

Climate Water Loss Experiment - Treatment Hydration Analysis

Savannah Weaver

2021

Contents

Packages	2
Background and Goals	2
Data	3
Load	3
Split	4
Check	4
Experiment Models	6
SMI	6
Building	6
Assumptions	8
Save	10
Hematocrit	11
Building	11
Selection	13
Assumptions	14
Save	16
Osmolality	18
Building	18
Selection	20
Assumptions	21
Save	23
CEWL	25
Building	25
Assumptions	28
Save	30
Group Export	31
Effect Estimates (from coefficients)	32
Rehydration Models	34
SMI	34
Hematocrit	34
Osmolality	35
Group Export	36
Figures	36
Means to Overlay	36
SMI	37

Ind + Means	37
Means Only	38
LM + SE	40
Ending Values Boxplot	41
Hct	43
Ind + Means	43
Means Only	45
Ending Values Boxplot	47
Osm1	49
Ind + Means	49
Means Only	51
Stats! Check Pairwise Diffs ~ Time	52
LM + SE	53
Ending Values Boxplot	54
CEWL	56
Ind + Means	56
Means Only	58
LM + SE	59
Ending Values Boxplot	61
CEWL ~ Osm1	63
Multi-Figures	65

Packages

```

`%nin%` = Negate(`%in%`)
if (!require("tidyverse")) install.packages("tidyverse")
library("tidyverse") # workflow and plots
if (!require("lme4")) install.packages("lme4")
library("lme4") # for LMMs
if (!require("lmerTest")) install.packages("lmerTest")
library("lmerTest") # for p-values
if (!require("ggpubr")) install.packages("ggpubr")
library("ggpubr") # for multi-ggplot figs
if (!require("UsingR")) install.packages("UsingR")
library("UsingR") # simple.eda model assumption checker
if (!require("broom.mixed")) install.packages("broom.mixed")
library("broom.mixed") # lmer model export
if (!require("car")) install.packages("car")
library("car") # VIFs
if (!require("AICcmodavg")) install.packages("AICcmodavg")
library("AICcmodavg") # model selection
if (!require("RColorBrewer")) install.packages("RColorBrewer")
library("RColorBrewer") # color

```

Background and Goals

This data was collected June - August by Master's student Savannah Weaver, advisor Dr. Emily Taylor, and research assistants Tess McIntyre and Taylor Van Rossum. Adult male *Sceloporus occidentalis* were caught across the Cal Poly campus then acclimated to 4 different climate treatments. **This R file analyzes the effect of experimental climate treatments on lizard body condition, osmotic balance, and osmoregulation.** Please refer to the published scientific journal article for full details.

Data

Load

Read-in data that was compiled, formatted, and checked for completeness in ‘wrangling_general’. See that file for information related to the variables.

```
dat <- read_rds("./data/analysis_data_experiment.RDS")
summary(dat)
```

```
## measurement_date      type      individual_ID      mass_g
## Min.   :2021-06-16    exp :804    201      : 7    Min.   : 7.00
## 1st Qu.:2021-07-01    rehab:132  202      : 7    1st Qu.: 9.50
## Median :2021-07-25                                203      : 7    Median :10.60
## Mean   :2021-07-22                                204      : 7    Mean   :10.64
## 3rd Qu.:2021-08-14                                205      : 7    3rd Qu.:11.70
## Max.   :2021-09-01                                206      : 7    Max.   :17.40
##                                     (Other):894
## hematocrit_percent trial_number temp_tmt  humidity_tmt  SVL_mm
## Min.   :13.00      1:175      Hot :467    Humid:468    Min.   :60.00
## 1st Qu.:26.00      2:203      Cool:469   Dry  :468    1st Qu.:66.00
## Median :32.00      3:231                                Median :67.00
## Mean   :31.99      4:189                                Mean   :67.74
## 3rd Qu.:37.00      5:138                                3rd Qu.:70.00
## Max.   :52.00                                Max.   :77.00
## NA's    :408
##          tmt          day_n      day_factor osmolality_mmol_kg_mean
## Hot Dry   :237    Min.   : 0.000    0 :134    Min.   :295.3
## Hot Humid :230    1st Qu.: 4.000    4 :134    1st Qu.:336.1
## Cool Dry  :231    Median : 6.000    5 :134    Median :351.3
## Cool Humid:238    Mean   : 5.705    6 :134    Mean   :354.3
##                                     3rd Qu.: 8.000    7 :134    3rd Qu.:370.0
##                                     Max.   :10.000   8 :134    Max.   :471.5
##                                     10:132   NA's    :414
## CEWL_g_m2h_mean  msmt_temp_C  msmt_RH_percent  cloacal_temp_C
## Min.   : 7.152    Min.   :24.80    Min.   :25.52    Min.   :23.00
## 1st Qu.:19.755    1st Qu.:26.20    1st Qu.:46.11    1st Qu.:25.00
## Median :24.152    Median :26.74    Median :47.88    Median :26.00
## Mean   :24.767    Mean   :26.72    Mean   :46.74    Mean   :25.92
## 3rd Qu.:28.505    3rd Qu.:27.11    3rd Qu.:50.50    3rd Qu.:27.00
## Max.   :56.066    Max.   :29.20    Max.   :56.16    Max.   :30.00
## NA's    :669      NA's    :668      NA's    :668      NA's    :668
## msmt_temp_K      e_s_kPa_m      e_a_kPa_m      msmt_VPD_kPa
## Min.   :297.9    Min.   :3.219    Min.   :0.9894    Min.   :1.486
## 1st Qu.:299.4    1st Qu.:3.504    1st Qu.:1.6464    1st Qu.:1.767
## Median :299.9    Median :3.620    Median :1.7411    Median :1.853
## Mean   :299.9    Mean   :3.620    Mean   :1.6833    Mean   :1.937
## 3rd Qu.:300.3    3rd Qu.:3.701    3rd Qu.:1.7992    3rd Qu.:2.012
## Max.   :302.4    Max.   :4.194    Max.   :1.9326    Max.   :3.021
## NA's    :668      NA's    :668      NA's    :668      NA's    :668
## SMI
## Min.   : 6.747
## 1st Qu.: 9.714
## Median :10.594
## Mean   :10.599
```

```
## 3rd Qu.:11.390
## Max.    :15.063
##
```

