1.903   5.192  65.103 
## 
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)    
## (Intercept)       -28.2950     5.8771  -4.814 1.84e-06 ***
## regionVentrum      10.1968     1.4573   6.997 6.58e-12 ***
## regionHead          5.7798     1.4573   3.966 8.13e-05 ***
## regionDewlap       -0.7740     1.4630  -0.529    0.597    
## regionMite Patch    5.7531     1.4659   3.925 9.62e-05 ***
## cloacal_temp_C      2.1439     0.2471   8.675  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 11.79 on 644 degrees of freedom
##   (49 observations deleted due to missingness)
## Multiple R-squared:  0.1925, Adjusted R-squared:  0.1862 
## F-statistic: 30.71 on 5 and 644 DF,  p-value: < 2.2e-16
```

CEWL ~ Capture Temperature

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = temp_C_interpol,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = temp_C_interpol,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  scale_color_brewer(palette = "Set2") +
  theme_classic() +
  xlab("Ambient Temperature (°C)") +
  #ylab("") +
  ylim(1, 100) +
  ylab(bquote('CEWL (g/'*m^2*'h)')) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 22),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 16),
        legend.text = element_text(color = "black",
                                 family = "sans",
                                 size = 26),
        plot.margin = unit(c(0.1,0,0.35,0.1), "cm"),
        legend.text.align = 0,
        legend.position = "none"
) -> cap_temp_CEWL
cap_temp_CEWL
```

```
## Warning: Removed 59 rows containing non-finite values (stat_smooth).

## Warning: Removed 59 rows containing missing values (geom_point).
```



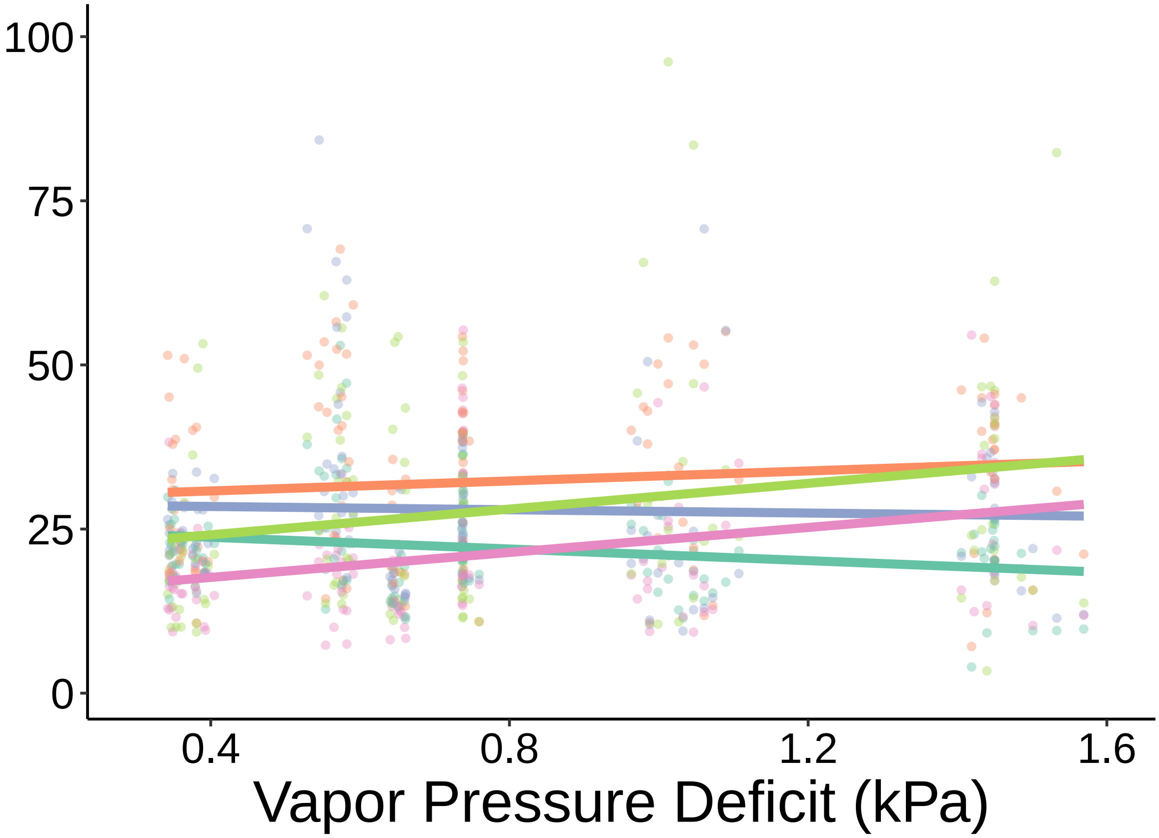
```
# lm: CEWL ~ region + capture temp
CEWL_lm5 <- lm(TEWL_g_m2h ~ region * temp_C_interpol,
           data = CEWL_data_full)
summary(CEWL_lm5)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region * temp_C_interpol, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -30.410  -8.223  -1.995   5.634  66.793
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         27.9085     7.9871   3.494 0.000509 ***
## regionVentrum                      -11.6762    11.3599  -1.028 0.304419
## regionHead                          -3.4845    11.3599  -0.307 0.759147
## regionDewlap                       -39.3213    11.3626  -3.461 0.000575 ***
## regionMite Patch                   -23.8290    11.3021  -2.108 0.035392 *
## temp_C_interpol                     -0.3132     0.4210  -0.744 0.457122
## regionVentrum:temp_C_interpol        1.1674     0.5993   1.948 0.051865 .
## regionHead:temp_C_interpol           0.5015     0.5993   0.837 0.403057
## regionDewlap:temp_C_interpol         2.0550     0.5997   3.427 0.000650 ***
## regionMite Patch:temp_C_interpol     1.5804     0.5961   2.651 0.008221 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.22 on 630 degrees of freedom
##   (59 observations deleted due to missingness)
## Multiple R-squared:  0.1406, Adjusted R-squared:  0.1283
## F-statistic: 11.45 on 9 and 630 DF,  p-value: < 2.2e-16
```

CEWL ~ Capture VPD

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_jitter(aes(x = VPD_kPa_int,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              size = 1,
              alpha = 0.4) +
  stat_smooth(aes(x = VPD_kPa_int,
                   y = TEWL_g_m2h,
                   color = region
                   ),
               formula = y ~ x,
               method = "lm",
               se = F,
               size = 1.6,
               alpha = 1 ) +
  theme_classic() +
  xlab("Vapor Pressure Deficit (kPa)") +
  #ylab(bquote('CEWL (g/'*m^2*'h)')) +
  ylab("") +
  #annotate("text", x = 1, y = 65, label = "a", size = 6) +
  scale_color_brewer(palette = "Set2",
                     name = "") +
  ylim(1, 100) +
  xlim(0.3, 1.6) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 22),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 16),
        legend.text = element_text(color = "black",
                                   family = "sans",
                                   size = 26),
        legend.text.align = 0,
        plot.margin = unit(c(0.1,0,0.35,0.45), "cm"),
        legend.position = "none"
        #legend.position = c(0.15, 0.85)
        ) -> CEWL_vpd_fig
CEWL_vpd_fig
```

```
## Warning: Removed 59 rows containing non-finite values (stat_smooth).
## Warning: Removed 59 rows containing missing values (geom_point).
```



```
# export figure
#ggsave(filename = "CEWL_vpd_fig.tiff",
 #      plot = CEWL_vpd_fig,
  #     path = "./final_figures",
   #    device = "tiff",
    #   dpi = 1200,
     #  width = 6, height = 4)
```

```
CEWL_lm6 <- lm(TEWL_g_m2h ~ region * VPD_kPa_int,
           data = CEWL_data_full)
summary(CEWL_lm6)
```

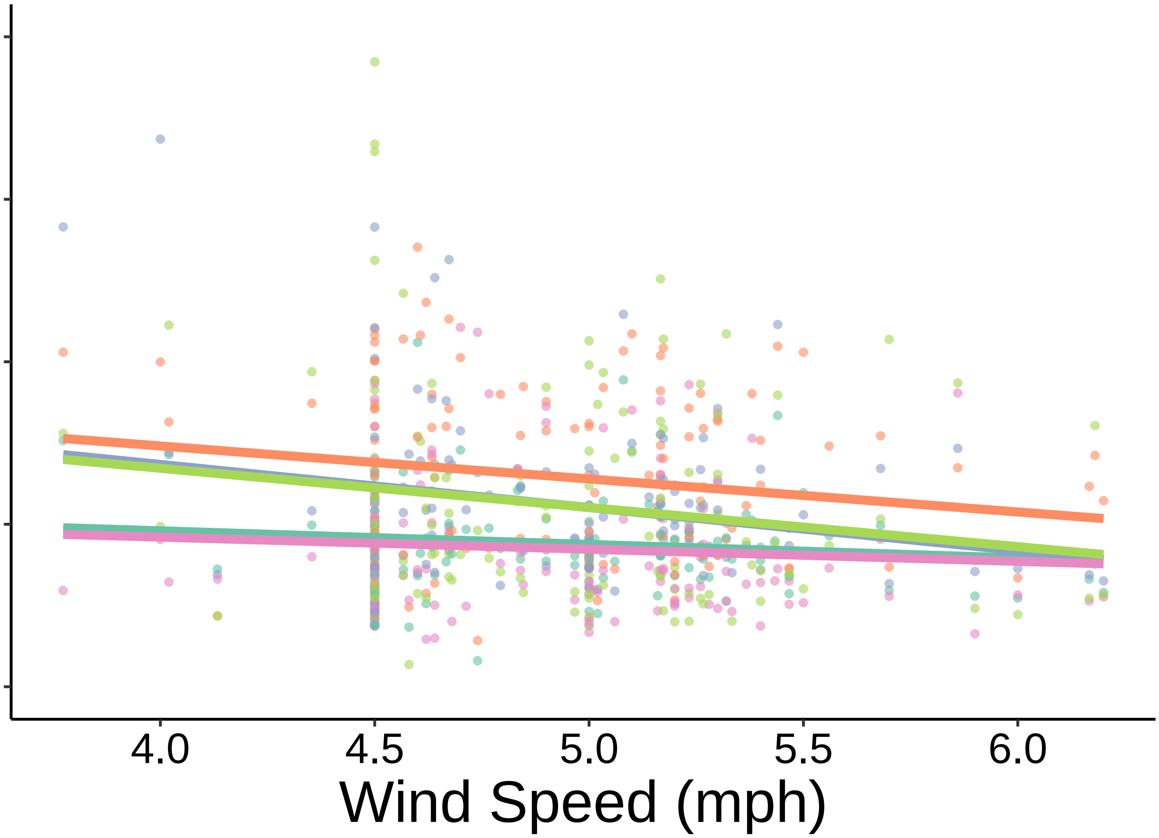
```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region * VPD_kPa_int, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -30.892  -8.435  -2.079   5.884  66.037
##
## Coefficients:
```

```
##                              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                    25.526      2.555   9.989  < 2e-16 ***
## regionVentrum                   3.750      3.625   1.035 0.301222    
## regionHead                      3.410      3.625   0.941 0.347194    
## regionDewlap                  -11.696      3.625  -3.227 0.001316 ** 
## regionMite Patch               -5.319      3.614  -1.472 0.141604    
## VPD_kPa_int                    -4.452      2.941  -1.514 0.130602    
## regionVentrum:VPD_kPa_int       8.265      4.183   1.976 0.048630 *  
## regionHead:VPD_kPa_int          3.203      4.183   0.766 0.444110    
## regionDewlap:VPD_kPa_int       13.958      4.188   3.333 0.000910 ***
## regionMite Patch:VPD_kPa_int   14.244      4.168   3.417 0.000673 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 12.27 on 630 degrees of freedom
##   (59 observations deleted due to missingness)
## Multiple R-squared:  0.1339, Adjusted R-squared:  0.1215 
## F-statistic: 10.82 on 9 and 630 DF,  p-value: 1.008e-15
```

CEWL ~ Wind Speed

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = Wind_mph_interpol,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = Wind_mph_interpol,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  scale_color_brewer(palette = "Set2") +
  xlab("Wind Speed (mph)") +
  #ylab(bquote('CEWL (g/'*m^2*'h)')) +
  ylab("") +
  ylim(0,100) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 22),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 16),
        legend.text = element_text(color = "black",
                                 family = "sans",
                                 size = 26),
        axis.text.y = element_blank(),
        plot.margin = unit(c(0.1,0,0.35,0.45), "cm"),
        legend.text.align = 0,
        legend.position = "none"
) -> wind_CEWL_plot
wind_CEWL_plot
```

```
## Warning: Removed 59 rows containing non-finite values (stat_smooth).

## Warning: Removed 59 rows containing missing values (geom_point).
```



model:

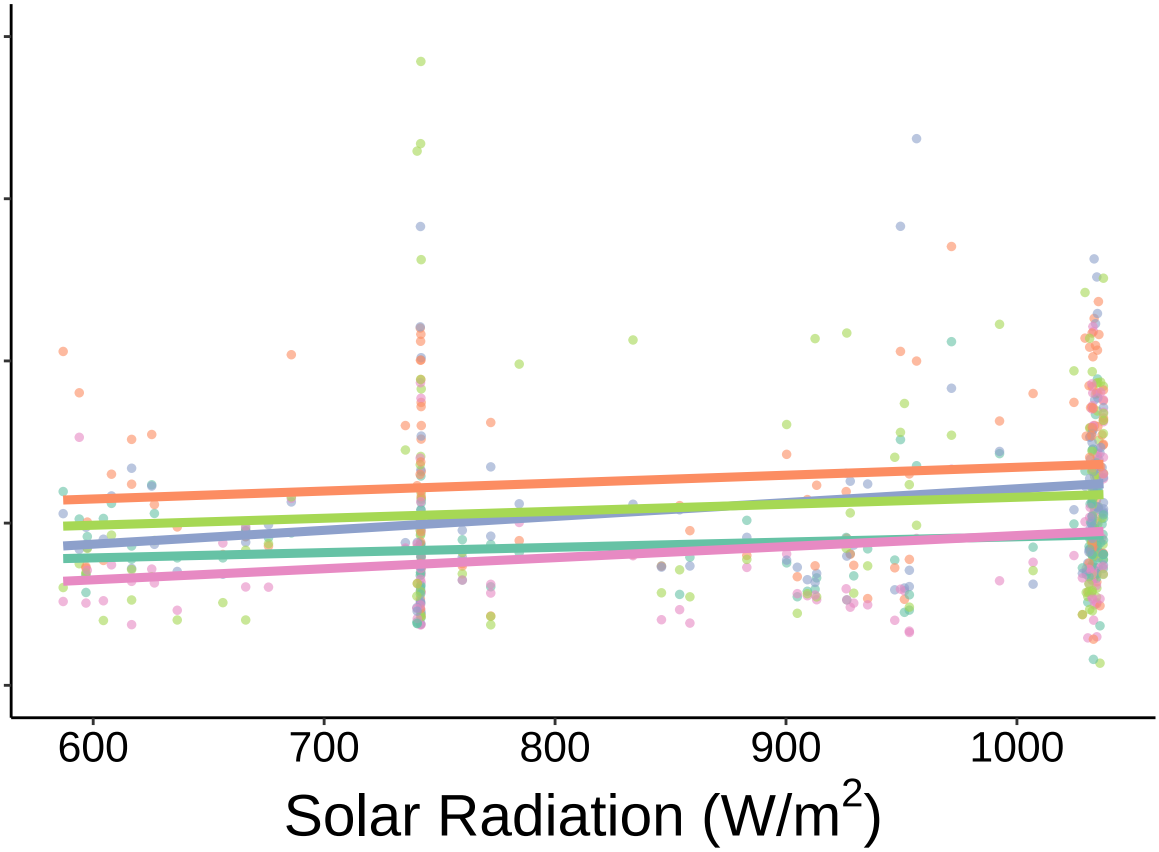
```
# lm: CEWL ~ region + aquaflux measurement RH
CEWL_lm7 <- lm(TEWL_g_m2h ~ region + Wind_mph_interpol,
           data = CEWL_data_full)
summary(CEWL_lm7)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region + Wind_mph_interpol, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -26.075  -8.470  -2.166   5.884  66.328
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          43.4629     5.5784   7.791 2.72e-14 ***
## regionVentrum      10.2431     1.5344   6.676 5.38e-11 ***
## regionHead          5.9349     1.5344   3.868 0.000121 ***
## regionDewlap       -0.7384     1.5405  -0.479 0.631886    
## regionMite Patch    5.8928     1.5436   3.818 0.000148 ***
## Wind_mph_interpol  -4.3387     1.1071  -3.919 9.87e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.32 on 634 degrees of freedom
##   (59 observations deleted due to missingness)
## Multiple R-squared:  0.1204, Adjusted R-squared:  0.1135
## F-statistic: 17.36 on 5 and 634 DF,  p-value: 3.977e-16
```

CEWL ~ Solar Rad

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = Solar_rad_Wm2_interpol,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = Solar_rad_Wm2_interpol,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  scale_color_brewer(palette = "Set2") +
  xlab(bquote('Solar Radiation (W/'*m^2*')')) +
  #ylab(bquote('CEWL (g/'*m^2*'h)')) +
  ylab("") +
  ylim(0,100) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 22),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 16),
        legend.text = element_text(color = "black",
                                 family = "sans",
                                 size = 26),
        axis.text.y = element_blank(),
        plot.margin = unit(c(0.1,0,0,0.45), "cm"),
        legend.text.align = 0,
        legend.position = "none"
) -> sorad_CEWL_plot
sorad_CEWL_plot
```

```
## Warning: Removed 59 rows containing non-finite values (stat_smooth).

## Warning: Removed 59 rows containing missing values (geom_point).
```



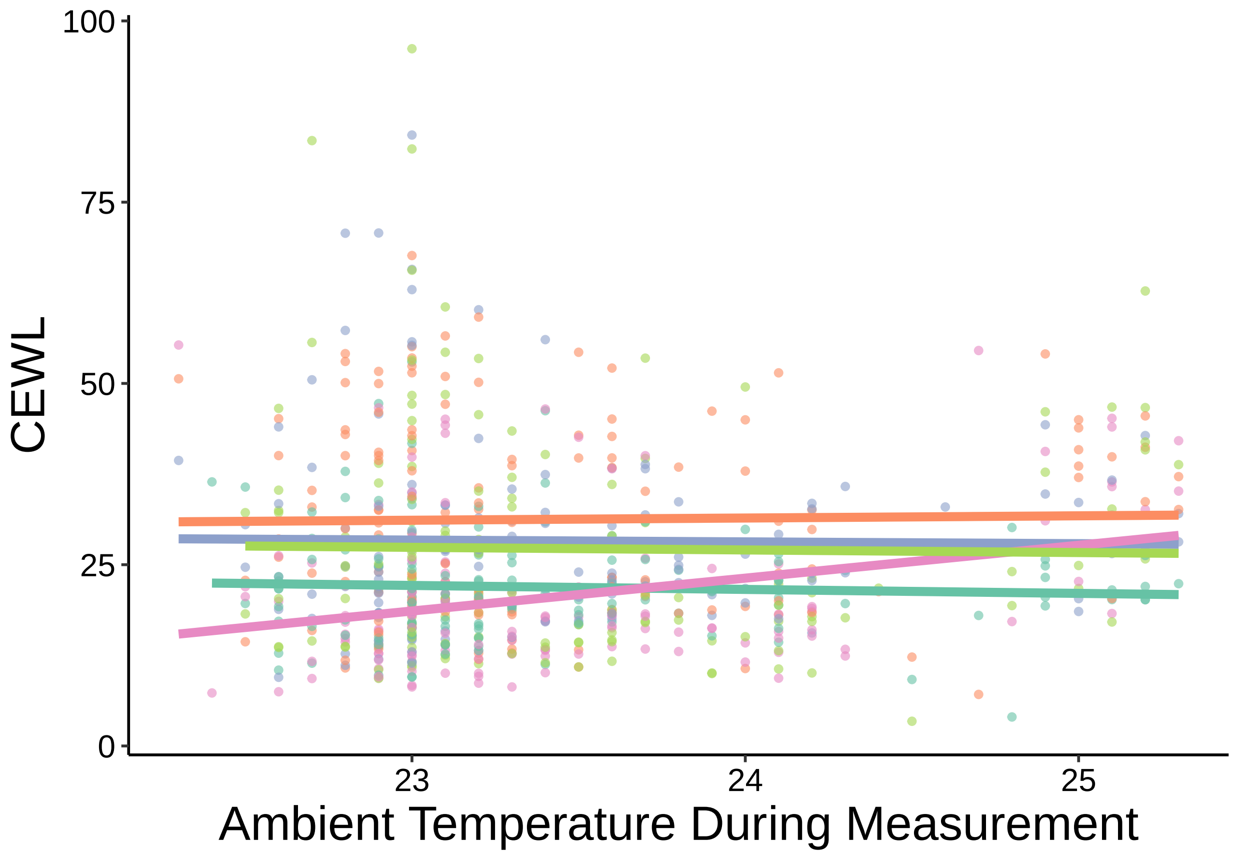
```
# lm: CEWL ~ region + aquaflux measurement RH
CEWL_lm8 <- lm(TEWL_g_m2h ~ region + Solar_rad_Wm2_interpol,
           data = CEWL_data_full)
summary(CEWL_lm8)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region + Solar_rad_Wm2_interpol, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -27.144  -8.072  -2.283   5.790  70.440
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)             9.596283   3.002380   3.196 0.001462 **
## regionVentrum          10.277224   1.529365   6.720 4.05e-11 ***
## regionHead              5.968969   1.529365   3.903 0.000105 ***
## regionDewlap           -0.766911   1.535357  -0.500 0.617600
## regionMite Patch        5.797146   1.538466   3.768 0.000180 ***
## Solar_rad_Wm2_interpol  0.013920   0.003138   4.436 1.08e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 12.28 on 634 degrees of freedom
##   (59 observations deleted due to missingness)
## Multiple R-squared:  0.1262, Adjusted R-squared:  0.1194
## F-statistic: 18.32 on 5 and 634 DF,  p-value: < 2.2e-16
```

CEWL ~ Measurement Temperature

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = ambient_temp_C,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = ambient_temp_C,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  scale_color_brewer(palette = "Set2") +
  xlab("Ambient Temperature During Measurement") +
  ylab("CEWL") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 18),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 12),
        legend.text.align = 0,
        legend.position = "none"
)
```



```r
# lm: CEWL ~ region + aquaflux measurement temp
CEWL_lm9 <- lm(TEWL_g_m2h ~ region * ambient_temp_C,
           data = CEWL_data_full)
summary(CEWL_lm9)
```

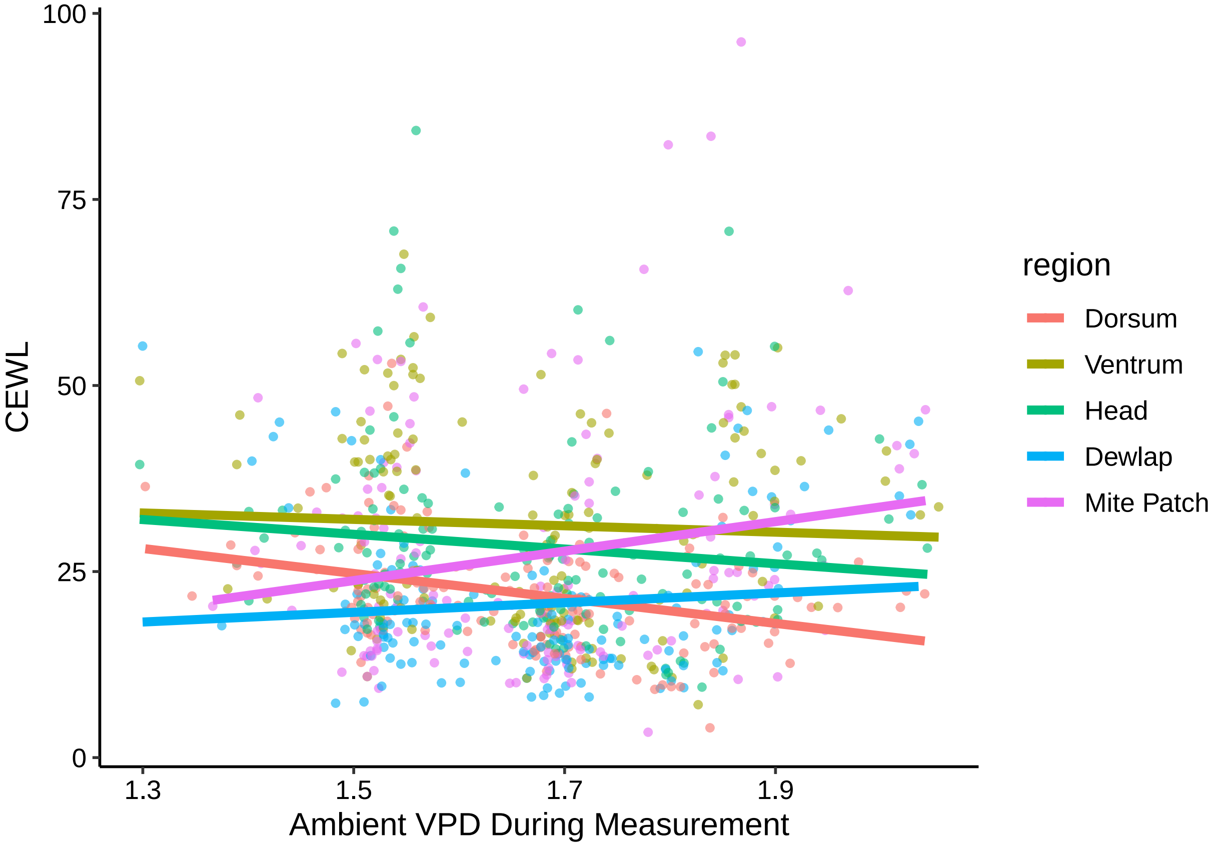
```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region * ambient_temp_C, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.542  -8.438  -2.262   5.727  68.764
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     34.7189    34.9450   0.994   0.3208
## regionVentrum                  -10.5974    49.1621  -0.216   0.8294
## regionHead                      -0.6233    49.3425  -0.013   0.9899
## regionDewlap                  -120.4092    49.9060  -2.413   0.0161 *
## regionMite Patch                 0.9480    49.2468   0.019   0.9846
## ambient_temp_C                  -0.5470     1.4907  -0.367   0.7138
## regionVentrum:ambient_temp_C     0.8518     2.0974   0.406   0.6848
## regionHead:ambient_temp_C        0.2992     2.1054   0.142   0.8870
## regionDewlap:ambient_temp_C      5.0819     2.1281   2.388   0.0172 *
## regionMite Patch:ambient_temp_C  0.1874     2.0990   0.089   0.9289
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 12.36 on 689 degrees of freedom
## Multiple R-squared:  0.1069, Adjusted R-squared:  0.09525
## F-statistic: 9.165 on 9 and 689 DF,  p-value: 3.517e-13
```

CEWL ~ Measurement VPD

Very interesting relationship! Mite patch CEWL decreases as VPD increases, but every other location appears to increase. In this case, an interaction term is warranted.

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = VPD_kPa,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = VPD_kPa,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Ambient VPD During Measurement") +
  ylab("CEWL") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)
```



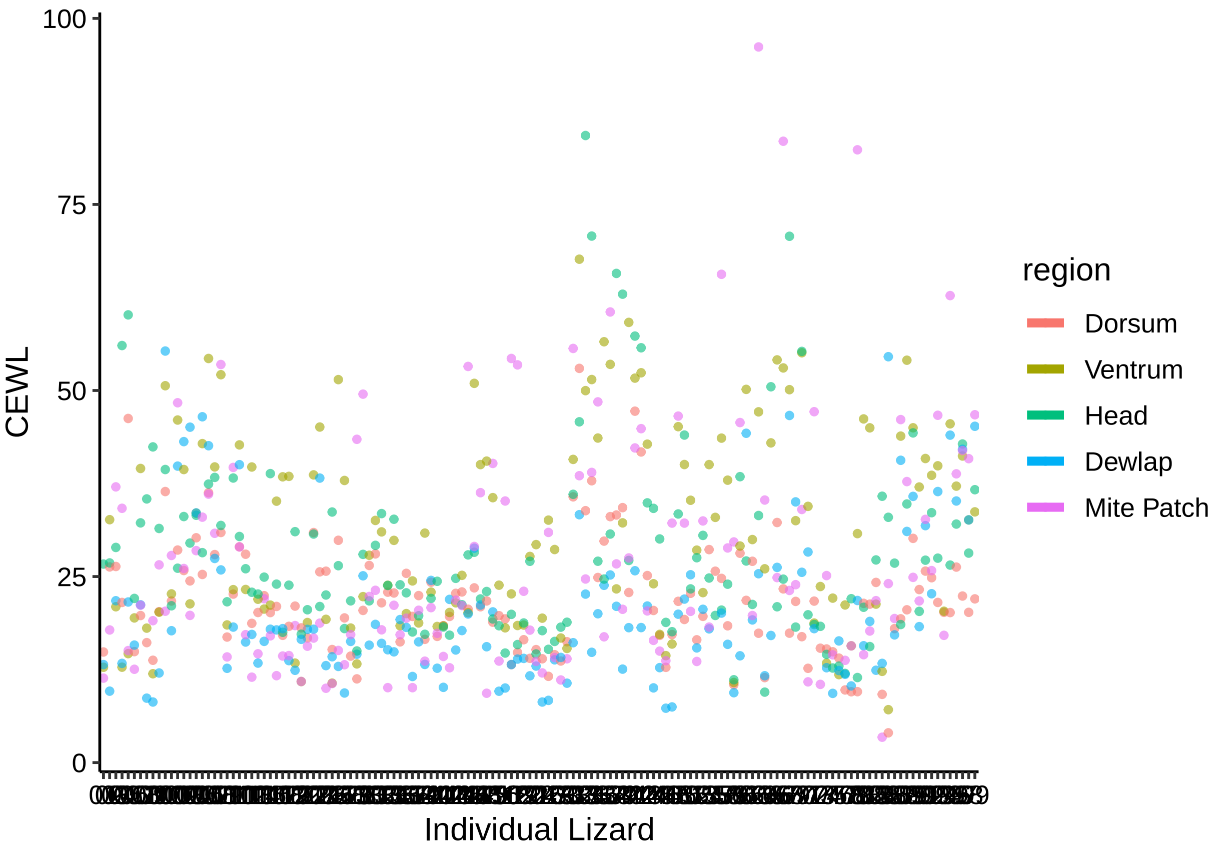
```
# lm: CEWL ~ region + aquaflux measurement RH
CEWL_lm10 <- lm(TEWL_g_m2h ~ region * ambient_RH_percent,
           data = CEWL_data_full)
summary(CEWL_lm10)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region * ambient_RH_percent, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.032  -8.244  -2.423   6.029  62.330
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         -3.9797     9.9831  -0.399  0.69028
## regionVentrum                       26.2701    14.0677   1.867  0.06227 .
## regionHead                          16.5311    14.1008   1.172  0.24146
## regionDewlap                        16.5378    14.2495   1.161  0.24621
## regionMite Patch                    65.3763    14.4063   4.538 6.70e-06 ***
## ambient_RH_percent                   0.5931     0.2275   2.607  0.00934 **
## regionVentrum:ambient_RH_percent    -0.3868     0.3212  -1.204  0.22891
## regionHead:ambient_RH_percent       -0.2310     0.3220  -0.717  0.47337
## regionDewlap:ambient_RH_percent     -0.4074     0.3252  -1.253  0.21070
## regionMite Patch:ambient_RH_percent -1.3763     0.3284  -4.190 3.15e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 12.25 on 689 degrees of freedom
## Multiple R-squared:  0.1227, Adjusted R-squared:  0.1112
## F-statistic:  10.7 on 9 and 689 DF,  p-value: 1.218e-15
```

CEWL ~ Individual

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = individual_ID,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = individual_ID,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Individual Lizard") +
  ylab("CEWL") +

  # just to get a better look
  # ylim(5, 40) +

  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)
```



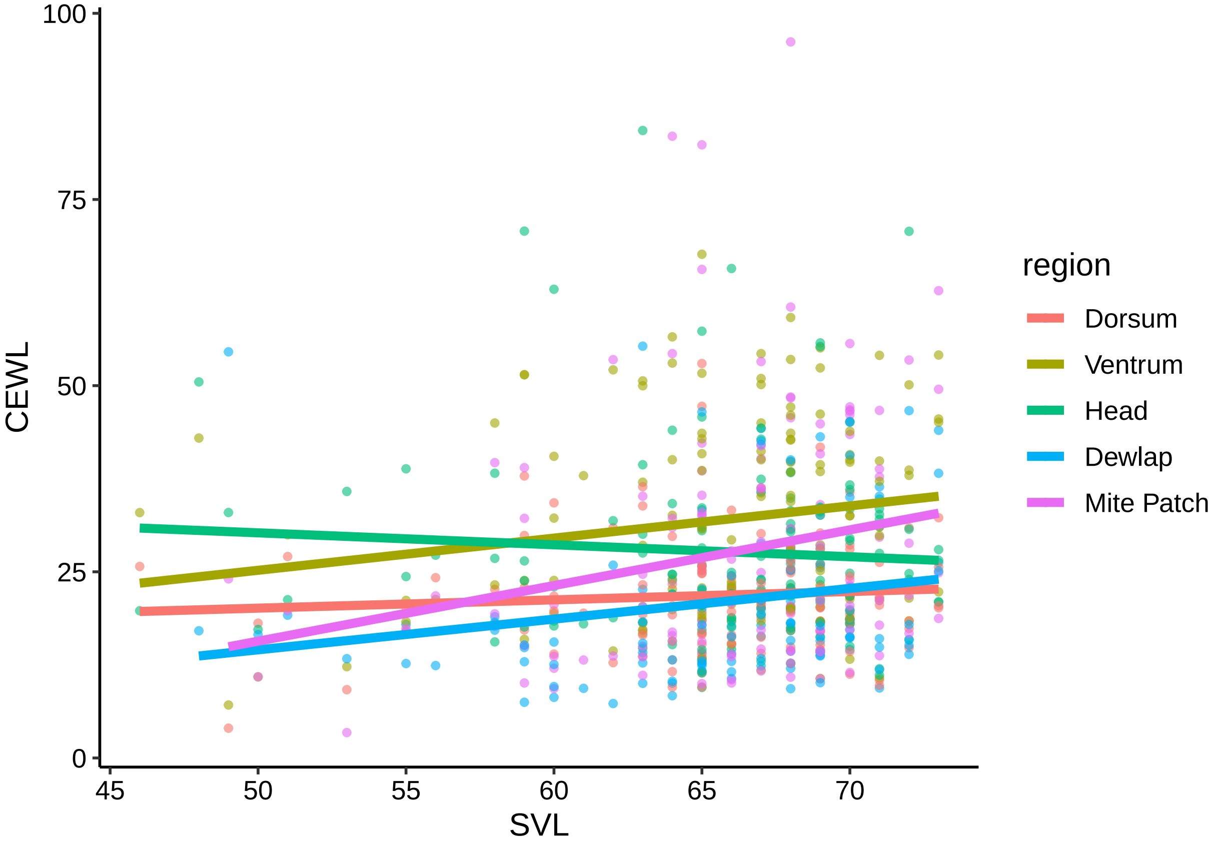
```
# lm
CEWL_lm11 <- lm(TEWL_g_m2h ~ region + individual_ID,
           data = CEWL_data_full)
summary(CEWL_lm11)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region + individual_ID, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -22.805  -5.066  -0.605   3.882  50.951
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        11.8646     4.6583   2.547 0.011135 *
## regionVentrum       9.2590     1.2245   7.561 1.67e-13 ***
## regionHead          6.2652     1.2247   5.116 4.32e-07 ***
## regionDewlap       -1.2460     1.2314  -1.012 0.312063
## regionMite Patch    5.2689     1.2344   4.268 2.32e-05 ***
## individual_ID02     6.8620     6.4959   1.056 0.291264
## individual_ID03    11.2260     6.4959   1.728 0.084516 .
## individual_ID04    11.8020     6.4959   1.817 0.069782 .
## individual_ID05    15.7600     6.4959   2.426 0.015579 *
## individual_ID06     1.1740     6.4959   0.181 0.856646
## individual_ID07    10.9940     6.4959   1.692 0.091122 .
## individual_ID08     4.1359     6.8927   0.600 0.548729
## individual_ID09    3.2900     6.4959   0.506 0.612725    
## individual_ID10    6.3380     6.4959   0.976 0.329642    
## individual_ID100  24.6420     6.4959   3.793 0.000165 ***
## individual_ID101   6.4260     6.4959   0.989 0.322977    
## individual_ID102  22.0060     6.4959   3.388 0.000755 ***
## individual_ID103  17.7180     6.4959   2.728 0.006583 ** 
## individual_ID104  12.2440     6.4959   1.885 0.059969 .  
## individual_ID105  16.0240     6.4959   2.467 0.013935 *  
## individual_ID106  19.3860     6.4959   2.984 0.002967 ** 
## individual_ID107  25.5560     6.4959   3.934 9.41e-05 ***
## individual_ID108  17.0740     6.4959   2.628 0.008816 ** 
## individual_ID109  23.0830     6.4959   3.553 0.000413 ***
## individual_ID11    0.9960     6.4959   0.153 0.878196    
## individual_ID110  12.6240     6.4959   1.943 0.052477 .  
## individual_ID111  18.4400     6.4959   2.839 0.004696 ** 
## individual_ID112   6.3660     6.4959   0.980 0.327512    
## individual_ID113   6.2300     6.4959   0.959 0.337943    
## individual_ID114   2.7780     6.4959   0.428 0.669068    
## individual_ID115   5.4820     6.4959   0.844 0.399079    
## individual_ID116   7.2460     6.4959   1.115 0.265131    
## individual_ID117   6.1460     6.4959   0.946 0.344491    
## individual_ID118   5.2920     6.4959   0.815 0.415612    
## individual_ID119   5.9640     6.4959   0.918 0.358956    
## individual_ID12    3.4740     6.4959   0.535 0.593003    
## individual_ID120  -1.0340     6.4959  -0.159 0.873587    
## individual_ID122   2.1560     6.4959   0.332 0.740088    
## individual_ID123  11.2120     6.4959   1.726 0.084903 .  
## individual_ID124  13.9580     6.4959   2.149 0.032088 *  
## individual_ID125   2.3280     6.4959   0.358 0.720195    
## individual_ID126   1.1020     6.4959   0.170 0.865351    
## individual_ID127  11.3820     6.4959   1.752 0.080296 .  
## individual_ID128   3.7980     6.4959   0.585 0.559002    
## individual_ID129   1.7460     6.4959   0.269 0.788195    
## individual_ID13    3.7280     6.4959   0.574 0.566267    
## individual_ID130  13.2940     6.4959   2.047 0.041177 *  
## individual_ID131   7.0620     6.4959   1.087 0.277444    
## individual_ID132  10.5240     6.4959   1.620 0.105780    
## individual_ID133   8.1840     6.4959   1.260 0.208246    
## individual_ID134   3.3780     6.4959   0.520 0.603256    
## individual_ID135   8.5000     6.4959   1.309 0.191240    
## individual_ID136   3.2040     6.4959   0.493 0.622042    
## individual_ID137   5.3940     6.4959   0.830 0.406687    
## individual_ID138   0.8740     6.4959   0.135 0.893019    
## individual_ID139   3.7500     6.4959   0.577 0.563979    
## individual_ID14    2.5180     6.4959   0.388 0.698439    
## individual_ID140   7.1280     6.4959   1.097 0.272983    
## individual_ID141   2.1580     6.4959   0.332 0.739856    
## individual_ID142   0.0960     6.4959   0.015 0.988214    
## individual_ID143   2.5460     6.4959   0.392 0.695253    
## individual_ID144   5.4120     6.4959   0.833 0.405123    
## individual_ID145   5.8540     6.4959   0.901 0.367881    
## individual_ID146  12.6000     6.4959   1.940 0.052925 .  
## individual_ID147  16.3360     6.4959   2.515 0.012192 *  
## individual_ID148  12.3080     6.4959   1.895 0.058648 .  
## individual_ID149   6.2480     6.4959   0.962 0.336550    
## individual_ID15   11.0700     6.4959   1.704 0.088913 .  
## individual_ID150   1.2660     6.4959   0.195 0.845548    
## individual_ID16    3.6880     6.4959   0.568 0.570439    
## individual_ID17    8.8720     6.4959   1.366 0.172560    
## individual_ID18    7.5120     6.4959   1.156 0.248007    
## individual_ID19    2.3800     6.4959   0.366 0.714217    
## individual_ID20    3.8700     6.4959   0.596 0.551578    
## individual_ID21    1.3540     6.4959   0.208 0.834962    
## individual_ID24   -1.5180     6.4959  -0.234 0.815315    
## individual_ID26    3.9620     6.4959   0.610 0.542162    
## individual_ID27    1.6940     6.4959   0.261 0.794359    
## individual_ID29   -1.0100     6.4959  -0.155 0.876497    
## individual_ID30   -0.7620     6.4959  -0.117 0.906661    
## individual_ID31   21.0780     6.4959   3.245 0.001246 ** 
## individual_ID32   31.8780     6.4959   4.907 1.22e-06 ***
## individual_ID33   27.3060     6.4959   4.204 3.06e-05 ***
## individual_ID34   27.0060     6.4959   4.157 3.73e-05 ***
## individual_ID35   17.0280     6.4959   2.621 0.008999 ** 
## individual_ID36   14.5640     6.4959   2.242 0.025355 *  
## individual_ID37   24.8320     6.4959   3.823 0.000147 ***
## individual_ID38   18.2340     6.4959   2.807 0.005177 ** 
## individual_ID39   16.7400     6.4959   2.577 0.010224 *  
## individual_ID40   15.1860     6.4959   2.338 0.019754 *  
## individual_ID41   29.0760     6.4959   4.476 9.24e-06 ***
## individual_ID42   26.8000     6.4959   4.126 4.27e-05 ***
## individual_ID43   13.0740     6.4959   2.013 0.044635 *  
## individual_ID44    5.2440     6.4959   0.807 0.419852    
## individual_ID45    2.6480     6.4959   0.408 0.683694    
## individual_ID46   -2.3720     6.4959  -0.365 0.715136    
## individual_ID47    2.3020     6.4959   0.354 0.723190    
## individual_ID48   17.5700     6.4959   2.705 0.007045 ** 
## individual_ID50   15.7140     6.4959   2.419 0.015882 *  
## individual_ID51    9.6060     6.4959   1.479 0.139768    
## individual_ID52    4.5420     6.4959   0.699 0.484713    
## individual_ID53    9.4440     6.4959   1.454 0.146557    
## individual_ID54   10.1540     6.4959   1.563 0.118591    
## individual_ID55    9.1074     7.5076   1.213 0.225613    
## individual_ID57   19.1340     6.4959   2.946 0.003359 ** 
## individual_ID58    9.2320     6.4959   1.421 0.155820    
## individual_ID59   -1.4960     6.4959  -0.230 0.817944    
## individual_ID60   15.3640     6.4959   2.365 0.018365 *  
## individual_ID61   20.3884     6.8927   2.958 0.003229 ** 
## individual_ID62    7.6660     6.4959   1.180 0.238455    
## individual_ID63   28.0760     6.4959   4.322 1.83e-05 ***
## individual_ID64    3.0020     6.4959   0.462 0.644163    
## individual_ID65   20.2227     7.5076   2.694 0.007282 ** 
## individual_ID66   15.9060     6.4959   2.449 0.014650 *  
## individual_ID67   29.0647     6.8927   4.217 2.90e-05 ***
## individual_ID68   25.8240     6.4959   3.975 7.96e-05 ***
## individual_ID69   10.4900     6.4959   1.615 0.106909    
## individual_ID70   21.5900     6.4959   3.324 0.000947 ***
## individual_ID71    5.4480     6.4959   0.839 0.402008    
## individual_ID72    9.0560     6.4959   1.394 0.163843    
## individual_ID73   -0.1003     6.8927  -0.015 0.988390
## individual_ID74    0.4360     6.4959   0.067 0.946511
## individual_ID75   -1.0780     6.4959  -0.166 0.868256
## individual_ID76   -0.9610     6.5006  -0.148 0.882526
## individual_ID77   -2.0700     6.4959  -0.319 0.750102
## individual_ID78   -1.1300     6.4959  -0.174 0.861963
## individual_ID79   15.3960     6.4959   2.370 0.018125 *
## individual_ID81    7.9560     6.4959   1.225 0.221181
## individual_ID82    7.9280     6.4959   1.220 0.222808
## individual_ID83    5.6060     6.4959   0.863 0.388507
## individual_ID84   -0.9820     6.4959  -0.151 0.879895
## individual_ID85    8.7580     6.4959   1.348 0.178133
## individual_ID86    5.8934     6.8926   0.855 0.392907
## individual_ID87   17.9160     6.4959   2.758 0.006007 **
## individual_ID88   19.8600     6.4959   3.057 0.002341 **
## individual_ID89   20.2420     6.4959   3.116 0.001928 **
## individual_ID90    8.3440     6.4959   1.285 0.199503
## individual_ID91   15.8880     6.4959   2.446 0.014761 *
## individual_ID92   13.3280     6.4959   2.052 0.040663 *
## individual_ID93   18.6180     6.4959   2.866 0.004313 **
## individual_ID94    2.5028     7.5076   0.333 0.738981
## individual_ID95   24.0200     6.4959   3.698 0.000239 ***
## individual_ID96   18.1100     6.4959   2.788 0.005487 **
## individual_ID97   22.3080     6.4959   3.434 0.000639 ***
## individual_ID98   15.1060     6.4959   2.325 0.020407 *
## individual_ID99   21.0840     6.4959   3.246 0.001242 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.27 on 553 degrees of freedom
## Multiple R-squared:  0.5047, Adjusted R-squared:  0.3749
## F-statistic: 3.887 on 145 and 553 DF,  p-value: < 2.2e-16
```

CEWL ~ SVL

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = SVL_mm,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = SVL_mm,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("SVL") +
  ylab("CEWL") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 44 rows containing non-finite values (stat_smooth).