Split

Make sub-dataframes without rehab data / with only rehab-related data:

```
dat_no_rehab <- dat %>%
  dplyr::filter(day_n %in% c(seq(0,8)))
dat_for_rehab <- dat %>%
  dplyr::filter(day_n %in% c(8, 10))
```

Check

Dates:

```
unique(dat$measurement_date)
```

```
## [1] "2021-06-16" "2021-06-20" "2021-06-21" "2021-06-22" "2021-06-23"
## [6] "2021-06-24" "2021-06-26" "2021-06-30" "2021-07-01" "2021-07-02"
## [11] "2021-07-03" "2021-07-04" "2021-07-06" "2021-07-20" "2021-07-24"
## [16] "2021-07-25" "2021-07-26" "2021-07-27" "2021-07-28" "2021-07-30"
## [21] "2021-08-08" "2021-08-12" "2021-08-13" "2021-08-14" "2021-08-15"
## [26] "2021-08-16" "2021-08-18" "2021-08-22" "2021-08-26" "2021-08-27"
## [31] "2021-08-28" "2021-08-29" "2021-08-30" "2021-09-01"
```

Number of measurements for each lizard:

```
dat_no_rehab %>%
  group_by(individual_ID) %>%
  summarise(n = n()) %>%
  arrange(n)
```

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

```
## # A tibble: 134 x 2
##   individual_ID     n
##   <fct>         <int>
## 1 201             6
## 2 202             6
## 3 203             6
## 4 204             6
## 5 205             6
## 6 206             6
## 7 207             6
## 8 208             6
## 9 209             6
## 10 210            6
## # ... with 124 more rows
```

Every lizard has 6 experimental measurements: pre-tmt, mid-tmt, post-tmt, and mass checks on each of the 3 days between mid and post-tmt.

Did any of the treatment groups inherently start out with large differences in response variables?

```
dat %>%
  dplyr::filter(day_n == 0) %>%
```

```
group_by(tmt) %>%
  summarise(mean(mass_g),
            sd(mass_g),
            mean(SMI),
            mean(hematocrit_percent),
            mean(osmolality_mmol_kg_mean),
            mean(CEWL_g_m2h_mean))
```

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

## # A tibble: 4 x 7
##   tmt           `mean(mass_g)` `sd(mass_g)` `mean(SMI)` `mean(hematocrit_percent)`
##   <fct>           <dbl>         <dbl>         <dbl>         <dbl>
## 1 Hot Dry           12.0           1.68           11.8           38.9
## 2 Hot Humid          11.6           1.75           11.5           37.9
## 3 Cool Dry           11.8           1.61           11.8           39.3
## 4 Cool Humid          11.6           1.35           11.7           39.6
## # ... with 2 more variables: mean(osmolality_mmol_kg_mean) <dbl>,
## #   mean(CEWL_g_m2h_mean) <dbl>
```

There are slight differences, but overall the starting values across groups are more or less the same.

Temp & RH during CEWL measurements:

```
summary(dat_no_rehab)
```

measurement_date	type	individual_ID	mass_g
Min. :2021-06-16	exp :804	201 : 6	Min. : 7.00
1st Qu.:2021-06-30	rehab: 0	202 : 6	1st Qu.: 9.50
Median :2021-07-25		203 : 6	Median :10.60
Mean :2021-07-22		204 : 6	Mean :10.65
3rd Qu.:2021-08-13		205 : 6	3rd Qu.:11.60
Max. :2021-08-30		206 : 6	Max. :17.40
		(Other):768	

hematocrit_percent	trial_number	temp_tmt	humidity_tmt	SVL_mm
Min. :13.00	1:150	Hot :402	Humid:402	Min. :60.00
1st Qu.:28.25	2:174	Cool:402	Dry :402	1st Qu.:66.00
Median :33.00	3:198			Median :67.00
Mean :33.75	4:162			Mean :67.73
3rd Qu.:39.00	5:120			3rd Qu.:70.00
Max. :52.00				Max. :77.00
NA's :406				

tmt	day_n	day_factor	osmolality_mmol_kg_mean
Hot Dry :204	Min. :0.0	0 :134	Min. :295.3
Hot Humid :198	1st Qu.:4.0	4 :134	1st Qu.:334.7
Cool Dry :198	Median :5.5	5 :134	Median :348.3
Cool Humid:204	Mean :5.0	6 :134	Mean :352.3
	3rd Qu.:7.0	7 :134	3rd Qu.:367.4
	Max. :8.0	8 :134	Max. :445.5
		10: 0	NA's :413

CEWL_g_m2h_mean	msmt_temp_C	msmt_RH_percent	cloacal_temp_C
Min. : 7.152	Min. :24.80	Min. :25.52	Min. :23.00
1st Qu.:19.755	1st Qu.:26.20	1st Qu.:46.11	1st Qu.:25.00
Median :24.152	Median :26.74	Median :47.88	Median :26.00
Mean :24.767	Mean :26.72	Mean :46.74	Mean :25.92
3rd Qu.:28.505	3rd Qu.:27.11	3rd Qu.:50.50	3rd Qu.:27.00

```
## Max. :56.066 Max. :29.20 Max. :56.16 Max. :30.00
## NA's :537 NA's :536 NA's :536 NA's :536
## msmt_temp_K e_s_kPa_m e_a_kPa_m msmt_VPD_kPa
## Min. :297.9 Min. :3.219 Min. :0.9894 Min. :1.486
## 1st Qu.:299.4 1st Qu.:3.504 1st Qu.:1.6464 1st Qu.:1.767
## Median :299.9 Median :3.620 Median :1.7411 Median :1.853
## Mean :299.9 Mean :3.620 Mean :1.6833 Mean :1.937
## 3rd Qu.:300.3 3rd Qu.:3.701 3rd Qu.:1.7992 3rd Qu.:2.012
## Max. :302.4 Max. :4.194 Max. :1.9326 Max. :3.021
## NA's :536 NA's :536 NA's :536 NA's :536
## SMI
## Min. : 7.317
## 1st Qu.: 9.748
## Median :10.624
## Mean :10.607
## 3rd Qu.:11.348
## Max. :14.263
##
```

```
dat_no_rehab %>%
  group_by(type) %>%
  summarise(mean(msmt_temp_C, na.rm = T),
            sd(msmt_temp_C, na.rm = T),
            mean(msmt_RH_percent, na.rm = T),
            sd(msmt_RH_percent, na.rm = T),
            mean(msmt_VPD_kPa, na.rm = T),
            mean(msmt_VPD_kPa, na.rm = T))
```

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

```
## # A tibble: 1 x 6
##   type `mean(msmt_temp_C~ `sd(msmt_temp_C,~ `mean(msmt_RH_pe~ `sd(msmt_RH_perc~
##   <fct>          <dbl>          <dbl>          <dbl>          <dbl>
## 1 exp              26.7              0.799              46.7              6.76
## # ... with 1 more variable: mean(msmt_VPD_kPa, na.rm = T) <dbl>
```

Experiment Models

We predicted that there would be effects of day, humidity, and temperature, but we only expect those **interactions with day** to have an effect because we're observing changes over time.

So, each full model will include the following: `day_n`, `day_n:humidity_tmt`, `day_n:temp_tmt`, and `day_n:humidity_tmt:temp_tmt`. Then I'll test whether dropping any variables will improve the model.

SMI

Building

Models are named based on the number of predictors.

```
SMI_mod4 <- lme4::lmer(data = dat_no_rehab,
                      SMI ~ day_n +
                        day_n:humidity_tmt + day_n:temp_tmt +
                        day_n:humidity_tmt:temp_tmt +
                        (1|trial_number/individual_ID))
car::vif(SMI_mod4) # check collinearity
```

```
##               day_n               day_n:humidity_tmt
##           3.963578               3.937194
##       day_n:temp_tmt day_n:humidity_tmt:temp_tmt
##           3.937208               3.909861

summary(SMI_mod4)

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## SMI ~ day_n + day_n:humidity_tmt + day_n:temp_tmt + day_n:humidity_tmt:temp_tmt +
##       (1 | trial_number/individual_ID)
## Data: dat_no_rehab
##
## REML criterion at convergence: 535.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -5.6017 -0.4555 -0.0526  0.3975  4.2640
##
## Random effects:
##   Groups                Name         Variance Std.Dev.
## individual_ID:trial_number (Intercept) 0.92846  0.9636
## trial_number              (Intercept) 0.21842  0.4673
## Residual                  0.04928  0.2220
## Number of obs: 804, groups: individual_ID:trial_number, 134; trial_number, 5
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)    11.647081   0.225987  51.539
## day_n          -0.238634   0.006036 -39.532
## day_n:humidity_tmtDry -0.065043   0.008440  -7.707
## day_n:temp_tmtCool    0.115552   0.008439  13.692
## day_n:humidity_tmtDry:temp_tmtCool 0.031887   0.011935   2.672
##
## Correlation of Fixed Effects:
##           (Intr) day_n  dy_:_D dy_:_C
## day_n      -0.034
## dy_n:hmdt_D 0.000 -0.710
## dy_n:tmp_tC 0.000 -0.709  0.507
## dy_n:h_D:_C 0.000  0.502 -0.707 -0.707
```

```
drop1(SMI_mod4)
```

```
## Single term deletions
##
## Model:
## SMI ~ day_n + day_n:humidity_tmt + day_n:temp_tmt + day_n:humidity_tmt:temp_tmt +
##       (1 | trial_number/individual_ID)
##               npar      AIC
## <none>                516.79
## day_n:humidity_tmt:temp_tmt    1 521.93
```

This is the best model based on AIC, but we will check dropping the 3-way interaction.

```
SMI_mod3a <- lme4::lmer(data = dat_no_rehab,
  SMI ~
```

```

day_n +
day_n:humidity_tmt + day_n:temp_tmt +
(1|trial_number/individual_ID))
car::vif(SMI_mod3a) # check collinearity

```