## Warning: Removed 44 rows containing missing values (geom_point).
```



```
# lm
CEWL_lm12 <- lm(TEWL_g_m2h ~ region + SVL_mm,
                data = CEWL_data_full)
summary(CEWL_lm12)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region + SVL_mm, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -22.778  -8.230  -2.224   5.569  67.956
##
## Coefficients:
```

```
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         2.7281     6.3710   0.428 0.668649
## regionVentrum      10.1495     1.5185   6.684 5.01e-11 ***
## regionHead          5.8810     1.5185   3.873 0.000118 ***
## regionDewlap       -0.8336     1.5243  -0.547 0.584654
## regionMite Patch    5.6889     1.5273   3.725 0.000212 ***
## SVL_mm              0.2910     0.0954   3.050 0.002381 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.34 on 649 degrees of freedom
##   (44 observations deleted due to missingness)
## Multiple R-squared:  0.1109, Adjusted R-squared:  0.1041
## F-statistic:  16.2 on 5 and 649 DF,  p-value: 4.529e-15
```

CEWL ~ SMI

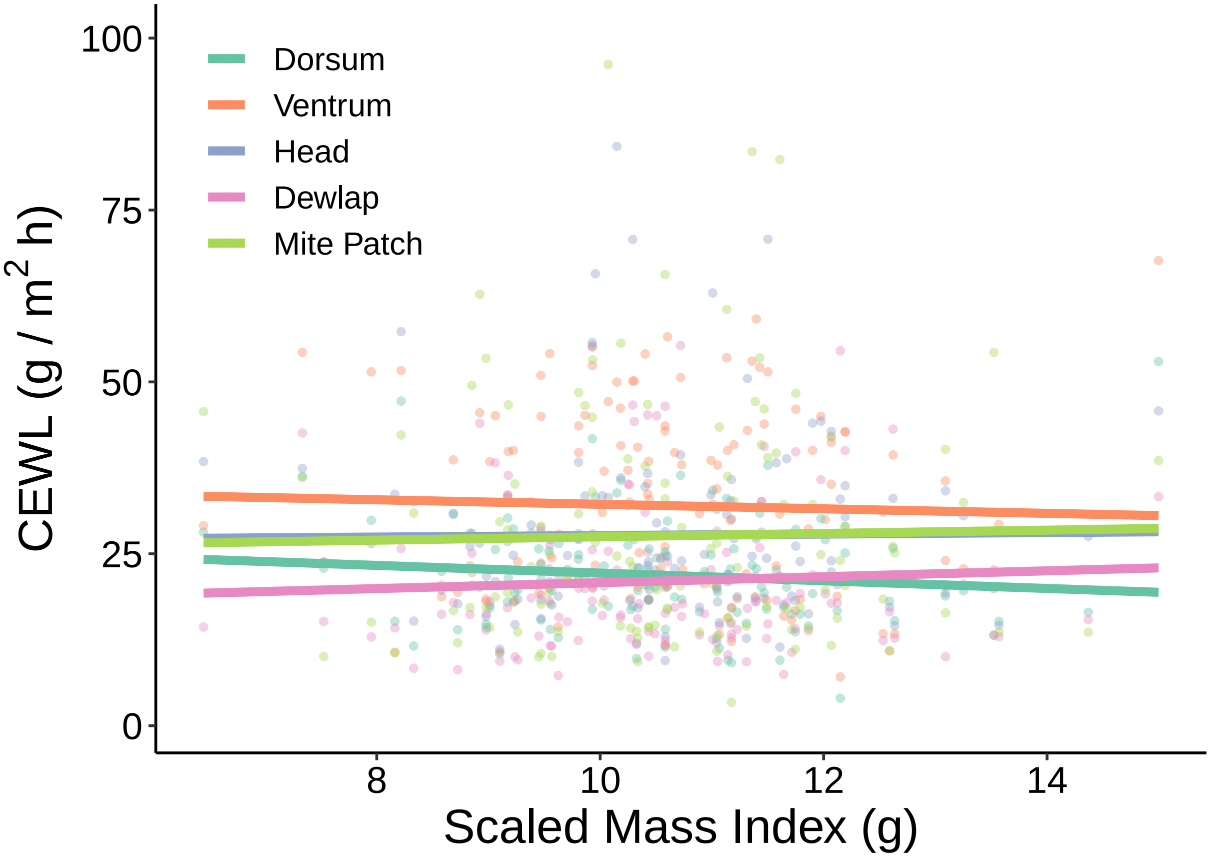
```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = SMI,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.4) +
  stat_smooth(aes(x = SMI,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Scaled Mass Index (g)") +
  ylab(bquote('CEWL (g / '*m^2~h*')')) +
  #annotate("text", x = 1, y = 65, label = "a", size = 6) +
  scale_color_brewer(palette = "Set2",
                     name = "") +
  ylim(1, 100) +
  xlim(2, 16) +
  scale_x_continuous(breaks = c(seq(2, 16, by = 2))) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 18),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 14),
        legend.text = element_text(color = "black",
                                   family = "sans",
                                   size = 12),
        plot.margin = unit(c(0.1,0.5,0.1,0.1), "cm"),
        legend.text.align = 0,
        legend.position = c(0.15, 0.85)
         ) -> CEWL_SMI_fig
```

```
## Scale for 'x' is already present. Adding another scale for 'x', which will
## replace the existing scale.
```

```
CEWL_SMI_fig
```

```
## Warning: Removed 44 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 44 rows containing missing values (geom_point).
```



```
# export figure
#ggsave(filename = "CEWL_mass_fig.tiff",
 #      plot = CEWL_mass_fig,
  #     path = "./final_figures",
   #    device = "tiff",
    #   dpi = 1200,
     #  width = 6, height = 4)


# lm
CEWL_lm13 <- lm(TEWL_g_m2h ~ region + SMI,
           data = CEWL_data_full)
summary(CEWL_lm13)
```

```
##
## Call:
```

```
## lm(formula = TEWL_g_m2h ~ region + SMI, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.867  -8.436  -2.543   5.906  68.539
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      22.10520    3.85059   5.741 1.45e-08 ***
## regionVentrum    10.12784    1.52929   6.623 7.41e-11 ***
## regionHead        5.83477    1.52928   3.815 0.000149 ***
## regionDewlap     -0.82970    1.53515  -0.540 0.589062
## regionMite Patch  5.72841    1.53814   3.724 0.000213 ***
## SMI              -0.02108    0.34976  -0.060 0.951967
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.42 on 649 degrees of freedom
##   (44 observations deleted due to missingness)
## Multiple R-squared:  0.09821,    Adjusted R-squared:  0.09126
## F-statistic: 14.14 on 5 and 649 DF,  p-value: 3.836e-13
```

CEWL ~ Mass

Head has an opposite trend from all the other body regions, so we need an interaction term.

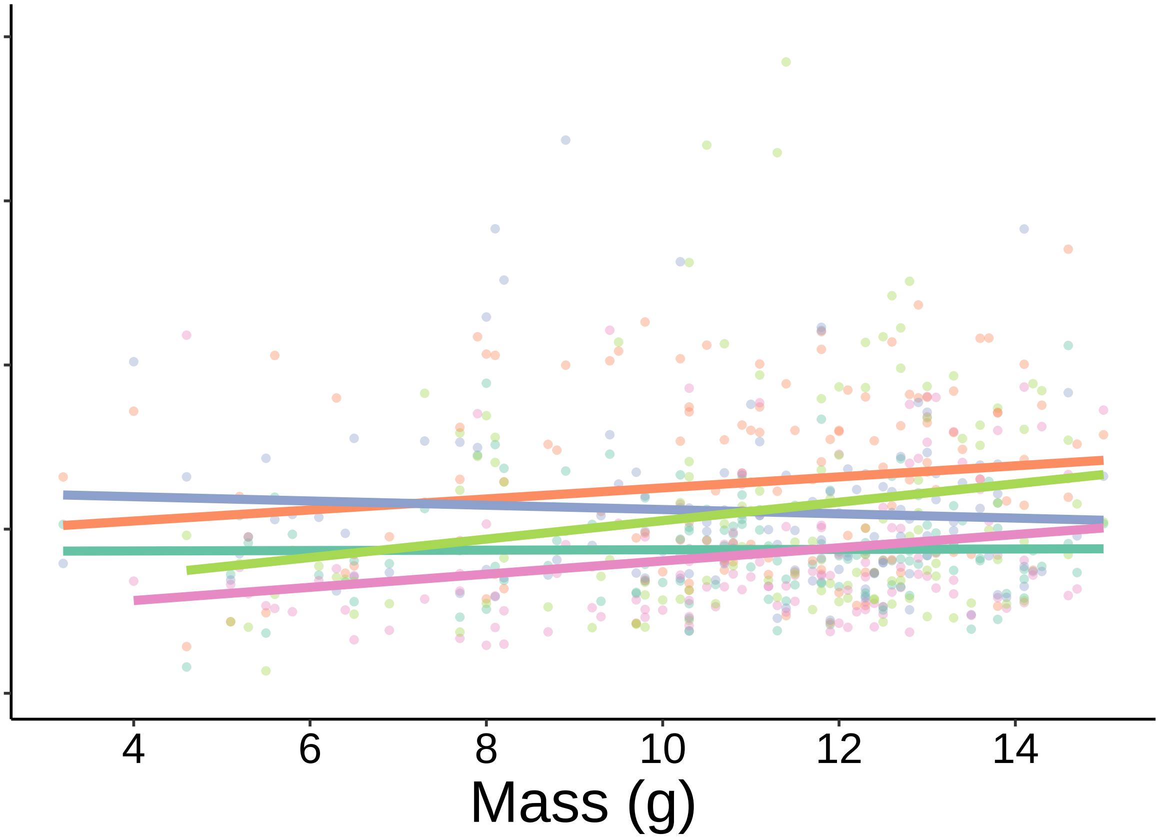
```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = mass_g,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.4) +
  stat_smooth(aes(x = mass_g,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Mass (g)") +
  ylab("") +
  #ylab(bquote('CEWL (g/'*m^2*'h)')) +
  #annotate("text", x = 1, y = 65, label = "a", size = 6) +
  scale_color_brewer(palette = "Set2",
                     name = "") +
  ylim(1, 100) +
  xlim(2, 16) +
  scale_x_continuous(breaks = c(seq(2, 16, by = 2))) +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 22),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                 size = 16),
        axis.text.y = element_blank(),
        legend.text = element_text(color = "black",
                                 family = "sans",
                                 size = 26),
        plot.margin = unit(c(0.1,0,0.35,0.45), "cm"),
        legend.text.align = 0,
        legend.position = "none"
        #legend.position = c(0.5, 0.5)
         ) -> CEWL_mass_fig
```

```
## Scale for 'x' is already present. Adding another scale for 'x', which will
## replace the existing scale.
```

```
CEWL_mass_fig
```

```
## Warning: Removed 44 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 44 rows containing missing values (geom_point).
```



```
# MUST CHANGE COMMENTS ABOVE to produce this
CEWL_legend <- as_ggplot(get_legend(CEWL_mass_fig))
```

```
## Warning: Removed 44 rows containing non-finite values (stat_smooth).
```

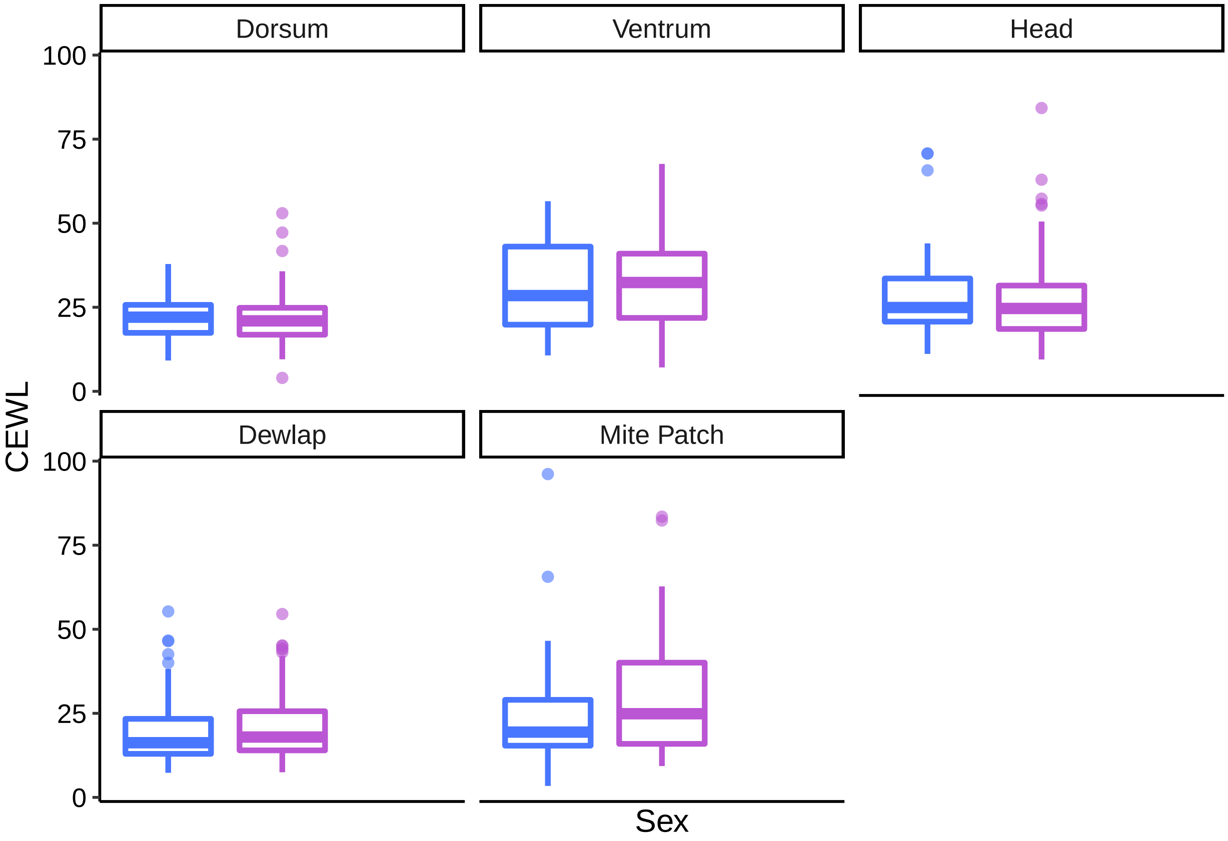
```
## Warning: Removed 44 rows containing missing values (geom_point).
```

```
# export figure
#ggsave(filename = "CEWL_mass_fig.tiff",
 #      plot = CEWL_mass_fig,
  #     path = "./final_figures",
   #    device = "tiff",
    #   dpi = 1200,
     #  width = 6, height = 4)


# lm
CEWL_lm13.1 <- lm(TEWL_g_m2h ~ region*mass_g,
           data = CEWL_data_full)
summary(CEWL_lm13.1)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region * mass_g, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -22.116  -8.402  -2.377   6.152  67.902
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              21.55748    4.73990   4.548 6.47e-06 ***
## regionVentrum             1.31245    6.65331   0.197   0.8437
## regionHead                9.68819    6.60514   1.467   0.1429
## regionDewlap            -11.45967    6.73795  -1.701   0.0895 .
## regionMite Patch         -9.31643    6.84313  -1.361   0.1739
## mass_g                    0.02986    0.42423   0.070   0.9439
## regionVentrum:mass_g      0.81102    0.59567   1.362   0.1738
## regionHead:mass_g        -0.35595    0.59243  -0.601   0.5482
## regionDewlap:mass_g       0.97567    0.60285   1.618   0.1061
## regionMite Patch:mass_g   1.37512    0.61138   2.249   0.0248 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.27 on 645 degrees of freedom
##   (44 observations deleted due to missingness)
## Multiple R-squared:  0.1258, Adjusted R-squared:  0.1136
## F-statistic: 10.31 on 9 and 645 DF,  p-value: 5.958e-15
```

CEWL ~ Sex

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = sex_M_F,
                   y = TEWL_g_m2h,
                   color = sex_M_F
                   ),
               size = 1,
               alpha = 0.6) +
  facet_wrap(~region) +
  scale_color_manual(values = c("royalblue1", "mediumorchid")) +
  scale_x_discrete(breaks = c(1,2,3)) +
  theme_classic() +
  xlab("Sex") +
  ylab("CEWL") +
  theme(text = element_text(color = "black", family = "sans", size = 12),
        axis.text = element_text(color = "black", family = "sans", size = 10),
        legend.text.align = 0,
        legend.position = "none"
)
```

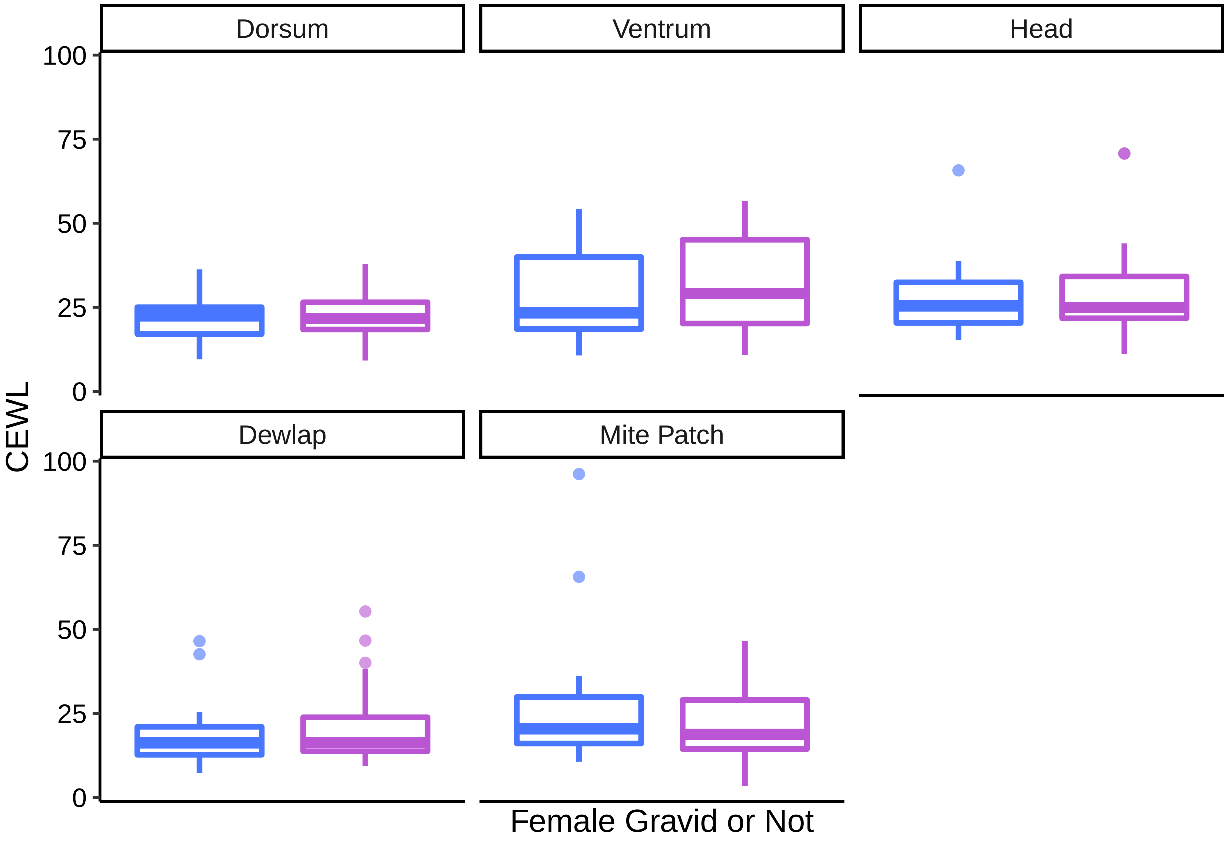


```
# lm
CEWL_lm14 <- lm(TEWL_g_m2h ~ region + sex_M_F,
           data = CEWL_data_full)
summary(CEWL_lm14)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region + sex_M_F, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -25.238  -8.527  -2.446   6.048  69.226
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       21.2061     1.2809  16.555  < 2e-16 ***
## regionVentrum     10.1275     1.5281   6.627 7.19e-11 ***
## regionHead         5.8270     1.5282   3.813 0.000150 ***
## regionDewlap      -0.8403     1.5340  -0.548 0.584045
## regionMite Patch   5.7284     1.5370   3.727 0.000211 ***
## sex_M_FM           1.0146     1.0319   0.983 0.325822
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.41 on 649 degrees of freedom
##   (44 observations deleted due to missingness)
## Multiple R-squared:  0.09955,    Adjusted R-squared:  0.09261
## F-statistic: 14.35 on 5 and 649 DF,  p-value: 2.418e-13
```