```

##          day_n day_n:humidity_tmt    day_n:temp_tmt
##          2.965381      1.968522      1.968280

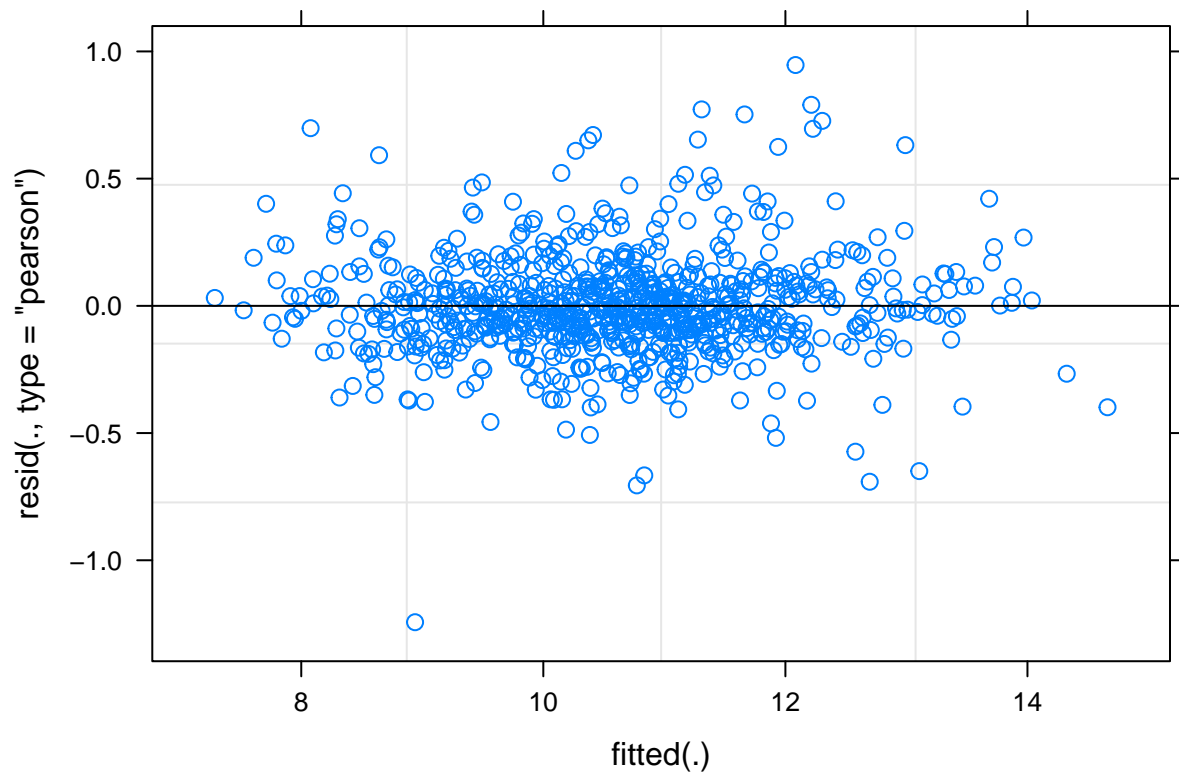
```

All the variables in the full models had huge t-values and we lose significant predictive power without the three-way interaction. So, the full model is the best model to predict SMI/ Body Condition.

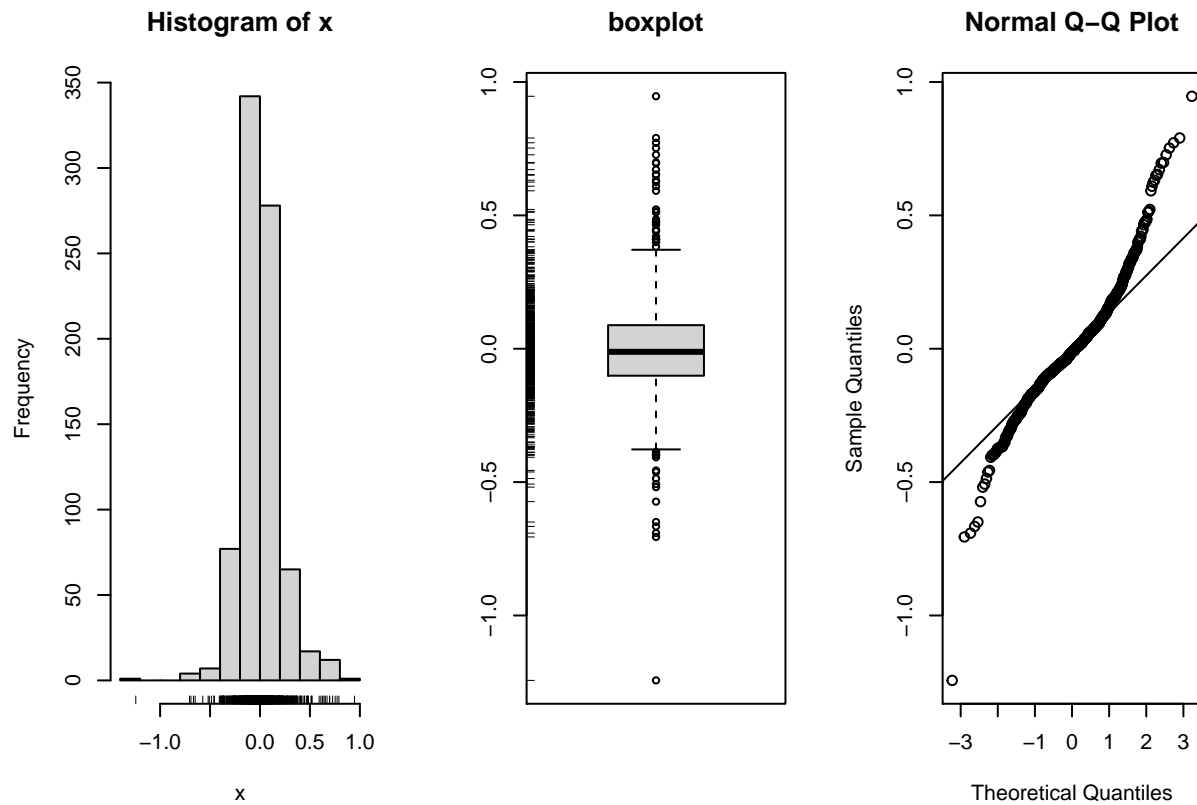
Assumptions

Check linear regression assumptions/conditions:

```
plot(SMI_mod4)
```



```
simple.eda(residuals(SMI_mod4))
```

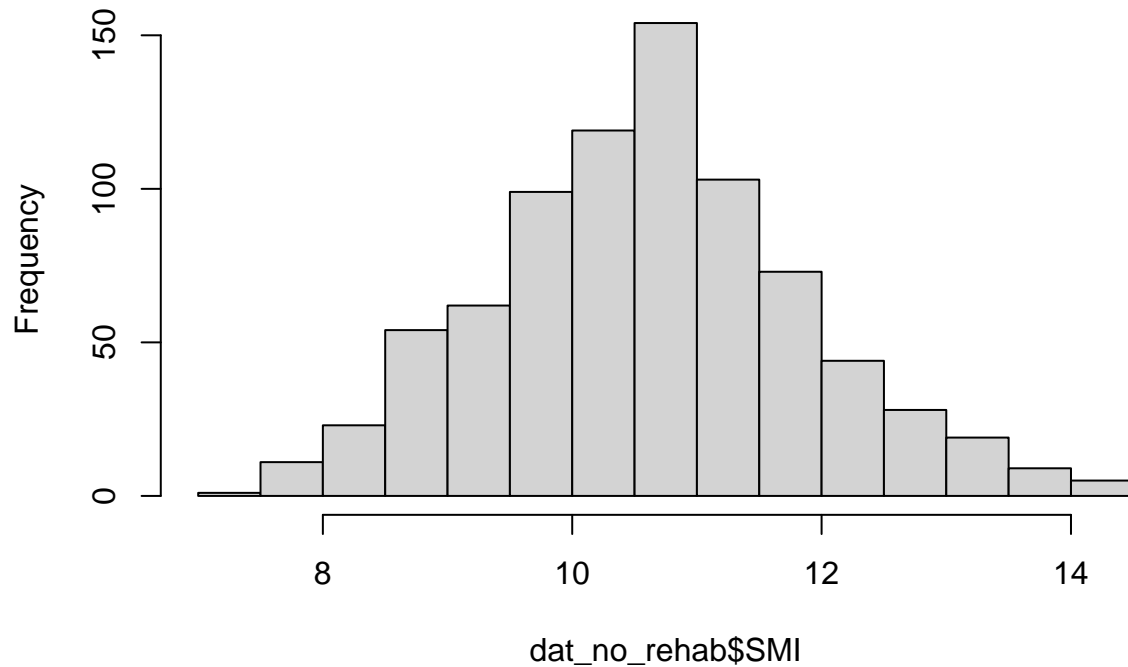



```
shapiro.test(residuals(SMI_mod4))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(SMI_mod4)
## W = 0.94323, p-value < 2.2e-16
```

```
hist(dat_no_rehab$SMI)
```

Histogram of dat_no_rehab\$SMI



Normality of the residuals is very violated, but with our large sample size that should not effect results, and the SMI variable is perfectly evenly distributed, so it should be fine. Equal error variance and linearity look fantastic.

Save

Save best model stats with p-values:

```
SMI_mod_best1 <- lmerTest::lmer(data = dat_no_rehab,
                                SMI ~ day_n +
                                day_n:humidity_tmt + day_n:temp_tmt +
                                day_n:humidity_tmt:temp_tmt +
                                (1|trial_number/individual_ID))
summary(SMI_mod_best1)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## SMI ~ day_n + day_n:humidity_tmt + day_n:temp_tmt + day_n:humidity_tmt:temp_tmt +
## (1 | trial_number/individual_ID)
## Data: dat_no_rehab
##
## REML criterion at convergence: 535.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -5.6017 -0.4555 -0.0526  0.3975  4.2640
##
## Random effects:
## Groups              Name                Variance Std.Dev.
```

```
## individual_ID:trial_number (Intercept) 0.92846 0.9636
## trial_number (Intercept) 0.21842 0.4673
## Residual 0.04928 0.2220
## Number of obs: 804, groups: individual_ID:trial_number, 134; trial_number, 5
##
## Fixed effects:
##
## Estimate Std. Error df t value
## (Intercept) 11.647081 0.225987 3.939851 51.539
## day_n -0.238634 0.006036 696.619034 -39.532
## day_n:humidity_tmtDry -0.065043 0.008440 706.429019 -7.707
## day_n:temp_tmtCool 0.115552 0.008439 706.566812 13.692
## day_n:humidity_tmtDry:temp_tmtCool 0.031887 0.011935 706.556523 2.672
##
## Pr(>|t|)
## (Intercept) 1.01e-06 ***
## day_n < 2e-16 ***
## day_n:humidity_tmtDry 4.38e-14 ***
## day_n:temp_tmtCool < 2e-16 ***
## day_n:humidity_tmtDry:temp_tmtCool 0.00772 **
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
## (Intr) day_n dy_:_D dy_:_C
## day_n -0.034
## dy_n:hmdt_D 0.000 -0.710
## dy_n:tmp_tC 0.000 -0.709 0.507
## dy_n:h_D:_C 0.000 0.502 -0.707 -0.707

SMI_stats <- broom.mixed::tidy(SMI_mod_best1) %>%
  mutate(response = "Body Condition (g)")
```

Hematocrit

Building

```
hct_mod4 <- lme4::lmer(data = dat_no_rehab,
  hematocrit_percent ~ day_n +
    day_n:humidity_tmt + day_n:temp_tmt +
    day_n:humidity_tmt:temp_tmt +
    (1|trial_number/individual_ID))
car::vif(hct_mod4) # check collinearity
```

```
## day_n day_n:humidity_tmt
## 3.254014 3.491829
## day_n:temp_tmt day_n:humidity_tmt:temp_tmt
## 3.491128 3.713867
```

```
summary(hct_mod4)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: hematocrit_percent ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
## day_n:humidity_tmt:temp_tmt + (1 | trial_number/individual_ID)
## Data: dat_no_rehab
##
## REML criterion at convergence: 2386.3
```

```
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.3033 -0.5422 -0.0534  0.5507  2.7645
##
## Random effects:
##      Groups                Name      Variance Std.Dev.
## individual_ID:trial_number (Intercept) 12.258   3.501
## trial_number                (Intercept)  7.004   2.647
## Residual                    14.923   3.863
## Number of obs: 398, groups: individual_ID:trial_number, 134; trial_number, 5
##
## Fixed effects:
##                                Estimate Std. Error t value
## (Intercept)                   38.8414     1.2604  30.817
## day_n                         -1.6014     0.1072 -14.932
## day_n:humidity_tmtDry          0.2226     0.1423   1.565
## day_n:temp_tmtCool             0.4835     0.1414   3.419
## day_n:humidity_tmtDry:temp_tmtCool -0.2278     0.1991  -1.144
##
## Correlation of Fixed Effects:
##              (Intr) day_n  dy_:_D dy_:_C
## day_n        -0.104
## dy_n:hmdt_D   0.000 -0.684
## dy_n:tmp_tC   0.001 -0.688  0.518
## dy_n:h_D:_C   0.000  0.488 -0.713 -0.710
```

```
drop1(hct_mod4)
```

```
## Single term deletions
##
## Model:
## hematocrit_percent ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
##   day_n:humidity_tmt:temp_tmt + (1 | trial_number/individual_ID)
##                                npar      AIC
## <none>                        2393.7
## day_n:humidity_tmt:temp_tmt    1 2393.0
```

It seems like it could be beneficial to drop variables, so test each 1-variable drop:

```
hct_mod3a <- lme4::lmer(data = dat_no_rehab,
                      hematocrit_percent ~
                        day_n +
                        day_n:humidity_tmt + day_n:temp_tmt +
                        (1|trial_number/individual_ID))
car::vif(hct_mod3a) # check collinearity
```

```
##              day_n day_n:humidity_tmt    day_n:temp_tmt
##              2.481855          1.715529          1.732715
```

```
anova(hct_mod4, hct_mod3a)
```

```
## refitting model(s) with ML (instead of REML)
## Data: dat_no_rehab
## Models:
## hct_mod3a: hematocrit_percent ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
```

```
## hct_mod3a:      (1 | trial_number/individual_ID)
## hct_mod4: hematocrit_percent ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
## hct_mod4:      day_n:humidity_tmt:temp_tmt + (1 | trial_number/individual_ID)
##              npar    AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## hct_mod3a      7 2393.0 2420.9 -1189.5   2379.0
## hct_mod4      8 2393.7 2425.6 -1188.9   2377.7 1.3191  1      0.2508
```

We do NOT lose predictive power by dropping the three-way interaction from the hematocrit model, so we should definitely drop it.

Test whether we should drop any of the remaining variables:

Not necessarily different or better.

```
hct_mod2b <- lme4::lmer(data = dat_no_rehab,
                        hematocrit_percent ~
                          day_n + day_n:temp_tmt +
                          (1|trial_number/individual_ID))
anova(hct_mod3a, hct_mod2b)

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

## Data: dat_no_rehab
## Models:
## hct_mod2b: hematocrit_percent ~ day_n + day_n:temp_tmt + (1 | trial_number/individual_ID)
## hct_mod3a: hematocrit_percent ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
## hct_mod3a:      (1 | trial_number/individual_ID)
##              npar    AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## hct_mod2b      6 2392.2 2416.1 -1190.1   2380.2
## hct_mod3a      7 2393.0 2420.9 -1189.5   2379.0 1.1628  1      0.2809
```

We do NOT lose predictive power if we drop humidity.

```
hct_mod2c <- lme4::lmer(data = dat_no_rehab,
                        hematocrit_percent ~
                          day_n +
                          day_n:humidity_tmt +
                          (1|trial_number/individual_ID))
anova(hct_mod3a, hct_mod2c)

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

## Data: dat_no_rehab
## Models:
## hct_mod2c: hematocrit_percent ~ day_n + day_n:humidity_tmt + (1 | trial_number/individual_ID)
## hct_mod3a: hematocrit_percent ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
## hct_mod3a:      (1 | trial_number/individual_ID)
##              npar    AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## hct_mod2c      6 2404.5 2428.5 -1196.3   2392.5
## hct_mod3a      7 2393.0 2420.9 -1189.5   2379.0 13.511  1 0.0002372 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We lose a very significant amount of predictive power if we drop temperature.