CEWL ~ Gravidity

```
CEWL_data_full %>%
  dplyr::filter(sex_M_F == "F") %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = gravid_Y_N,
                   y = TEWL_g_m2h,
                   color = gravid_Y_N
                   ),
               size = 1,
               alpha = 0.6) +
  facet_wrap(~region) +
  scale_color_manual(values = c("royalblue1", "mediumorchid")) +
  scale_x_discrete(breaks = c(1,2,3)) +
  theme_classic() +
  xlab("Female Gravid or Not") +
  ylab("CEWL") +
  theme(text = element_text(color = "black", family = "sans", size = 12),
        axis.text = element_text(color = "black", family = "sans", size = 10),
        legend.text.align = 0,
        legend.position = "none"
)
```

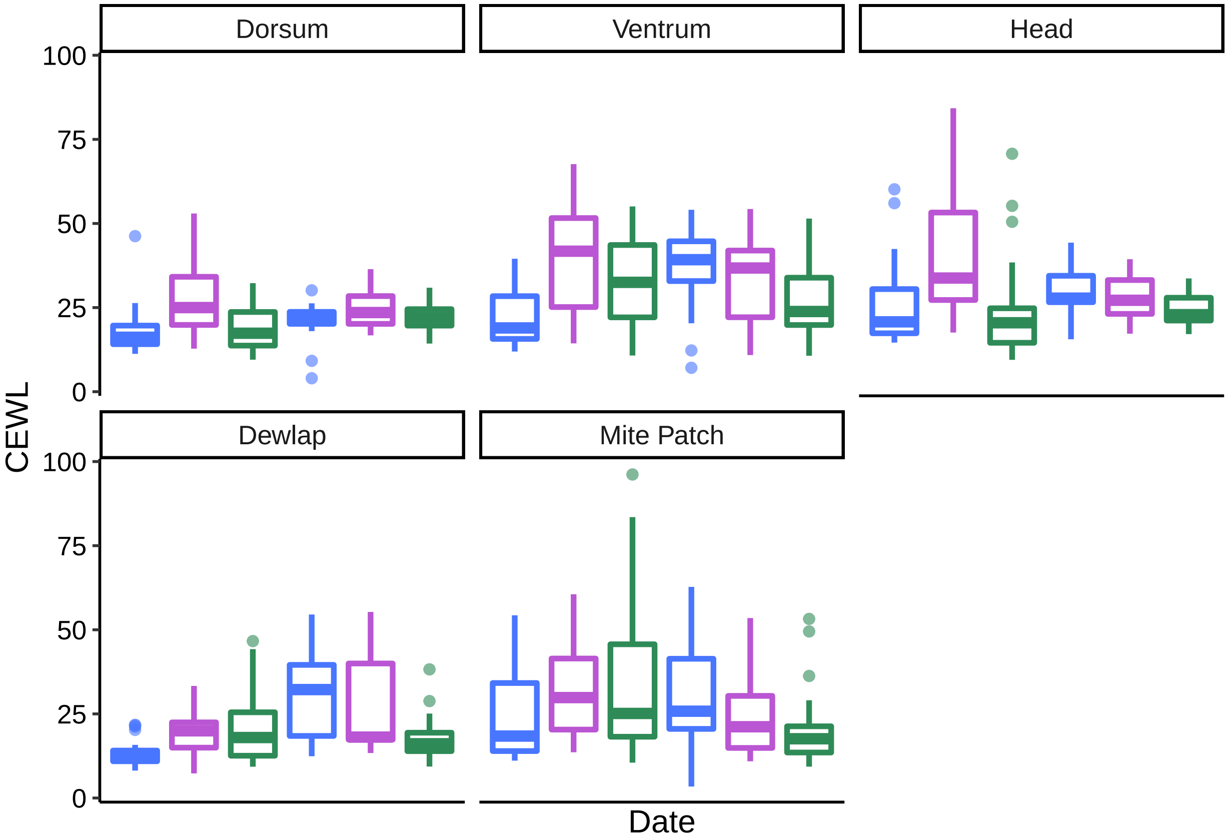


```r
# lm
CEWL_lm15 <- lm(TEWL_g_m2h ~ region + gravid_Y_N,
           data = CEWL_data_full)
summary(CEWL_lm15)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region + gravid_Y_N, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.808  -7.963  -2.657   4.874  72.433
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        21.8066     2.1540  10.124  < 2e-16 ***
## regionVentrum       8.9009     2.7032   3.293  0.00116 **
## regionHead          7.1853     2.7190   2.643  0.00885 **
## regionDewlap       -1.5956     2.7356  -0.583  0.56032
## regionMite Patch    1.9209     2.7190   0.706  0.48068
## gravid_Y_NY         0.4904     1.7475   0.281  0.77927
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.68 on 210 degrees of freedom
##   (483 observations deleted due to missingness)
## Multiple R-squared:  0.09649,    Adjusted R-squared:  0.07498
## F-statistic: 4.486 on 5 and 210 DF,  p-value: 0.000663
```

CEWL ~ Week

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_boxplot(aes(x = as.factor(date),
                   y = TEWL_g_m2h,
                   color = as.factor(date)
                   ),
               size = 1,
               alpha = 0.6) +
  facet_wrap(~region) + # could not figure out how to change facet labels without changing underlying d
  scale_color_manual(values = c("royalblue1", "mediumorchid", "seagreen4",
                                "royalblue1", "mediumorchid", "seagreen4")) +
  scale_x_discrete(breaks = c(1,2,3)) +
  theme_classic() +
  xlab("Date") +
  ylab("CEWL") +
  theme(text = element_text(color = "black", family = "sans", size = 12),
        axis.text = element_text(color = "black", family = "sans", size = 10),
        legend.text.align = 0,
        legend.position = "none"
)
```



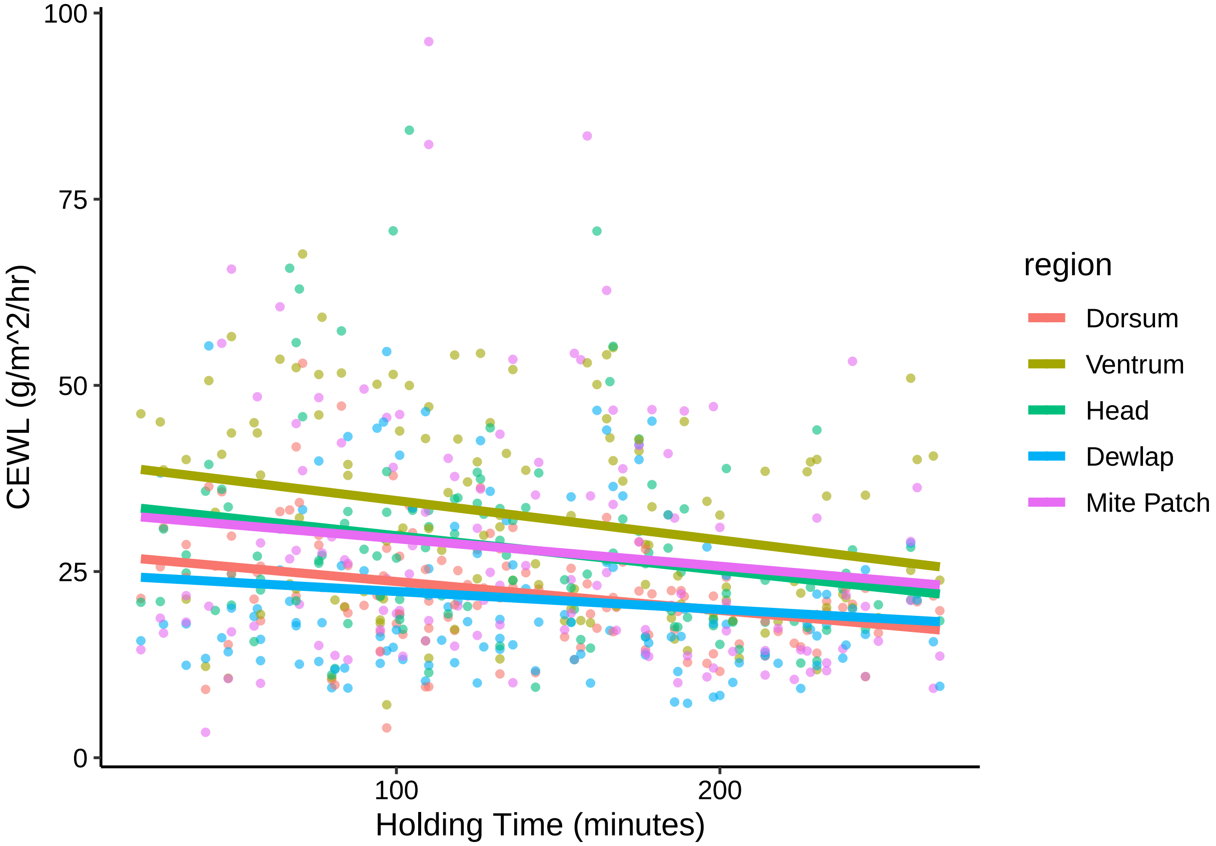
```
# lm
CEWL_lm16 <- lm(TEWL_g_m2h ~ region + date,
           data = CEWL_data_full)
summary(CEWL_lm16)
```

```
##
## Call:
## lm(formula = TEWL_g_m2h ~ region + date, data = CEWL_data_full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.299  -8.269  -2.413   5.549  69.013
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -558.45093  607.60784  -0.919 0.358364
## regionVentrum        9.36218    1.47612   6.342 4.08e-10 ***
## regionHead           6.39242    1.47612   4.331 1.71e-05 ***
## regionDewlap        -1.25433    1.48142  -0.847 0.397449
## regionMite Patch     5.31666    1.48687   3.576 0.000374 ***
## date                 0.03096    0.03241   0.955 0.339837
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.39 on 693 degrees of freedom
## Multiple R-squared:  0.09621,    Adjusted R-squared:  0.08968
## F-statistic: 14.75 on 5 and 693 DF,  p-value: 9.094e-14
```

CEWL ~ holding time

```
CEWL_data_full %>%
  ggplot(data = .) +
  geom_point(aes(x = hold_time,
                 y = TEWL_g_m2h,
                 color = region
                 ),
             size = 1,
             alpha = 0.6) +
  stat_smooth(aes(x = hold_time,
                  y = TEWL_g_m2h,
                  color = region
                  ),
              formula = y ~ x,
              method = "lm",
              se = F,
              size = 1.6,
              alpha = 1 ) +
  theme_classic() +
  xlab("Holding Time (minutes)") +
  ylab("CEWL (g/m^2/hr)") +
  theme(text = element_text(color = "black",
                            family = "sans",
                            size = 12),
        axis.text = element_text(color = "black",
                                 family = "sans",
                                        size = 10),
        legend.text.align = 0,
)
```

```
## Warning: Removed 69 rows containing non-finite values (stat_smooth).

## Warning: Removed 69 rows containing missing values (geom_point).
```



```
CEWL_lm17 <- glm(data = CEWL_data_full,
                 TEWL_g_m2h ~ hold_time + region)
summary(CEWL_lm17)
```

```
##
## Call:
## glm(formula = TEWL_g_m2h ~ hold_time + region, data = CEWL_data_full)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -28.477   -8.346   -2.076    5.613   67.028
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       27.698386   1.524887  18.164  < 2e-16 ***
## hold_time         -0.039935   0.007613  -5.246 2.14e-07 ***
## regionVentrum     10.294533   1.540722   6.682 5.24e-11 ***
## regionHead         5.841615   1.540717   3.791 0.000164 ***
## regionDewlap      -0.739762   1.546876  -0.478 0.632654
## regionMite Patch   5.826284   1.550011   3.759 0.000187 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 150.7369)
##
##     Null deviance: 108878  on 629  degrees of freedom
## Residual deviance:  94060  on 624  degrees of freedom
##   (69 observations deleted due to missingness)
## AIC: 4955.6
##
## Number of Fisher Scoring iterations: 2
```

Conclusion

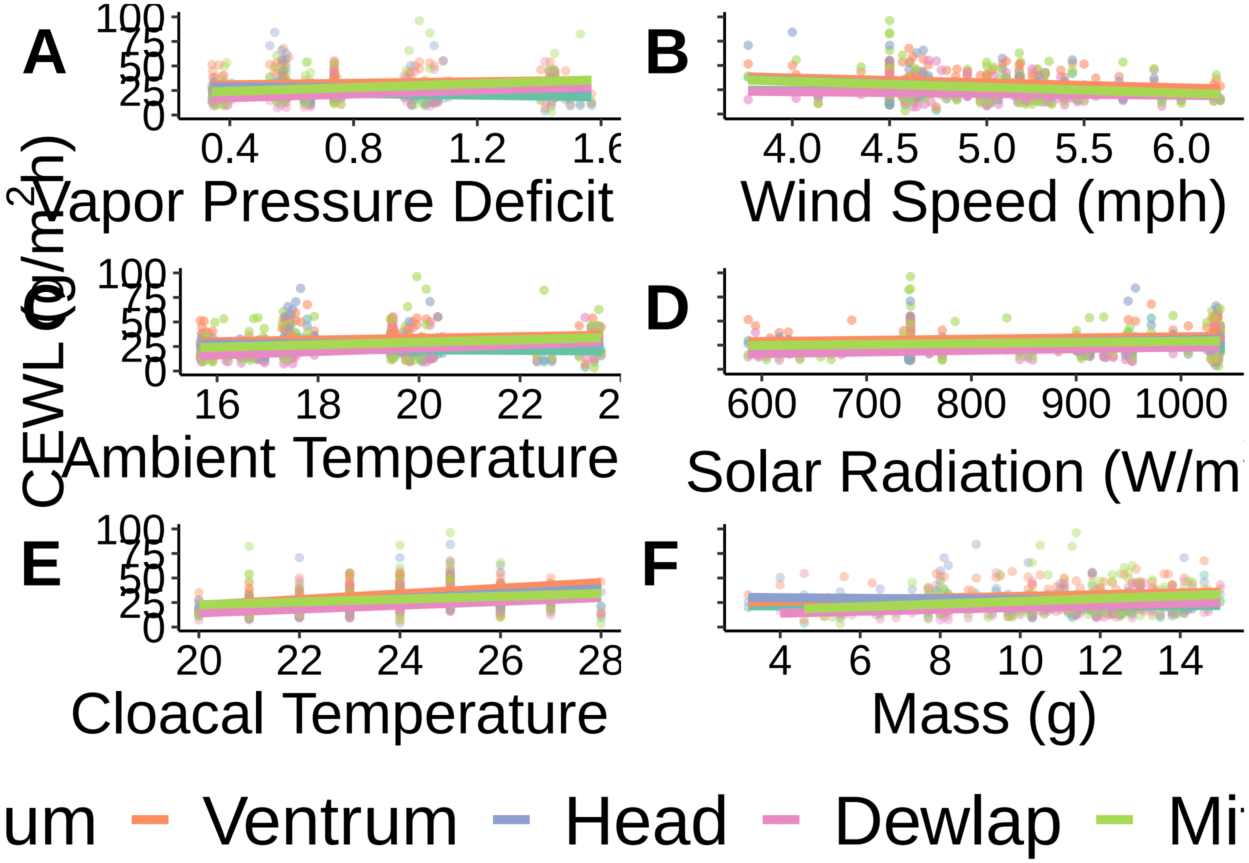
The following variables should be included in the full LMM: - *body region* - *cloacal temperature* at measurement - *capture temp* (and try an interaction with region) - capture *wind speed* and *solar radiation* - measurement temperature (and try an interaction with region) - individual_ID (as a random effect) - *SVL* and *mass* - *hold time*

CEWL Multi-Figure

```
ggarrange(CEWL_vpd_fig, wind_CEWL_plot,
          cap_temp_CEWL, sorad_CEWL_plot,
          CEWL_ctemp_fig, CEWL_mass_fig,
          labels = c("A", "B", "C", "D", "E", "F"),
          font.label = list(size = 24, face = "bold", color ="black"),
          ncol = 2, nrow = 3,
          common.legend = TRUE,
          legend = "bottom"
          ) -> CEWL_multi_fig
```

```
## Warning: Removed 59 rows containing non-finite values (stat_smooth).

## Warning: Removed 59 rows containing missing values (geom_point).

## Warning: Removed 59 rows containing non-finite values (stat_smooth).

## Warning: Removed 59 rows containing missing values (geom_point).

## Warning: Removed 59 rows containing non-finite values (stat_smooth).

## Warning: Removed 59 rows containing missing values (geom_point).

## Warning: Removed 59 rows containing non-finite values (stat_smooth).

## Warning: Removed 59 rows containing missing values (geom_point).

## Warning: Removed 59 rows containing non-finite values (stat_smooth).

## Warning: Removed 59 rows containing missing values (geom_point).

## Warning: Removed 49 rows containing non-finite values (stat_smooth).

## Warning: Removed 49 rows containing missing values (geom_point).

## Warning: Removed 44 rows containing non-finite values (stat_smooth).

## Warning: Removed 44 rows containing missing values (geom_point).
```

```
CEWL_multi_fig
```



```r
# export figure
ggsave(filename = "CEWL_multi_fig.jpeg",
       plot = CEWL_multi_fig,
       path = "./final_figures",
       device = "jpeg",
       dpi = 1200,
       width = 12, height = 14)
```

LMMs

Hydration

Based on the simple linear models and figures above, osmolality should be predicted by sample eye, hemolysis, date/week, individual, capture temperature, and capture VPD.

Prep dataframe for computing models:

```r
hydrat_mod_dat <- morpho_blood_SMI %>%
  dplyr::select(date,
                individual_ID,
                osmolality_mmol_kg,
                temp_C_interpol,
                VPD_kPa_int,
                SMI,
                hemolyzed
                ) %>%
  dplyr::filter(complete.cases(.))
```

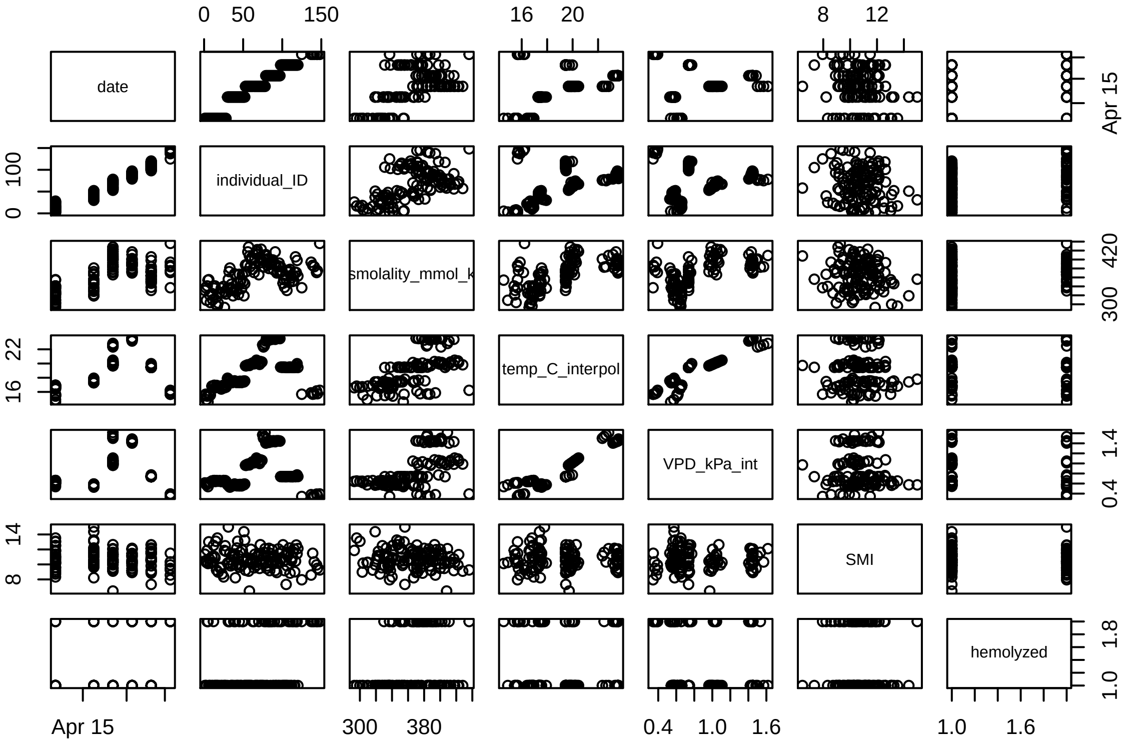
```r
summary(hydrat_mod_dat)
```

```
##       date            individual_ID osmolality_mmol_kg temp_C_interpol
##  Min.   :2021-04-05   1      :  1   Min.   :293.0      Min.   :14.58  
##  1st Qu.:2021-04-19   2      :  1   1st Qu.:338.0      1st Qu.:16.95  
##  Median :2021-04-26   3      :  1   Median :370.0      Median :19.46  
##  Mean   :2021-04-25   4      :  1   Mean   :365.5      Mean   :19.08  
##  3rd Qu.:2021-05-03   5      :  1   3rd Qu.:389.0      3rd Qu.:20.14  
##  Max.   :2021-05-17   6      :  1   Max.   :436.0      Max.   :23.61  
##                       (Other):115                                     
##   VPD_kPa_int          SMI         hemolyzed
##  Min.   :0.3424   Min.   : 6.450   N:82     
##  1st Qu.:0.5907   1st Qu.: 9.929   Y:39     
##  Median :0.7372   Median :10.583            
##  Mean   :0.8509   Mean   :10.714            
##  3rd Qu.:1.0465   3rd Qu.:11.501            
##  Max.   :1.6167   Max.   :14.999            
## 
```

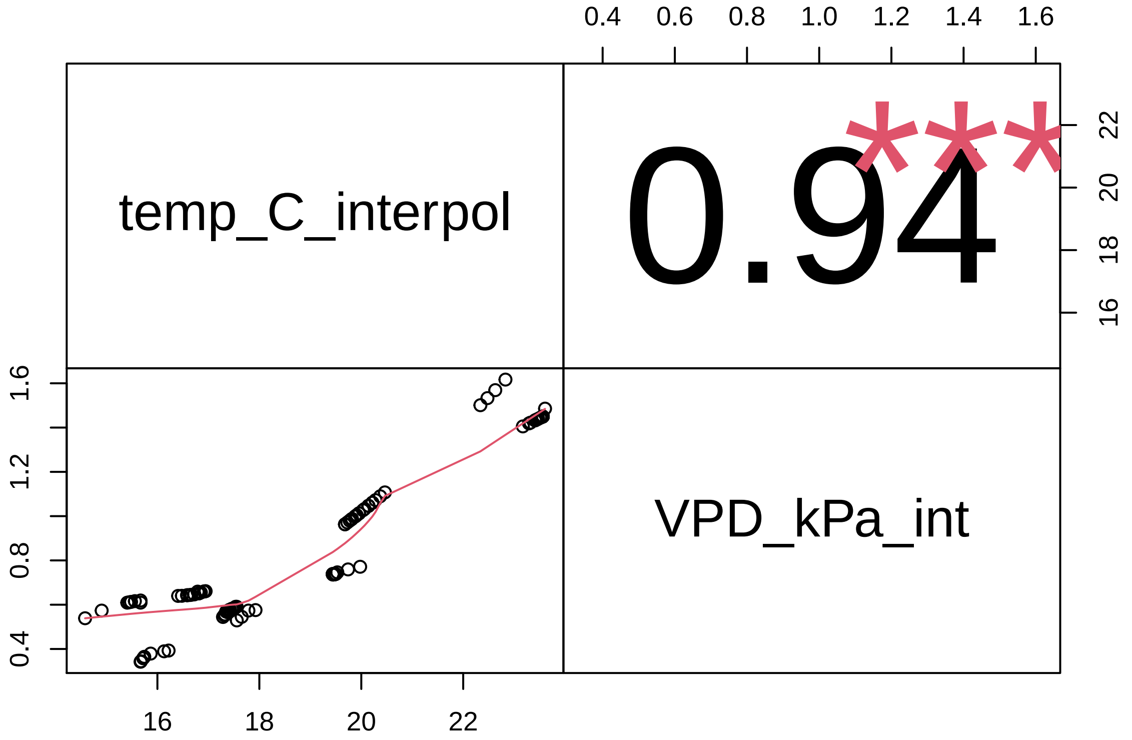
Multicollinearity

First, check for multicollinearity among independent variables:

```r
pairs(hydrat_mod_dat)
```



```r
# also make another plot with r-sq values
# only continuous numeric variables work for this one
hydrat_mod_dat %>%
  # select cont variables
  dplyr::select(temp_C_interpol,
                VPD_kPa_int) %>%
  # multicollinearity plot
  chart.Correlation(., histogram = F, pch = 19)
```



Date and individual_ID are collinear and should not both be used as fixed effects. But, individual_ID is a factor and there is not repetition in the dataset, so that's not feasible as a variable. Temperature and VPD are the only continuous variables, and they are very badly collinear.

Try centering VPD and temp:

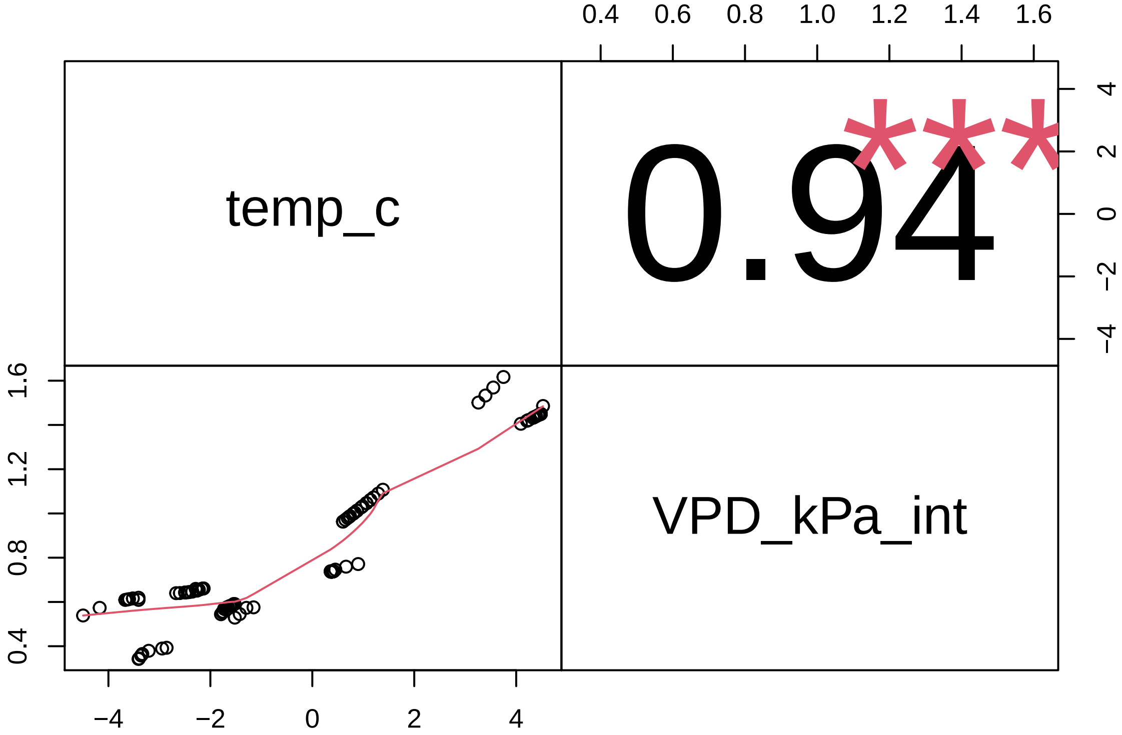
```
hydrat_mod_dat_c <- hydrat_mod_dat %>%
  mutate(VPD_mean = mean(VPD_kPa_int),
         temp_mean = mean(temp_C_interpol),
         VPD_c = VPD_kPa_int - VPD_mean,
         temp_c = temp_C_interpol - temp_mean)
summary(hydrat_mod_dat_c)
```

```
##       date              individual_ID osmolality_mmol_kg temp_C_interpol
##  Min.   :2021-04-05   1      :  1     Min.   :293.0      Min.   :14.58
##  1st Qu.:2021-04-19   2      :  1     1st Qu.:338.0      1st Qu.:16.95
##  Median :2021-04-26   3      :  1     Median :370.0      Median :19.46
##  Mean   :2021-04-25   4      :  1     Mean   :365.5      Mean   :19.08
##  3rd Qu.:2021-05-03   5      :  1     3rd Qu.:389.0      3rd Qu.:20.14
##  Max.   :2021-05-17   6      :  1     Max.   :436.0      Max.   :23.61
##                       (Other):115
##   VPD_kPa_int          SMI          hemolyzed    VPD_mean         temp_mean
##  Min.   :0.3424   Min.   : 6.450   N:82      Min.   :0.8509   Min.   :19.08
##  1st Qu.:0.5907   1st Qu.: 9.929   Y:39      1st Qu.:0.8509   1st Qu.:19.08
##  Median :0.7372   Median :10.583             Median :0.8509   Median :19.08
##  Mean   :0.8509   Mean   :10.714             Mean   :0.8509   Mean   :19.08
##  3rd Qu.:1.0465   3rd Qu.:11.501             3rd Qu.:0.8509   3rd Qu.:19.08
##  Max.   :1.6167   Max.   :14.999             Max.   :0.8509   Max.   :19.08
##
```

```
##      VPD_c               temp_c       
##  Min.   :-0.5085   Min.   :-4.4971  
##  1st Qu.:-0.2603   1st Qu.:-2.1317  
##  Median :-0.1137   Median : 0.3776  
##  Mean   : 0.0000   Mean   : 0.0000  
##  3rd Qu.: 0.1956   3rd Qu.: 1.0623  
##  Max.   : 0.7658   Max.   : 4.5269  
##
```

Test collinearity again:

```
hydrat_mod_dat_c %>%
  # select cont variables
  dplyr::select(temp_c,
                VPD_kPa_int) %>%
  # multicollinearity plot
  chart.Correlation(., histogram = F, pch = 19)
```



Did not help.

Models & Selection

Start with all the variables that were significant individually in SLRs.

```
# model 1
hydrat_mod1 <- lme4::lmer(data = hydrat_mod_dat,
                          # response variable
                          osmolality_mmol_kg ~
                            # start with interaction + singular effect
                           VPD_kPa_int*temp_C_interpol +
                            # other potentially important factor
                            SMI +
                            # random effects
                            (1|date) + (1|hemolyzed))
summary(hydrat_mod1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: osmolality_mmol_kg ~ VPD_kPa_int * temp_C_interpol + SMI + (1 |
##     date) + (1 | hemolyzed)
##    Data: hydrat_mod_dat
##
## REML criterion at convergence: 1031.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.54535 -0.62257  0.06894  0.59500  2.92299
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  date      (Intercept) 831.66   28.839
##  hemolyzed (Intercept)  31.27    5.592
##  Residual              313.35   17.702
## Number of obs: 121, groups:  date, 6; hemolyzed, 2
##
## Fixed effects:
##                              Estimate Std. Error t value
## (Intercept)                  288.6818   119.0957   2.424
## VPD_kPa_int                  142.8030   248.3575   0.575
## temp_C_interpol                3.5783     5.9178   0.605
## SMI                           -0.8692     1.2506  -0.695
## VPD_kPa_int:temp_C_interpol   -5.8006     9.5158  -0.610
##
## Correlation of Fixed Effects:
##             (Intr) VPD_kP_ tmp_C_ SMI
## VPD_kPa_int -0.652
## tmp_C_ntrpl -0.888  0.268
## SMI         -0.061 -0.056  -0.018
## VPD_kP_:_C_  0.734 -0.990  -0.385  0.048
```

```
drop1(hydrat_mod1)
```

```
## Single term deletions
##
## Model:
## osmolality_mmol_kg ~ VPD_kPa_int * temp_C_interpol + SMI + (1 |
##     date) + (1 | hemolyzed)
##                             npar    AIC
## <none>                           1074.6
## SMI                            1 1073.2
## VPD_kPa_int:temp_C_interpol    1 1072.8
```

Dropping terms will not meaningfully improve the model based on AIC…

Re-run model using lmertest to get p-values:

```
hydrat_mod2 <- lmerTest::lmer(data = hydrat_mod_dat,
                          # response variable
                          osmolality_mmol_kg ~
                            # start with interaction + singular effect
```

```
                           VPD_kPa_int*temp_C_interpol +
                            # other potentially important factor
                            SMI +
                            # random effects
                            (1|date) + (1|hemolyzed))
summary(hydrat_mod2)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: osmolality_mmol_kg ~ VPD_kPa_int * temp_C_interpol + SMI + (1 |
##     date) + (1 | hemolyzed)
##    Data: hydrat_mod_dat
##
## REML criterion at convergence: 1031.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.54535 -0.62257  0.06894  0.59500  2.92299
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  date      (Intercept) 831.66   28.839
##  hemolyzed (Intercept)  31.27    5.592
##  Residual              313.35   17.702
## Number of obs: 121, groups:  date, 6; hemolyzed, 2
##
## Fixed effects:
##                             Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)                 288.6818   119.0957  18.7788   2.424   0.0256 *
## VPD_kPa_int                 142.8030   248.3575   8.8915   0.575   0.5796
## temp_C_interpol               3.5783     5.9178  53.2602   0.605   0.5480
## SMI                          -0.8692     1.2506 112.5191  -0.695   0.4885
## VPD_kPa_int:temp_C_interpol  -5.8006     9.5158   9.4498  -0.610   0.5565
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) VPD_kP_ tmp_C_ SMI
## VPD_kPa_int -0.652
## tmp_C_ntrpl -0.888  0.268
## SMI         -0.061 -0.056  -0.018
## VPD_kP_:_C_  0.734 -0.990  -0.385  0.048
```

Best Model

```
# save model 2 summary object
osml_best_mod <- broom.mixed::tidy(hydrat_mod2)
```

```
## Registered S3 method overwritten by 'broom.mixed':
##   method      from
##   tidy.gamlss broom
```

```
# export
write.csv(osml_best_mod, "./best models/osml_best_mod_vals.csv")
```

Check LM Assumptions (Hydration Model)

First, get residuals:

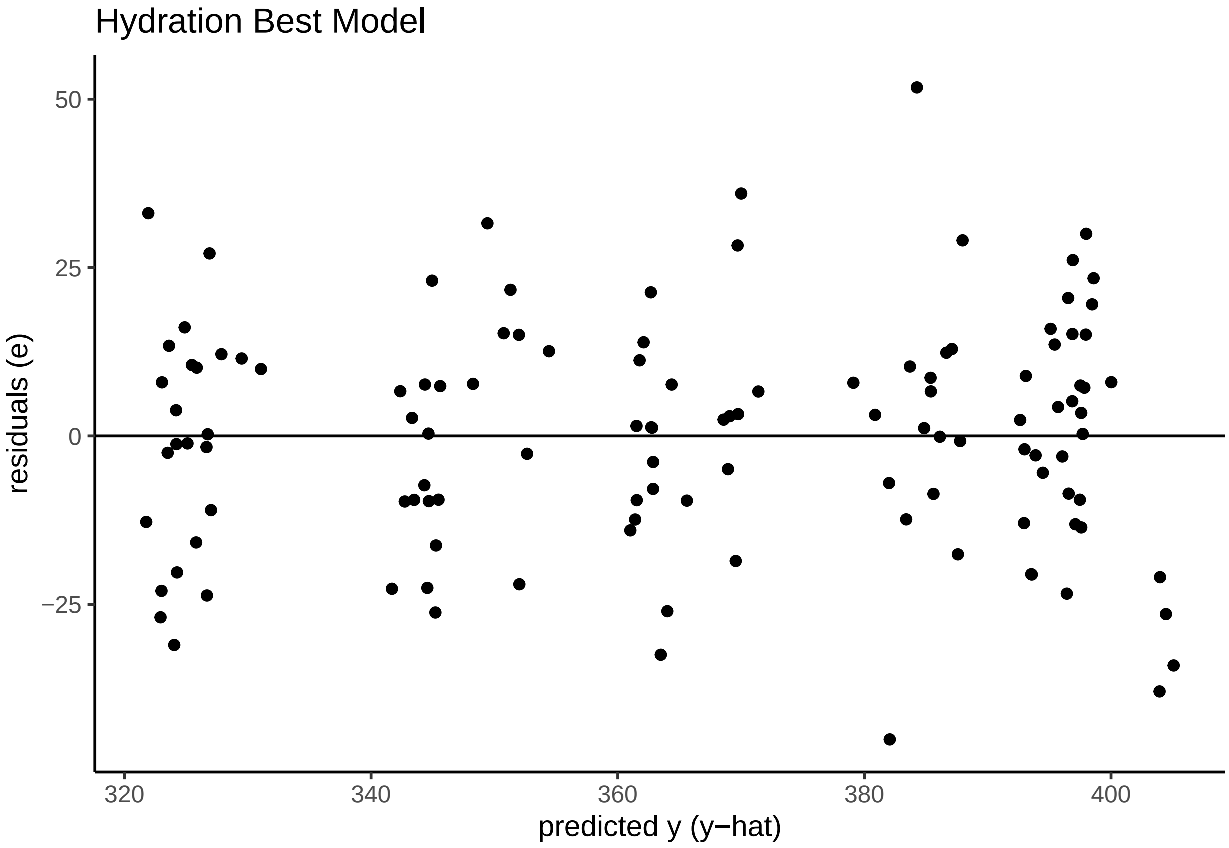
```
res_hydrat_mod <- hydrat_mod_dat %>%
  mutate(y_hat = predict(hydrat_mod2),
         e = residuals(hydrat_mod2))
```

Linearity and Equal Variance

Is the function **linear**? Is there **equal** variance of the residuals? The residuals should be homoskedactic relative to y_hat (or x). We don't care if there is a relationship between the residuals ~ dependent variable (actual y).

Plotting residuals shows us whether the data meets linearity and equal variance assumptions:

```
ggplot(data = res_hydrat_mod, aes(x = y_hat, y = e)) +
  geom_point() +
  theme_classic() +
  xlab("predicted y (y-hat)") +
  ylab("residuals (e)") +
  ggtitle("Hydration Best Model") +
  geom_hline(yintercept = 0)
```



It looks pretty evenly distributed. No clear patterning, so linearity and equal error variance should be satisfied.

Brown-Forsythe test to statistically check equal variance:

H0: normally distributed (non-sig test is GOOD) HA: NOT normally distributed (reject nul == assumption not satisfied)

```
# need to create the right data & format first
bf_data_temp <- res_hydrat_mod %>%
  dplyr::filter(complete.cases(temp_C_interpol)) %>%
  dplyr::mutate(middle = median(temp_C_interpol),
                side = temp_C_interpol > middle)
bf_data_temp$side <- as.factor(bf_data_temp$side)

# now run test
bf.test(formula = e ~ side, # y~x
        data = bf_data_temp, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

```
##
##   Brown-Forsythe Test (alpha = 0.05)
## -------------------------------------------------------------
##   data : e and side
##
##   statistic  : 0.03717374
##   num df     : 1
##   denom df   : 117.8096
##   p.value    : 0.8474437
##
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

```
# need to create the right data & format first
bf_data_vpd <- res_hydrat_mod %>%
  dplyr::filter(complete.cases(VPD_kPa_int)) %>%
  dplyr::mutate(middle = median(VPD_kPa_int),
                side = VPD_kPa_int > middle)
bf_data_vpd$side <- as.factor(bf_data_vpd$side)

# now run test
bf.test(formula = e ~ side, # y~x
        data = bf_data_vpd, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

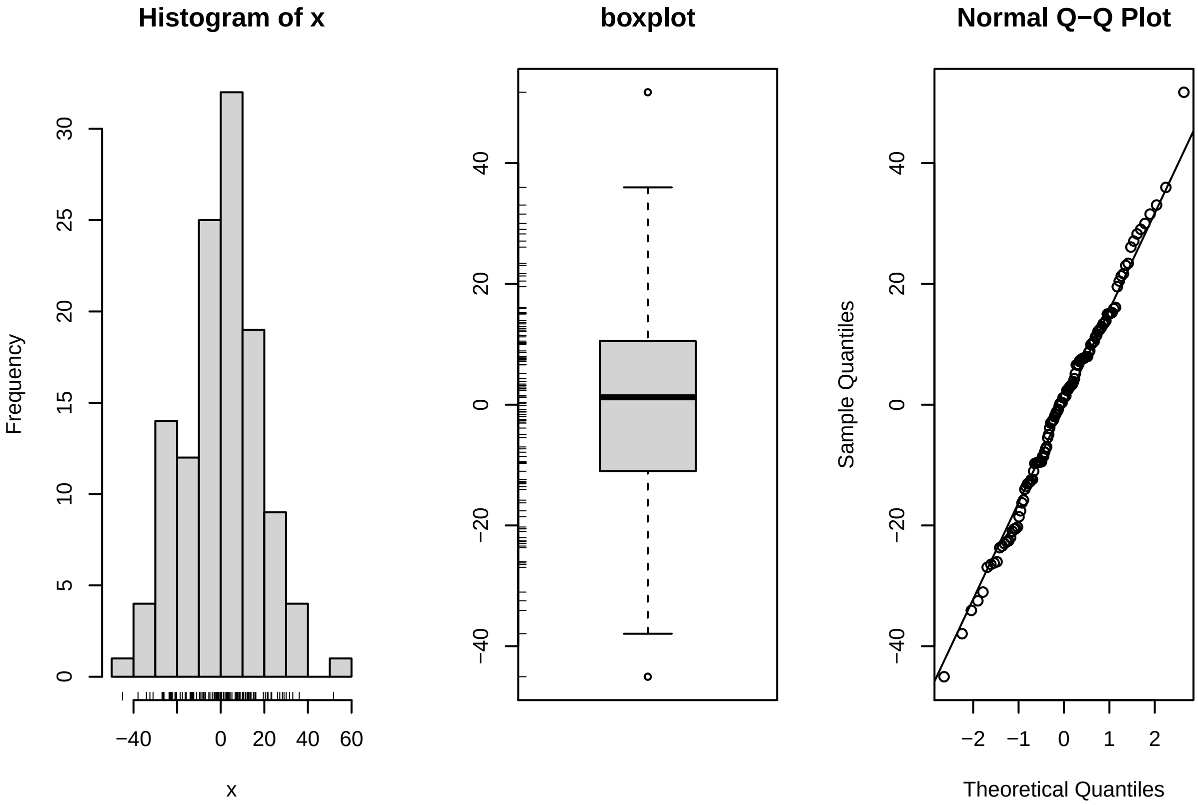
```
##
##   Brown-Forsythe Test (alpha = 0.05)
## -------------------------------------------------------------
##   data : e and side
##
##   statistic  : 0.008988726
##   num df     : 1
##   denom df   : 118.073
##   p.value    : 0.9246273
##
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

Equal variance is satisfied.

Now check normality. Is the distribution of residuals **normal**?

use Shapiro-Wilk normality test: H0: data is NOT significantly different from normal distribution HA: data IS significantly different from normal distribution

```
simple.eda(res_hydrat_mod$e)
```



```
shapiro.test(res_hydrat_mod$e)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res_hydrat_mod$e
## W = 0.99444, p-value = 0.9173
```

Normality is satisfied.

Conclusion

hydrat_mod2 is the best model to predict osmolality, and the model satisfies all linear regression conditions/assumptions.

CEWL

Based on the simple linear models and figures above, CEWL should be predicted by: - body region - cloacal temperature at measurement - capture temperature, VPD, wind speed, and solar radiation - ambient temperature during measurement - SVL and mass - hold time (time between capture vs measurement) - individual ID (as a random effect)

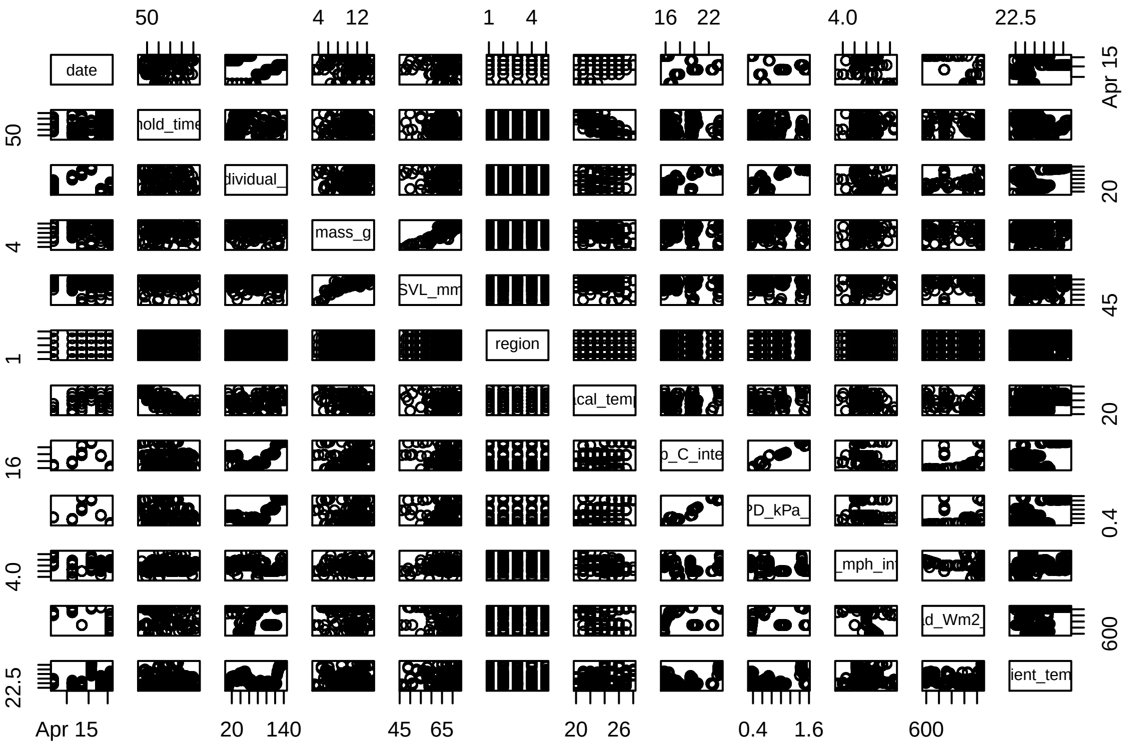
Prep dataframe for models:

```
CEWL_mod_dat <- CEWL_data_full %>%
  # select variables of interest
  dplyr::select(date,
                hold_time,
                individual_ID,
                mass_g,
                SVL_mm,
                TEWL_g_m2h,
                region,
                cloacal_temp_C,
                temp_C_interpol,
                VPD_kPa_int,
                Wind_mph_interpol,
                Solar_rad_Wm2_interpol,
                ambient_temp_C
                ) %>%
  dplyr::filter(complete.cases(.)) # removes 69 observations
```

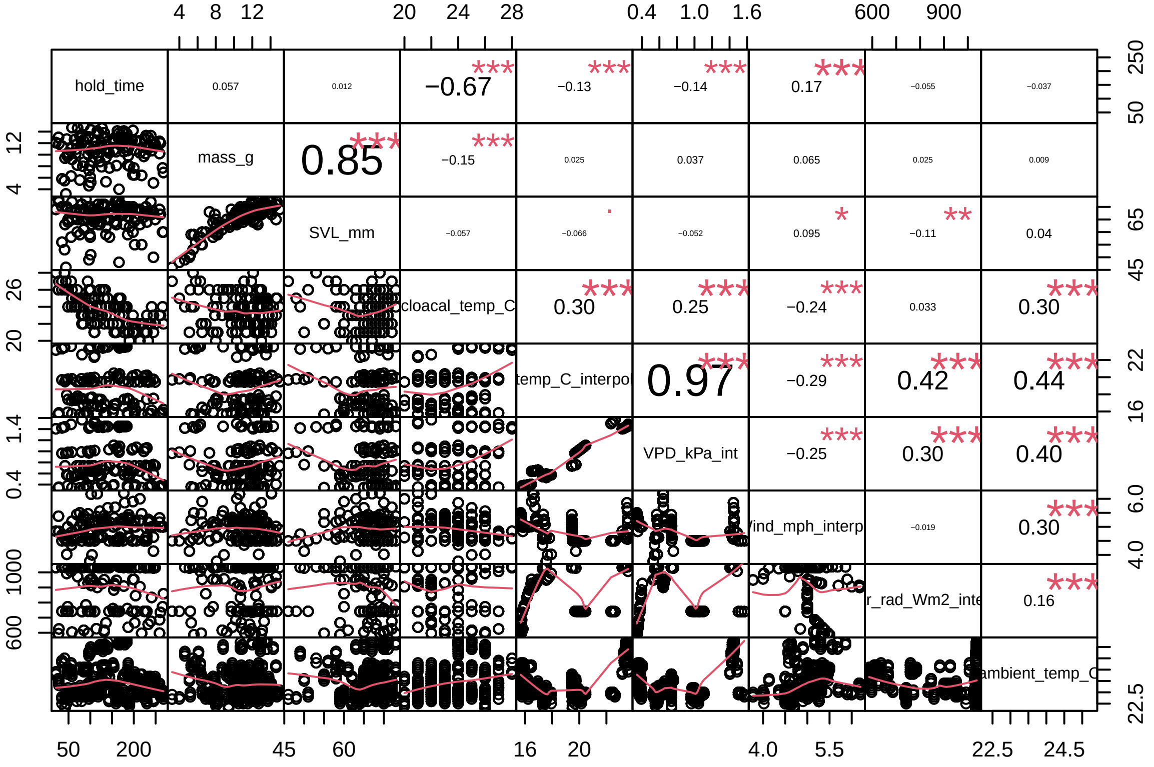
Multicollinearity

Check for multicollinearity among independent variables:

```
CEWL_mod_dat %>%
  # get rid of dependent variable
  dplyr::select(-TEWL_g_m2h, individual_ID) %>%
  # multicollinearity plot
  pairs(.)
```



```
# also make another plot with r-sq values
# non-numeric variables don't work for this
CEWL_mod_dat %>%
  # select variables of interest
  dplyr::select(-TEWL_g_m2h, -date, -region, -individual_ID) %>%
  # multicollinearity plot
  chart.Correlation(., histogram = F, pch = 19)
```



Mass and SVL are very collinear variables that should not be used in combination. Cloacal temp and hold time are pretty collinear, which makes sense; as we held them, they got cooler because they were inside and not basking. Individual ID and VPD are also pretty collinear, but it's less intuitive. Temperature and solar radiation, as well as VPD and solar radiation, are each collinear pairs. We will use model selection to figure out which variable from each collinear pair is better to include in the model. Temperature and VPD are very badly collinear… But as I tried above, centering one or both did not help.

Models & Selection

This is the full model with every potential variable and interaction indicated to be meaningful based on figures and SLRs.

```
# model 1
CEWL_mod1 <- lme4::lmer(data = CEWL_mod_dat,
                        TEWL_g_m2h ~
                         region * (mass_g + ambient_temp_C +
                                     temp_C_interpol +
                                     VPD_kPa_int) +
                         hold_time + SVL_mm +
                         cloacal_temp_C +
                         temp_C_interpol:VPD_kPa_int +
                         Wind_mph_interpol +
                         Solar_rad_Wm2_interpol +
                         (1|individual_ID))
summary(CEWL_mod1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: TEWL_g_m2h ~ region * (mass_g + ambient_temp_C + temp_C_interpol +
##     VPD_kPa_int) + hold_time + SVL_mm + cloacal_temp_C + temp_C_interpol:VPD_kPa_int +
##     Wind_mph_interpol + Solar_rad_Wm2_interpol + (1 | individual_ID)
##    Data: CEWL_mod_dat
##
## REML criterion at convergence: 4704.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.2012 -0.5498 -0.1069  0.4079  5.2081
##
## Random effects:
##  Groups        Name        Variance Std.Dev.
##  individual_ID (Intercept) 29.44    5.426
##  Residual                  95.47    9.771
## Number of obs: 630, groups:  individual_ID, 128
##
## Fixed effects:
##                                    Estimate Std. Error t value
## (Intercept)                      -62.623740  55.590335  -1.127
## regionVentrum                     26.021842  45.113741   0.577
## regionHead                         4.485975  45.298933   0.099
## regionDewlap                     -82.879947  46.148074  -1.796
## regionMite Patch                 115.221488  46.039027   2.503
## mass_g                             0.112632   0.586454   0.192
## ambient_temp_C                    -1.056817   2.066328  -0.511
## temp_C_interpol                    2.758655   1.951301   1.414
## VPD_kPa_int                       -4.529698  29.662032  -0.153
## hold_time                          0.007260   0.014290   0.508
## SVL_mm                             0.121833   0.244732   0.498
## cloacal_temp_C                     2.486388   0.565504   4.397
## Wind_mph_interpol                 -0.532971   1.851924  -0.288
## Solar_rad_Wm2_interpol             0.009751   0.005491   1.776
## regionVentrum:mass_g               0.897131   0.482177   1.861
## regionHead:mass_g                 -0.299643   0.478622  -0.626
## regionDewlap:mass_g                1.000705   0.488917   2.047
## regionMite Patch:mass_g            1.161157   0.495413   2.344
## regionVentrum:ambient_temp_C      -1.976251   1.899868  -1.040
## regionHead:ambient_temp_C         -0.566364   1.905627  -0.297
## regionDewlap:ambient_temp_C        1.879978   1.934356   0.972
## regionMite Patch:ambient_temp_C   -2.427163   1.925730  -1.260
## regionVentrum:temp_C_interpol      0.999062   2.019771   0.495
## regionHead:temp_C_interpol         1.108260   2.020269   0.549
## regionDewlap:temp_C_interpol       1.291461   2.019807   0.639
## regionMite Patch:temp_C_interpol  -5.747061   2.022081  -2.842
## regionVentrum:VPD_kPa_int          2.628229  13.690220   0.192
## regionHead:VPD_kPa_int            -3.883782  13.692489  -0.284
## regionDewlap:VPD_kPa_int           3.654348  13.705967   0.267
## regionMite Patch:VPD_kPa_int      54.851651  13.726802   3.996
## temp_C_interpol:VPD_kPa_int       -0.855093   1.196028  -0.715
##
## Correlation matrix not shown by default, as p = 31 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

Check which variables to drop:

```
drop1(CEWL_mod1)
```

```
## Single term deletions
##
## Model:
## TEWL_g_m2h ~ region * (mass_g + ambient_temp_C + temp_C_interpol +
##     VPD_kPa_int) + hold_time + SVL_mm + cloacal_temp_C + temp_C_interpol:VPD_kPa_int +
##     Wind_mph_interpol + Solar_rad_Wm2_interpol + (1 | individual_ID)
##                              npar    AIC
## <none>                            4812.0
## hold_time                       1 4810.3
## SVL_mm                          1 4810.3
## cloacal_temp_C                  1 4829.7
## Wind_mph_interpol               1 4810.1
## Solar_rad_Wm2_interpol          1 4813.4
## region:mass_g                   4 4819.1
## region:ambient_temp_C           4 4810.5
## region:temp_C_interpol          4 4821.8
## region:VPD_kPa_int              4 4829.6
## temp_C_interpol:VPD_kPa_int     1 4810.6
```

Based on AIC, dropping SVL, hold time, wind speed, region:ambient_temp_C, and temp_C_interpol:VPD_kPa_int would result in a better model.

Start with region:ambient_temp_C and temp_C_interpol:VPD_kPa_int interactions:

```
# model 2
CEWL_mod2 <- lme4::lmer(data = CEWL_mod_dat,
                        TEWL_g_m2h ~
                         region * (mass_g +
                                     temp_C_interpol +
                                     VPD_kPa_int) +
                         hold_time + SVL_mm +
                         cloacal_temp_C +
                         Wind_mph_interpol +
                         Solar_rad_Wm2_interpol +
                         (1|individual_ID))
summary(CEWL_mod2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: TEWL_g_m2h ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
##     hold_time + SVL_mm + cloacal_temp_C + Wind_mph_interpol +
##     Solar_rad_Wm2_interpol + (1 | individual_ID)
##    Data: CEWL_mod_dat
##
## REML criterion at convergence: 4731.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.1586 -0.5691 -0.1021  0.3923  5.3758
##
## Random effects:
```

```
##  Groups        Name        Variance Std.Dev.
##  individual_ID (Intercept) 30.37    5.511
##  Residual                  96.00    9.798
## Number of obs: 630, groups:  individual_ID, 128
##
## Fixed effects:
##                                    Estimate Std. Error t value
## (Intercept)                      -47.039209  27.787661  -1.693
## regionVentrum                    -10.283730  27.201174  -0.378
## regionHead                        -5.560778  27.233488  -0.204
## regionDewlap                     -46.552931  27.353675  -1.702
## regionMite Patch                  68.695412  27.391991   2.508
## mass_g                             0.188124   0.590585   0.319
## temp_C_interpol                    1.655662   1.836532   0.902
## VPD_kPa_int                      -20.567866  12.193793  -1.687
## hold_time                          0.002290   0.014162   0.162
## SVL_mm                             0.076642   0.246360   0.311
## cloacal_temp_C                     2.109902   0.537107   3.928
## Wind_mph_interpol                 -2.680219   1.540994  -1.739
## Solar_rad_Wm2_interpol             0.011840   0.005456   2.170
## regionVentrum:mass_g               0.862345   0.483200   1.785
## regionHead:mass_g                 -0.307348   0.479892  -0.640
## regionDewlap:mass_g                0.996438   0.490248   2.033
## regionMite Patch:mass_g            1.206684   0.496335   2.431
## regionVentrum:temp_C_interpol      0.369782   1.951904   0.189
## regionHead:temp_C_interpol         0.900094   1.952275   0.461
## regionDewlap:temp_C_interpol       1.788876   1.955837   0.915
## regionMite Patch:temp_C_interpol  -6.442226   1.957873  -3.290
## regionVentrum:VPD_kPa_int          5.346806  13.553514   0.394
## regionHead:VPD_kPa_int            -2.919353  13.553480  -0.215
## regionDewlap:VPD_kPa_int           1.747045  13.589702   0.129
## regionMite Patch:VPD_kPa_int      57.376098  13.622254   4.212
```

```
##
## Correlation matrix not shown by default, as p = 25 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

```
# compare
anova(CEWL_mod2, CEWL_mod1)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: CEWL_mod_dat
## Models:
## CEWL_mod2: TEWL_g_m2h ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
## CEWL_mod2:     hold_time + SVL_mm + cloacal_temp_C + Wind_mph_interpol +
## CEWL_mod2:     Solar_rad_Wm2_interpol + (1 | individual_ID)
## CEWL_mod1: TEWL_g_m2h ~ region * (mass_g + ambient_temp_C + temp_C_interpol +
## CEWL_mod1:     VPD_kPa_int) + hold_time + SVL_mm + cloacal_temp_C + temp_C_interpol:VPD_kPa_int +
## CEWL_mod1:     Wind_mph_interpol + Solar_rad_Wm2_interpol + (1 | individual_ID)
##           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## CEWL_mod2   27 4811.8 4931.9 -2378.9   4757.8
## CEWL_mod1   33 4812.0 4958.7 -2373.0   4746.0 11.792  6    0.06677 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

AIC improved somewhat and we do not lose a significant amount of predictive power using alpha 0.05, so the reduced model is better. Continue dropping…

Check drop terms again:

```
drop1(CEWL_mod2)
```

```
## Single term deletions
##
## Model:
## TEWL_g_m2h ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
##     hold_time + SVL_mm + cloacal_temp_C + Wind_mph_interpol +
##     Solar_rad_Wm2_interpol + (1 | individual_ID)
##                        npar    AIC
## <none>                      4811.8
## hold_time                 1 4809.9
## SVL_mm                    1 4809.9
## cloacal_temp_C            1 4825.5
## Wind_mph_interpol         1 4813.0
## Solar_rad_Wm2_interpol    1 4814.8
## region:mass_g             4 4819.0
## region:temp_C_interpol    4 4826.6
## region:VPD_kPa_int        4 4831.6
```

Hold time and SVL should still be deleted.

```
# model 3
CEWL_mod3 <- lme4::lmer(data = CEWL_mod_dat,
                        TEWL_g_m2h ~
                         region * (mass_g +
                                     temp_C_interpol +
                                     VPD_kPa_int) +
                         cloacal_temp_C +
                         Wind_mph_interpol +
                         Solar_rad_Wm2_interpol +
                         (1|individual_ID))
summary(CEWL_mod3)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: TEWL_g_m2h ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
##     cloacal_temp_C + Wind_mph_interpol + Solar_rad_Wm2_interpol +
##     (1 | individual_ID)
##    Data: CEWL_mod_dat
##
## REML criterion at convergence: 4724.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.1748 -0.5697 -0.1018  0.3989  5.3880
##
## Random effects:
##  Groups        Name        Variance Std.Dev.
##  individual_ID (Intercept) 29.62    5.442
##  Residual                  95.99    9.797
## Number of obs: 630, groups:  individual_ID, 128
##
```

```
## Fixed effects:
##                                    Estimate Std. Error t value
## (Intercept)                      -43.381020  25.621513  -1.693
## regionVentrum                    -10.298846  27.199558  -0.379
## regionHead                        -5.575132  27.231768  -0.205
## regionDewlap                     -46.522134  27.351965  -1.701
## regionMite Patch                  68.760070  27.389966   2.510
## mass_g                             0.323085   0.393521   0.821
## temp_C_interpol                    1.756118   1.756816   1.000
## VPD_kPa_int                      -21.244263  11.749943  -1.808
## cloacal_temp_C                     2.068132   0.373125   5.543
## Wind_mph_interpol                 -2.606043   1.512285  -1.723
## Solar_rad_Wm2_interpol             0.011280   0.005112   2.207
## regionVentrum:mass_g               0.863129   0.483096   1.787
## regionHead:mass_g                 -0.306699   0.479806  -0.639
## regionDewlap:mass_g                0.994309   0.490169   2.029
## regionMite Patch:mass_g            1.203405   0.496201   2.425
## regionVentrum:temp_C_interpol      0.370323   1.951820   0.190
## regionHead:temp_C_interpol         0.900670   1.952186   0.461
## regionDewlap:temp_C_interpol       1.788339   1.955746   0.914
## regionMite Patch:temp_C_interpol  -6.444419   1.957777  -3.292
## regionVentrum:VPD_kPa_int          5.341313  13.552865   0.394
## regionHead:VPD_kPa_int            -2.924713  13.552834  -0.216
## regionDewlap:VPD_kPa_int           1.750500  13.589053   0.129
## regionMite Patch:VPD_kPa_int      57.394210  13.621513   4.213
```

```
##
## Correlation matrix not shown by default, as p = 23 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

```r
# compare
anova(CEWL_mod3, CEWL_mod1)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: CEWL_mod_dat
## Models:
## CEWL_mod3: TEWL_g_m2h ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
## CEWL_mod3:     cloacal_temp_C + Wind_mph_interpol + Solar_rad_Wm2_interpol +
## CEWL_mod3:     (1 | individual_ID)
## CEWL_mod1: TEWL_g_m2h ~ region * (mass_g + ambient_temp_C + temp_C_interpol +
## CEWL_mod1:     VPD_kPa_int) + hold_time + SVL_mm + cloacal_temp_C + temp_C_interpol:VPD_kPa_int +
## CEWL_mod1:     Wind_mph_interpol + Solar_rad_Wm2_interpol + (1 | individual_ID)
##           npar  AIC    BIC logLik deviance  Chisq Df Pr(>Chisq)
## CEWL_mod3   25 4808 4919.1  -2379     4758
## CEWL_mod1   33 4812 4958.7  -2373     4746 11.927  8     0.1545
```

```r
anova(CEWL_mod3, CEWL_mod2)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: CEWL_mod_dat
## Models:
## CEWL_mod3: TEWL_g_m2h ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
## CEWL_mod3:     cloacal_temp_C + Wind_mph_interpol + Solar_rad_Wm2_interpol +
## CEWL_mod3:     (1 | individual_ID)
## CEWL_mod2: TEWL_g_m2h ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
## CEWL_mod2:     hold_time + SVL_mm + cloacal_temp_C + Wind_mph_interpol +
## CEWL_mod2:     Solar_rad_Wm2_interpol + (1 | individual_ID)
##           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## CEWL_mod3   25 4808.0 4919.1 -2379.0   4758.0
## CEWL_mod2   27 4811.8 4931.9 -2378.9   4757.8 0.1347  2     0.9349
```

Once again, the AIC is slightly lower, and our model fit is not significantly worse by going to model 3, so the reduced model is better.

Check drop terms again to see if anything else could potentially be removed:

```
drop1(CEWL_mod3)
```

```
## Single term deletions
##
## Model:
## TEWL_g_m2h ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
##     cloacal_temp_C + Wind_mph_interpol + Solar_rad_Wm2_interpol +
##     (1 | individual_ID)
##                        npar    AIC
## <none>                      4808.0
## cloacal_temp_C            1 4834.9
## Wind_mph_interpol         1 4809.1
## Solar_rad_Wm2_interpol    1 4811.0
## region:mass_g             4 4815.1
## region:temp_C_interpol    4 4822.7
## region:VPD_kPa_int        4 4827.7
```

We have arrived at the best model based on AIC.

Since we dropped a lot of variables, I want to go back and make sure to use as much data as possible in the final model, so I can add back in observations with missing values for variables no longer included in the reduced model.

```
# filter
CEWL_mod_dat2 <- CEWL_data_full %>%
  # select variables of interest
  dplyr::select(individual_ID,
                mass_g,
                TEWL_g_m2h,
                region,
                cloacal_temp_C,
                temp_C_interpol,
                VPD_kPa_int,
                Wind_mph_interpol,
                Solar_rad_Wm2_interpol,
                ) %>%
  dplyr::filter(complete.cases(.)) # able to have 5 more observations

# redo best model

# model 3
CEWL_mod3_adj <- lmerTest::lmer(data = CEWL_mod_dat2,
                        TEWL_g_m2h ~
                         region * (mass_g +
                                    temp_C_interpol +
                                     VPD_kPa_int) +
                         cloacal_temp_C +
                         Wind_mph_interpol +
                         Solar_rad_Wm2_interpol +
                         (1|individual_ID))
summary(CEWL_mod3_adj)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: TEWL_g_m2h ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
##     cloacal_temp_C + Wind_mph_interpol + Solar_rad_Wm2_interpol +
##     (1 | individual_ID)
##    Data: CEWL_mod_dat2
##
## REML criterion at convergence: 4759.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.1879 -0.5721 -0.1016  0.4071  5.4153
##
## Random effects:
##  Groups        Name        Variance Std.Dev.
##  individual_ID (Intercept) 29.43    5.425
##  Residual                  95.45    9.770
## Number of obs: 635, groups:  individual_ID, 129
##
## Fixed effects:
##                                    Estimate Std. Error         df t value
## (Intercept)                      -42.378229  25.131551 343.787553  -1.686
## regionVentrum                     -8.218142  26.710581 488.908884  -0.308
## regionHead                        -2.806060  26.741643 489.200900  -0.105
## regionDewlap                     -45.251113  26.851641 490.430562  -1.685
## regionMite Patch                  66.729016  26.884494 489.245020   2.482
## mass_g                             0.318174   0.391483 500.630064   0.813
## temp_C_interpol                    1.722317   1.732287 377.283562   0.994
## VPD_kPa_int                      -21.020190  11.594518 416.139702  -1.813
## cloacal_temp_C                     2.043544   0.369246 119.807395   5.534
## Wind_mph_interpol                 -2.617886   1.507455 118.941200  -1.737
## Solar_rad_Wm2_interpol             0.011454   0.005087 119.058285   2.251
## regionVentrum:mass_g               0.851005   0.480955 493.752878   1.769
## regionHead:mass_g                 -0.322479   0.477675 492.340391  -0.675
## regionDewlap:mass_g                0.986513   0.487880 497.800808   2.022
## regionMite Patch:mass_g            1.217770   0.493838 493.400398   2.466
## regionVentrum:temp_C_interpol      0.231907   1.920942 488.533912   0.121
## regionHead:temp_C_interpol         0.716331   1.921218 488.608438   0.373
## regionDewlap:temp_C_interpol       1.704093   1.924330 489.248337   0.886
## regionMite Patch:temp_C_interpol  -6.310994   1.926114 488.733058  -3.277
## regionVentrum:VPD_kPa_int          6.231356  13.363814 488.573433   0.466
## regionHead:VPD_kPa_int            -1.743219  13.363780 488.571124  -0.130
## regionDewlap:VPD_kPa_int           2.292340  13.396519 489.558266   0.171
## regionMite Patch:VPD_kPa_int      56.527834  13.426716 488.958765   4.210
##                                  Pr(>|t|)
## (Intercept)                       0.09265 .
## regionVentrum                     0.75846
## regionHead                        0.91647    
## regionDewlap                      0.09258 .  
## regionMite Patch                  0.01340 *  
## mass_g                            0.41675    
## temp_C_interpol                   0.32074    
## VPD_kPa_int                       0.07056 .  
## cloacal_temp_C                   1.87e-07 ***
## Wind_mph_interpol                 0.08504 .  
## Solar_rad_Wm2_interpol            0.02619 *  
## regionVentrum:mass_g              0.07744 .  
## regionHead:mass_g                 0.49993    
## regionDewlap:mass_g               0.04371 *  
## regionMite Patch:mass_g           0.01400 *  
## regionVentrum:temp_C_interpol     0.90396    
## regionHead:temp_C_interpol        0.70942    
## regionDewlap:temp_C_interpol      0.37629    
## regionMite Patch:temp_C_interpol  0.00113 ** 
## regionVentrum:VPD_kPa_int         0.64122    
## regionHead:VPD_kPa_int            0.89627    
## regionDewlap:VPD_kPa_int          0.86420    
## regionMite Patch:VPD_kPa_int     3.04e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## 
## Correlation matrix not shown by default, as p = 23 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

Best Model

The best model is CEWL predicted by: body region and its interaction with mass and capture temp and VPD, cloacal temperature, solar radiation at capture time, and individual ID as a random effect.

```r
# save best model info
CEWL_best_mod <- broom.mixed::tidy(CEWL_mod3_adj)
# export
#write.csv(CEWL_best_mod, "./best models/CEWL_best_mod_vals.csv")
```

Check LM Assumptions

First, get residuals:

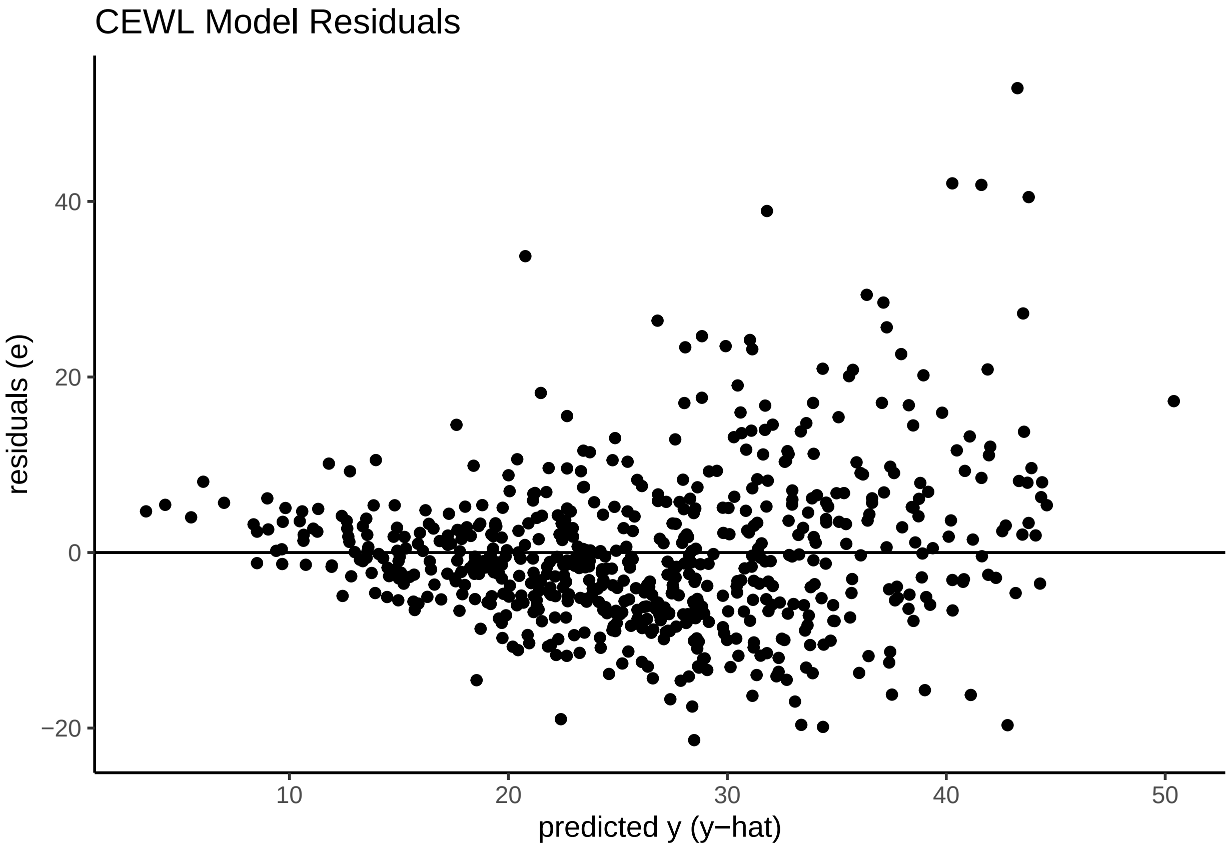
```r
CEWL_mod_res <- CEWL_mod_dat2 %>%
  mutate(y_hat = predict(CEWL_mod3_adj),
         e = residuals(CEWL_mod3_adj))
```

Linearity and Equal Variance

Is the function **linear**? Is there **equal** variance of the residuals? The residuals should be homoskedactic relative to y_hat (or x). We don't care if there is a relationship between the residuals ~ dependent variable (actual y).

Plotting residuals shows us whether the data meets linearity and equal variance assumptions:

```r
ggplot(data = CEWL_mod_res, aes(x = y_hat, y = e)) +
  geom_point() +
  theme_classic() +
  xlab("predicted y (y-hat)") +
  ylab("residuals (e)") +
  ggtitle("CEWL Model Residuals") +
  geom_hline(yintercept = 0)
```



It's definitely making a fan shape. :(

Brown-Forsythe test to statistically check equal variance, for each continuous predictor variable:

H0: normally distributed (non-sig test is GOOD) HA: NOT normally distributed (reject nul == assumption not satisfied)

```
# need to create the right data & format first
bf_data_CEWL <- CEWL_mod_res %>%
  dplyr::mutate(middle_mass = median(mass_g), # mass
                side_mass = as.factor(mass_g > middle_mass),
                # solar radiation
                middle_sorad = median(Solar_rad_Wm2_interpol),
                side_sorad = as.factor(Solar_rad_Wm2_interpol > middle_sorad),
                # temperature
                middle_temp = median(temp_C_interpol),
                side_temp = as.factor(temp_C_interpol > middle_temp),
                # VPD
                middle_absh = median(VPD_kPa_int),
                side_absh = as.factor(VPD_kPa_int > middle_absh),
                middle_ct = median(cloacal_temp_C),
                side_ct = as.factor(cloacal_temp_C > middle_ct)
                )
```

```r
# now run test
bf.test(formula = e ~ side_absh, # y~x
        data = bf_data_CEWL, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

```
##
##   Brown-Forsythe Test (alpha = 0.05)
## -------------------------------------------------------------
##   data : e and side_absh
##
##   statistic  : 0.01899694
##   num df     : 1
##   denom df   : 615.9257
##   p.value    : 0.8904203
##
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

```r
bf.test(formula = e ~ side_temp, # y~x
        data = bf_data_CEWL, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

```
##
##   Brown-Forsythe Test (alpha = 0.05)
## -------------------------------------------------------------
##   data : e and side_temp
##
##   statistic  : 0.003577622
##   num df     : 1
##   denom df   : 581.4948
##   p.value    : 0.9523249
##
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

```r
bf.test(formula = e ~ side_sorad, # y~x
        data = bf_data_CEWL, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

```
##
##   Brown-Forsythe Test (alpha = 0.05)
## -------------------------------------------------------------
##   data : e and side_sorad
##
##   statistic  : 0.004690308
##   num df     : 1
##   denom df   : 630.4428
##   p.value    : 0.9454206
##
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

```r
bf.test(formula = e ~ side_ct, # y~x
        data = bf_data_CEWL, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

```
##
##   Brown-Forsythe Test (alpha = 0.05)
## -------------------------------------------------------------
##   data : e and side_ct
##
##   statistic  : 1.03705
##   num df     : 1
##   denom df   : 572.0646
##   p.value    : 0.3089391
##
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

```r
bf.test(formula = e ~ side_mass, # y~x
        data = bf_data_CEWL, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

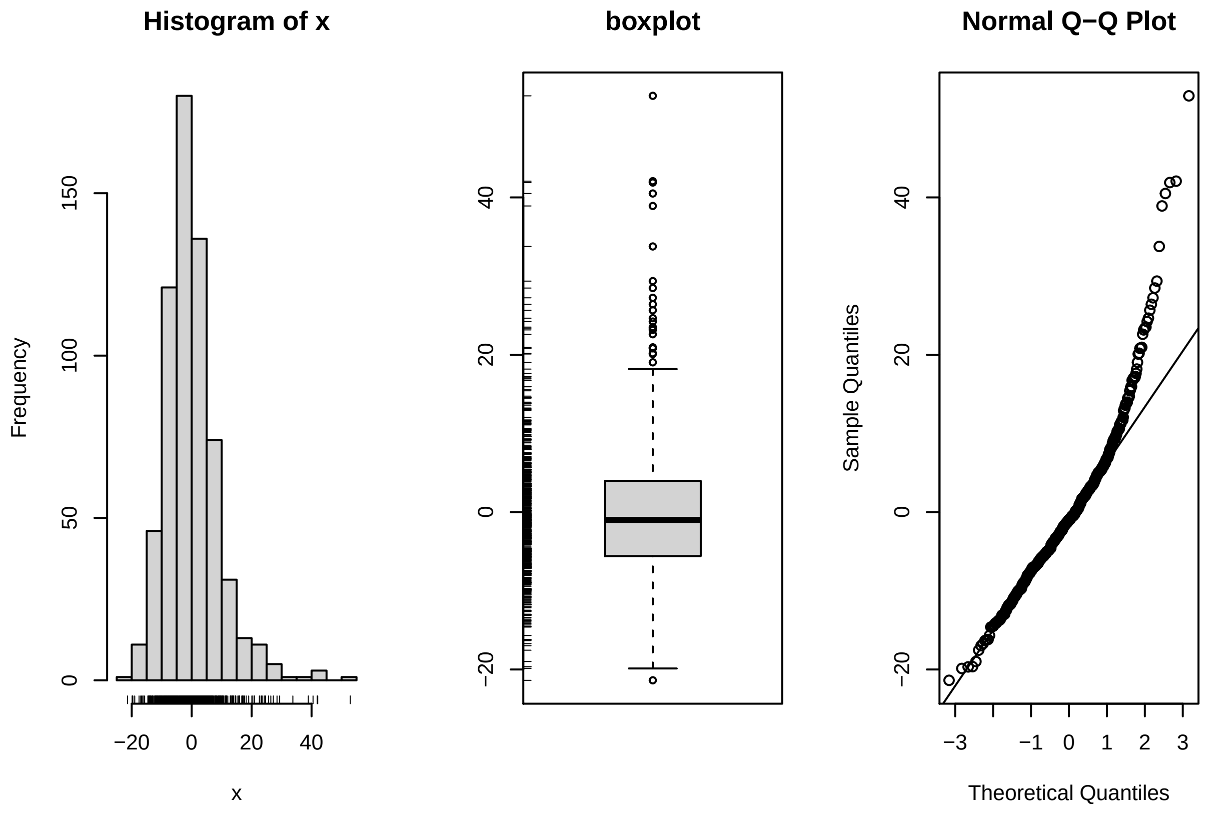
```
##
##   Brown-Forsythe Test (alpha = 0.05)
## -------------------------------------------------------------
##   data : e and side_mass
##
##   statistic  : 0.1660767
##   num df     : 1
##   denom df   : 632.7778
##   p.value    : 0.6837602
##
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

Equal variance is satisfied for all 5 continuous predictor variables.

Now check normality. Is the distribution of residuals **normal**?

use Shapiro-Wilk normality test: H0: data is NOT significantly different from normal distribution HA: data IS significantly different from normal distribution

```r
simple.eda(CEWL_mod_res$e)
```



```r
shapiro.test(CEWL_mod_res$e)
```

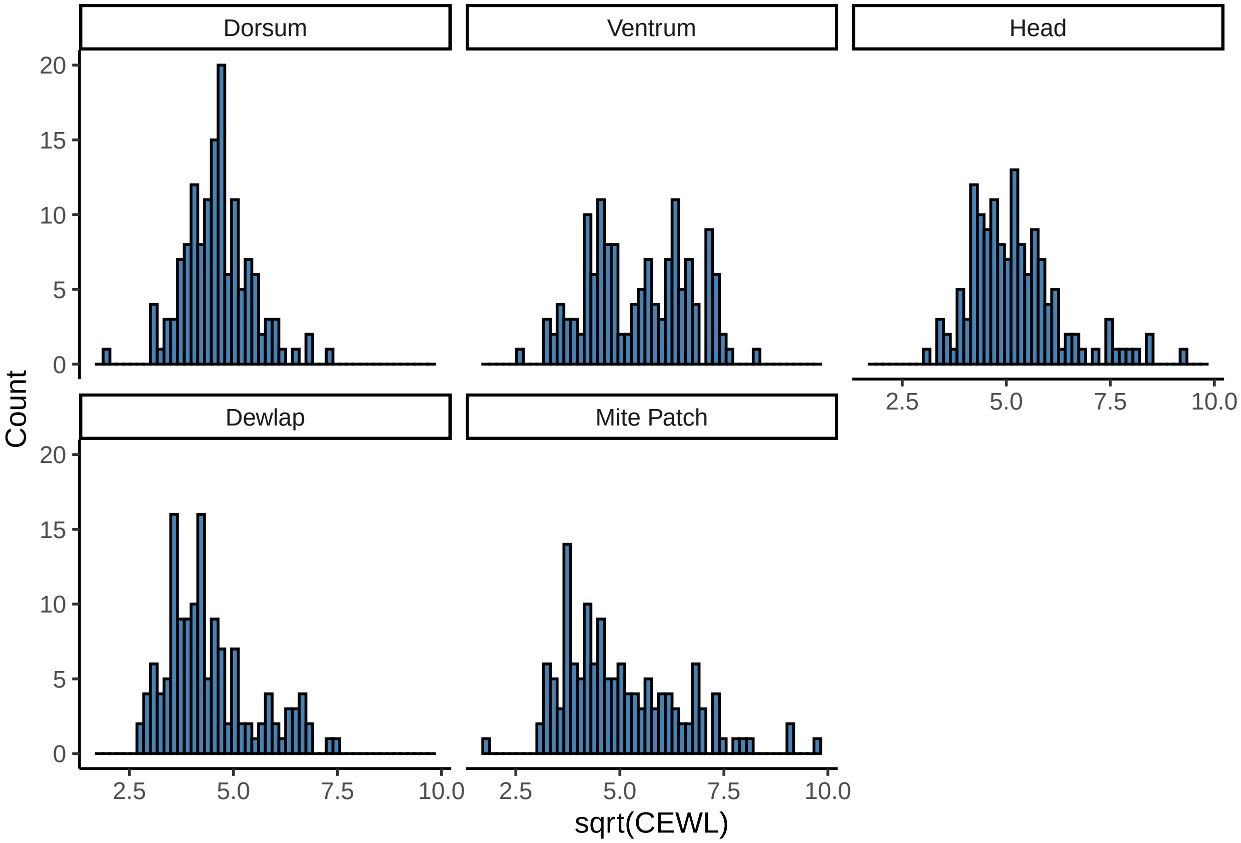
```
##
##  Shapiro-Wilk normality test
##
## data:  CEWL_mod_res$e
## W = 0.92297, p-value < 2.2e-16
```

not normal!

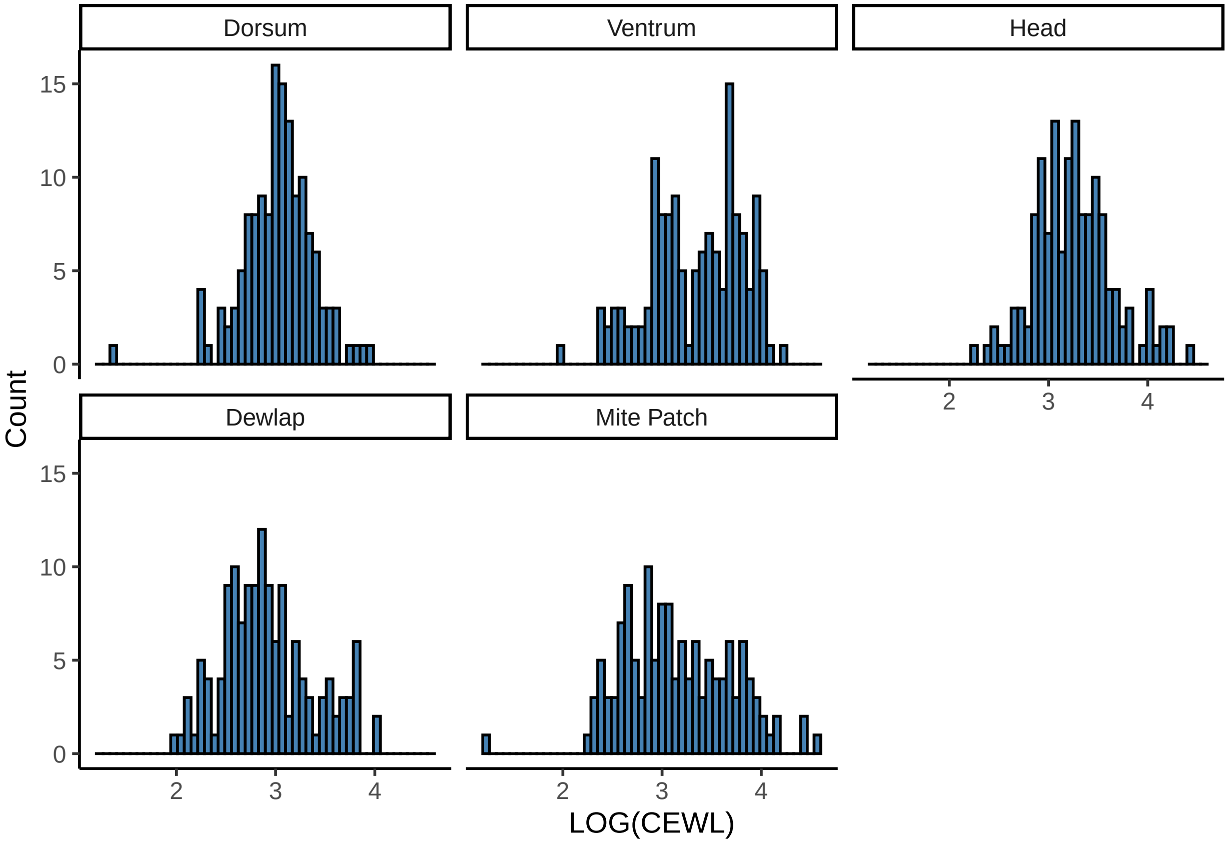
Test Transformations

Can I improve satisfaction of LM assumptions by transforming the dependent variable? In particular, conditions of linearity and normality are not met.

```r
# sqrt(TEWL)
CEWL_data_full %>%
  ggplot(., aes(x = sqrt(TEWL_g_m2h))) +
  geom_histogram(color = "black", fill="steelblue", bins=50) +
  theme_classic() +
  xlab("sqrt(CEWL)") +
  ylab("Count") +
  facet_wrap(~region)
```



```r
# log(TEWL)
CEWL_data_full %>%
  ggplot(., aes(x = log(TEWL_g_m2h))) +
  geom_histogram(color = "black", fill="steelblue", bins=50) +
  theme_classic() +
  xlab("LOG(CEWL)") +
  ylab("Count") +
  facet_wrap(~region)
```



Log transforming seems to be pretty effective across body regions.

Transform & Re-Model

I will log-transform CEWL and see whether it makes the models satisfy LMM assumptions better.

Run CEWL model 4 with log-transformed CEWL:

```
# log-transformed model 3
CEWL_mod3_t <- lme4::lmer(data = CEWL_mod_dat2,
                          log(TEWL_g_m2h) ~
                            region * (mass_g +
                                     temp_C_interpol +
                                     VPD_kPa_int) +
                            cloacal_temp_C +
                            Solar_rad_Wm2_interpol +
                            Wind_mph_interpol +
                            (1|individual_ID))
summary(CEWL_mod3_t)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: log(TEWL_g_m2h) ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
##     cloacal_temp_C + Solar_rad_Wm2_interpol + Wind_mph_interpol +
##     (1 | individual_ID)
##    Data: CEWL_mod_dat2
##
## REML criterion at convergence: 627.9
##
## Scaled residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -4.4274 -0.5429 -0.0339  0.5225  3.8671
##
## Random effects:
##  Groups        Name        Variance Std.Dev.
##  individual_ID (Intercept) 0.04171  0.2042
##  Residual                  0.10871  0.3297
## Number of obs: 635, groups:  individual_ID, 129
##
## Fixed effects:
##                                    Estimate Std. Error t value
## (Intercept)                      -0.7816294  0.8817391  -0.886
## regionVentrum                     0.8513610  0.9014481   0.944
## regionHead                        0.4224072  0.9025140   0.468
## regionDewlap                     -1.7969919  0.9063002  -1.983
## regionMite Patch                  2.5305313  0.9073375   2.789
## mass_g                            0.0210995  0.0135813   1.554
## temp_C_interpol                   0.1359529  0.0606223   2.243
## VPD_kPa_int                      -1.3361800  0.4046246  -3.302
## cloacal_temp_C                    0.0828729  0.0133486   6.208
## Solar_rad_Wm2_interpol            0.0003676  0.0001839   1.999
## Wind_mph_interpol                -0.0375935  0.0545089  -0.690
## regionVentrum:mass_g              0.0248582  0.0162368   1.531
## regionHead:mass_g                -0.0164431  0.0161246  -1.020
## regionDewlap:mass_g               0.0416121  0.0164751   2.526
## regionMite Patch:mass_g           0.0461077  0.0166714   2.766
## regionVentrum:temp_C_interpol    -0.0772165  0.0648277  -1.191
## regionHead:temp_C_interpol       -0.0133516  0.0648374  -0.206
## regionDewlap:temp_C_interpol      0.0571879  0.0649451   0.881
## regionMite Patch:temp_C_interpol -0.2452283  0.0650031  -3.773
## regionVentrum:VPD_kPa_int         0.8599227  0.4510016   1.907
## regionHead:VPD_kPa_int            0.2743674  0.4510004   0.608
## regionDewlap:VPD_kPa_int          0.2352187  0.4521347   0.520
## regionMite Patch:VPD_kPa_int      2.1796293  0.4531358   4.810
```

```
##
## Correlation matrix not shown by default, as p = 23 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

```
# compare
anova(CEWL_mod3_t, CEWL_mod3_adj)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: CEWL_mod_dat2
## Models:
## CEWL_mod3_t: log(TEWL_g_m2h) ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
## CEWL_mod3_t:     cloacal_temp_C + Solar_rad_Wm2_interpol + Wind_mph_interpol +
## CEWL_mod3_t:     (1 | individual_ID)
## CEWL_mod3_adj: TEWL_g_m2h ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
## CEWL_mod3_adj:     cloacal_temp_C + Wind_mph_interpol + Solar_rad_Wm2_interpol +
## CEWL_mod3_adj:     (1 | individual_ID)
##               npar    AIC    BIC   logLik deviance Chisq Df Pr(>Chisq)
## CEWL_mod3_t     25  556.2  667.5  -253.09    506.2
## CEWL_mod3_adj   25 4842.3 4953.6 -2396.15   4792.3     0  0
```

re-run using lmertest for p-values:

```
CEWL_mod_final <- lmerTest::lmer(data = CEWL_mod_dat2,
                          log(TEWL_g_m2h) ~
                            region * (mass_g +
                                     temp_C_interpol +
                                     VPD_kPa_int) +
                            cloacal_temp_C +
                            Solar_rad_Wm2_interpol +
                            Wind_mph_interpol +
                            (1|individual_ID))
summary(CEWL_mod_final)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: log(TEWL_g_m2h) ~ region * (mass_g + temp_C_interpol + VPD_kPa_int) +
##     cloacal_temp_C + Solar_rad_Wm2_interpol + Wind_mph_interpol +
##     (1 | individual_ID)
##    Data: CEWL_mod_dat2
##
## REML criterion at convergence: 627.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.4274 -0.5429 -0.0339  0.5225  3.8671
##
## Random effects:
##  Groups        Name        Variance Std.Dev.
##  individual_ID (Intercept) 0.04171  0.2042
##  Residual                  0.10871  0.3297
## Number of obs: 635, groups:  individual_ID, 129
##
## Fixed effects:
##                                     Estimate Std. Error         df t value
## (Intercept)                       -7.816e-01  8.817e-01  3.171e+02  -0.886
## regionVentrum                      8.514e-01  9.014e-01  4.896e+02   0.944
## regionHead                         4.224e-01  9.025e-01  4.899e+02   0.468
## regionDewlap                      -1.797e+00  9.063e-01  4.910e+02  -1.983
## regionMite Patch                   2.531e+00  9.073e-01  4.899e+02   2.789
## mass_g                             2.110e-02  1.358e-02  4.698e+02   1.554
## temp_C_interpol                    1.360e-01  6.062e-02  3.477e+02   2.243
## VPD_kPa_int                       -1.336e+00  4.046e-01  3.843e+02  -3.302
## cloacal_temp_C                     8.287e-02  1.335e-02  1.206e+02   6.208
## Solar_rad_Wm2_interpol             3.676e-04  1.839e-04  1.200e+02   1.999
## Wind_mph_interpol                 -3.759e-02  5.451e-02  1.198e+02  -0.690
## regionVentrum:mass_g               2.486e-02  1.624e-02  4.940e+02   1.531
## regionHead:mass_g                 -1.644e-02  1.612e-02  4.928e+02  -1.020
## regionDewlap:mass_g                4.161e-02  1.648e-02  4.978e+02   2.526
## regionMite Patch:mass_g            4.611e-02  1.667e-02  4.937e+02   2.766
## regionVentrum:temp_C_interpol     -7.722e-02  6.483e-02  4.893e+02  -1.191
## regionHead:temp_C_interpol        -1.335e-02  6.484e-02  4.893e+02  -0.206
## regionDewlap:temp_C_interpol       5.719e-02  6.495e-02  4.899e+02   0.881
## regionMite Patch:temp_C_interpol -2.452e-01  6.500e-02  4.895e+02  -3.773
## regionVentrum:VPD_kPa_int          8.599e-01  4.510e-01  4.893e+02   1.907
## regionHead:VPD_kPa_int             2.744e-01  4.510e-01  4.893e+02   0.608
## regionDewlap:VPD_kPa_int          2.352e-01  4.521e-01  4.902e+02   0.520
## regionMite Patch:VPD_kPa_int      2.180e+00  4.531e-01  4.897e+02   4.810
##                                   Pr(>|t|)
## (Intercept)                       0.376040
## regionVentrum                     0.345412
## regionHead                        0.639968
## regionDewlap                      0.047949 *
## regionMite Patch                  0.005494 **
## mass_g                            0.120962
## temp_C_interpol                   0.025551 *
## VPD_kPa_int                       0.001049 **
## cloacal_temp_C                    7.85e-09 ***
## Solar_rad_Wm2_interpol            0.047910 *
## Wind_mph_interpol                 0.491731
## regionVentrum:mass_g              0.126415
## regionHead:mass_g                 0.308346
## regionDewlap:mass_g               0.011854 *
## regionMite Patch:mass_g           0.005893 **
## regionVentrum:temp_C_interpol     0.234190
## regionHead:temp_C_interpol        0.836936
## regionDewlap:temp_C_interpol      0.378989
## regionMite Patch:temp_C_interpol 0.000181 ***
## regionVentrum:VPD_kPa_int         0.057145 .
## regionHead:VPD_kPa_int            0.543236
## regionDewlap:VPD_kPa_int          0.603131
## regionMite Patch:VPD_kPa_int      2.01e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Correlation matrix not shown by default, as p = 23 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

Re-Check Assumptions (transformed model 3)

First, get residuals:

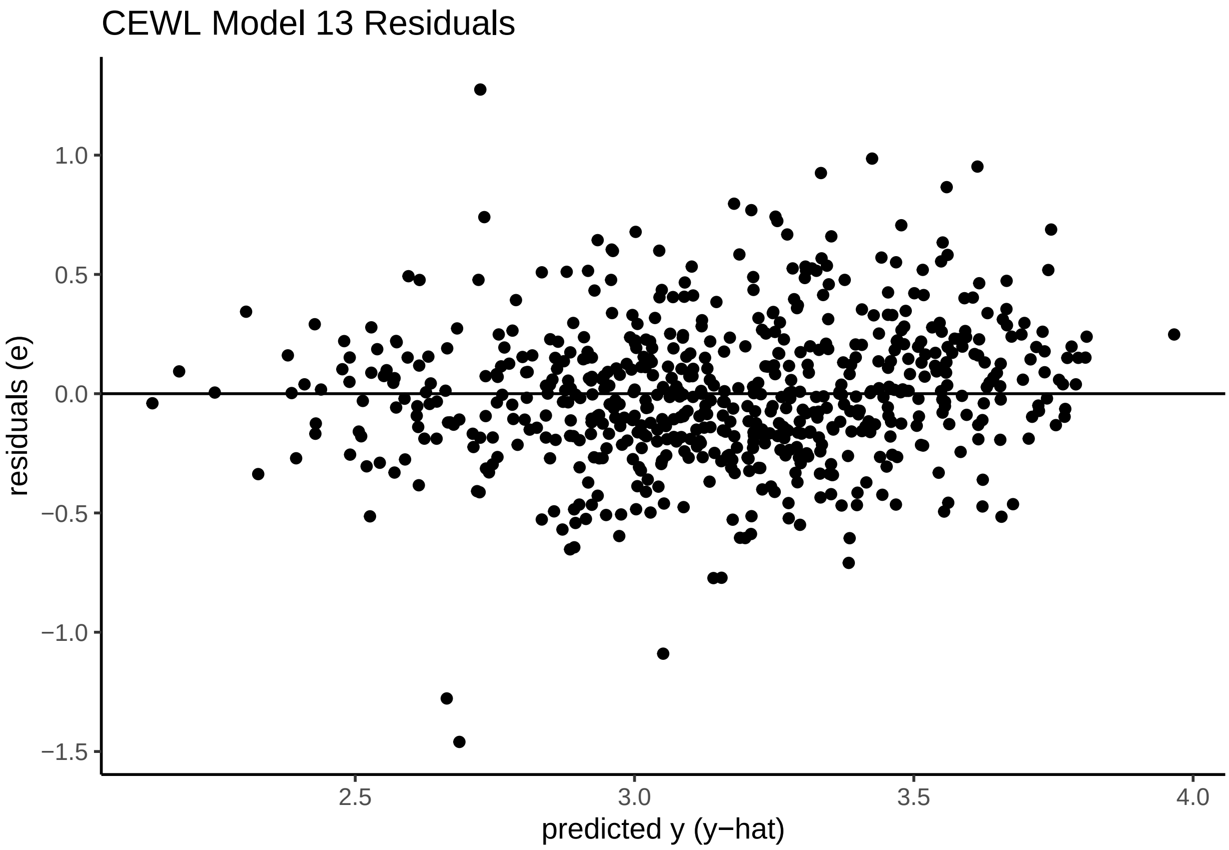
```
CEWL_t_mod_res <- CEWL_mod_dat2 %>%
  mutate(y_hat = predict(CEWL_mod_final),
         e = residuals(CEWL_mod_final))
```

Linearity and Equal Variance

Is the function **linear**? Is there **equal** variance of the residuals? The residuals should be homoskedactic relative to y_hat (or x). We don't care if there is a relationship between the residuals ~ dependent variable (actual y).

Plotting residuals shows us whether the data meets linearity and equal variance assumptions:

```
ggplot(data = CEWL_t_mod_res, aes(x = y_hat, y = e)) +
  geom_point() +
  theme_classic() +
  xlab("predicted y (y-hat)") +
  ylab("residuals (e)") +
  ggtitle("CEWL Model 13 Residuals") +
  geom_hline(yintercept = 0)
```



It looks much much better. :) Equal error variance doesn't look perfect, but there's no more fan shape.

Brown-Forsythe test to statistically check equal variance:

H0: normally distributed (non-sig test is GOOD) HA: NOT normally distributed (reject nul == assumption not satisfied)

```
# need to create the right data & format first
bf_data_CEWL_t <- CEWL_t_mod_res %>%
  dplyr::mutate(middle_mass = median(mass_g), # mass
                side_mass = as.factor(mass_g > middle_mass),
                # solar radiation
                middle_sorad = median(Solar_rad_Wm2_interpol),
                side_sorad = as.factor(Solar_rad_Wm2_interpol > middle_sorad),
                # temperature
                middle_temp = median(temp_C_interpol),
                side_temp = as.factor(temp_C_interpol > middle_temp),
                # VPD
                middle_absh = median(VPD_kPa_int),
                side_absh = as.factor(VPD_kPa_int > middle_absh),
                middle_ct = median(cloacal_temp_C),
                side_ct = as.factor(cloacal_temp_C > middle_ct)
                )


# now run test
bf.test(formula = e ~ side_absh, # y~x
        data = bf_data_CEWL_t, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

```
## 
##   Brown-Forsythe Test (alpha = 0.05) 
## ------------------------------------------------------------- 
##   data : e and side_absh 
## 
##   statistic  : 0.03149034 
##   num df     : 1 
##   denom df   : 614.5099 
##   p.value    : 0.8592094 
## 
##   Result     : Difference is not statistically significant. 
## -------------------------------------------------------------
```

```
bf.test(formula = e ~ side_temp, # y~x
        data = bf_data_CEWL_t, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

```
## 
##   Brown-Forsythe Test (alpha = 0.05) 
## ------------------------------------------------------------- 
##   data : e and side_temp 
## 
##   statistic  : 1.665708e-05 
##   num df     : 1 
##   denom df   : 579.1463 
##   p.value    : 0.996745 
## 
##   Result     : Difference is not statistically significant. 
## -------------------------------------------------------------
```

```
bf.test(formula = e ~ side_sorad, # y~x
        data = bf_data_CEWL_t, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

```
## 
##   Brown-Forsythe Test (alpha = 0.05) 
## ------------------------------------------------------------- 
##   data : e and side_sorad 
## 
##   statistic  : 6.453783e-05 
##   num df     : 1 
##   denom df   : 632.9506 
##   p.value    : 0.9935928 
## 
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

```r
bf.test(formula = e ~ side_ct, # y~x
        data = bf_data_CEWL_t, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

```
##
##   Brown-Forsythe Test (alpha = 0.05)
## -------------------------------------------------------------
##   data : e and side_ct
##
##   statistic  : 0.8637264
##   num df     : 1
##   denom df   : 614.4172
##   p.value    : 0.353063
##
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

```r
bf.test(formula = e ~ side_mass, # y~x
        data = bf_data_CEWL_t, # dataframe
        alpha = 0.05, # default 0.05
        na.rm = TRUE, # remove missing data before running?
        verbose = TRUE # print output to console?
        )
```

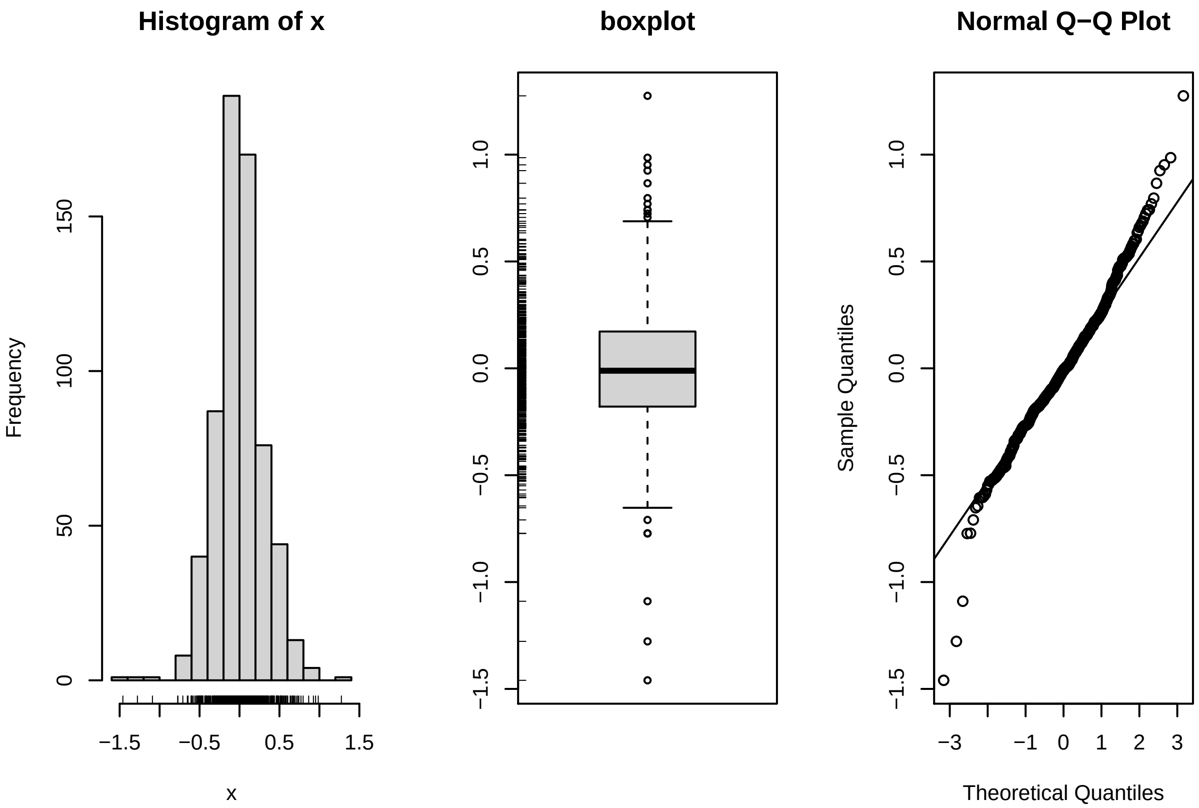
```
##
##   Brown-Forsythe Test (alpha = 0.05)
## -------------------------------------------------------------
##   data : e and side_mass
##
##   statistic  : 0.1937521
##   num df     : 1
##   denom df   : 629.8677
##   p.value    : 0.6599629
##
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

Equal variance is still satisfied for all continuous predictor variables.

Now check normality. Is the distribution of residuals **normal**?

use Shapiro-Wilk normality test: H0: data is NOT significantly different from normal distribution HA: data IS significantly different from normal distribution

```r
simple.eda(CEWL_t_mod_res$e)
```



```r
shapiro.test(CEWL_t_mod_res$e)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  CEWL_t_mod_res$e
## W = 0.98104, p-value = 2.51e-07
```

Still not statistically normal… but the distribution looks a lot better.

Conclusion

The best CEWL model should use log-transformed CEWL because this greatly improves the model based on AIC and it allows the model to satisfy the linearity assumption of LMM.

```r
# save log model 3 summary object
CEWL_best_t_mod <- broom.mixed::tidy(CEWL_mod_final)
# export
write.csv(CEWL_best_t_mod, "./best models/CEWL_best_mod_t_vals.csv")
```

What to Present in the Paper

- figures (exported)
- hct SLR
- best osml mod
- best CEWL mod (CEWL log-transformed)