Selection

Compare models that did not lose predictive power when three-way interaction and humidity were dropped.

```

hct_models <- list(hct_mod4, hct_mod3a, hct_mod2b, hct_mod2c)

EXP_mod_names1 <- data.frame(predictors =
  c('Day, Day:Humd, Day:Temp, Day:Humd:Temp', #4
    'Day, Day:Humd, Day:Temp', #3a
    'Day, Day:Temp', #2b
    'Day, Day:Humd' #2c
  ))
rownames(EXP_mod_names1) <- c('4', '3a', '2b', '2c')
#calculate AIC of each model
hct_AICc <- data.frame(aictab(cand.set = hct_models,
                             modnames = EXP_mod_names1)) %>%
  mutate(response = "Hematocrit (%)")

```

```

## Warning in aictab.AIClmerMod(cand.set = hct_models, modnames = EXP_mod_names1):
## Model selection for fixed effects is only appropriate with ML estimation:
## REML (default) should only be used to select random effects for a constant set of fixed effects

```

```
hct_AICc
```

```

##               predictors K      AICc Delta_AICc   ModelLik
## 1                Day, Day:Temp 6 2396.833   0.000000 1.000000000
## 2           Day, Day:Humd, Day:Temp 7 2400.538   3.705308 0.156820426
## 3 Day, Day:Humd, Day:Temp, Day:Humd:Temp 8 2402.702   5.869851 0.053134675
## 4                Day, Day:Humd 6 2409.123  12.290321 0.002143831
##      AICcWt   Res.LL   Cum.Wt   response
## 1 0.825015164 -1192.309 0.8250152 Hematocrit (%)
## 2 0.129379230 -1193.125 0.9543944 Hematocrit (%)
## 3 0.043836912 -1193.166 0.9982313 Hematocrit (%)
## 4 0.001768693 -1198.454 1.0000000 Hematocrit (%)

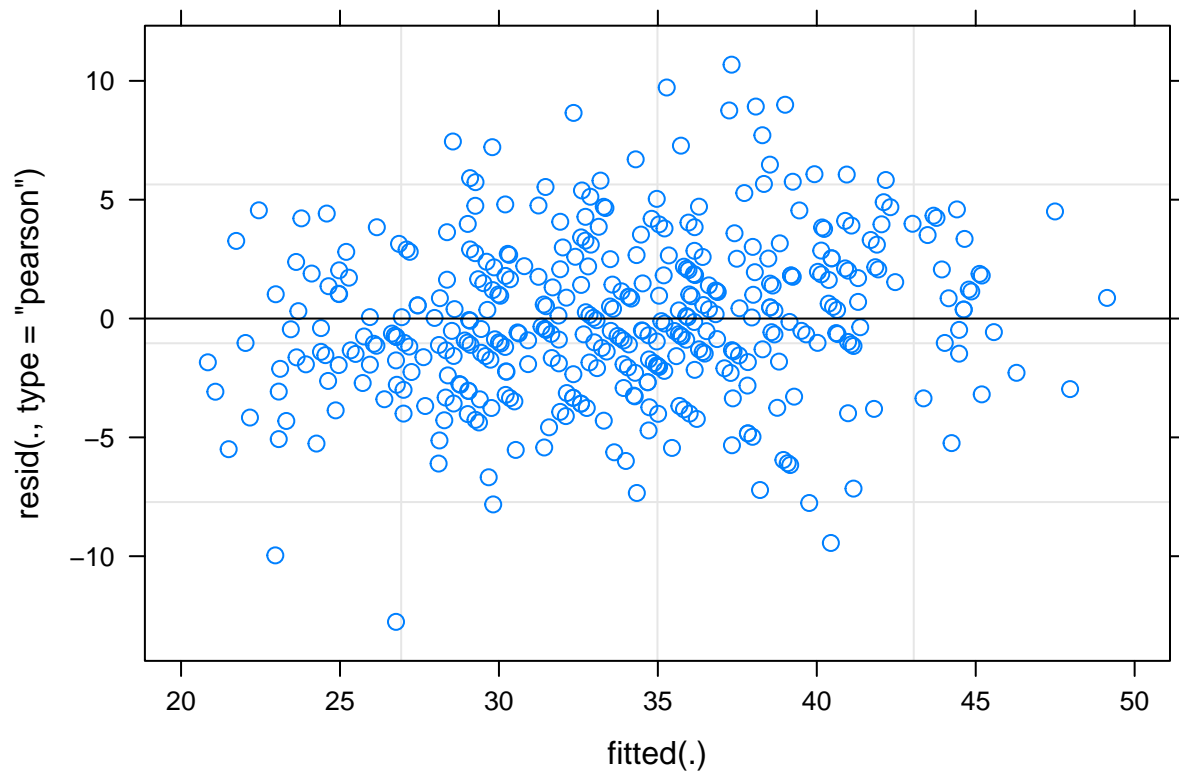
```

Model 2b with only predictors Day and Day*Temp is the best by a good margin.

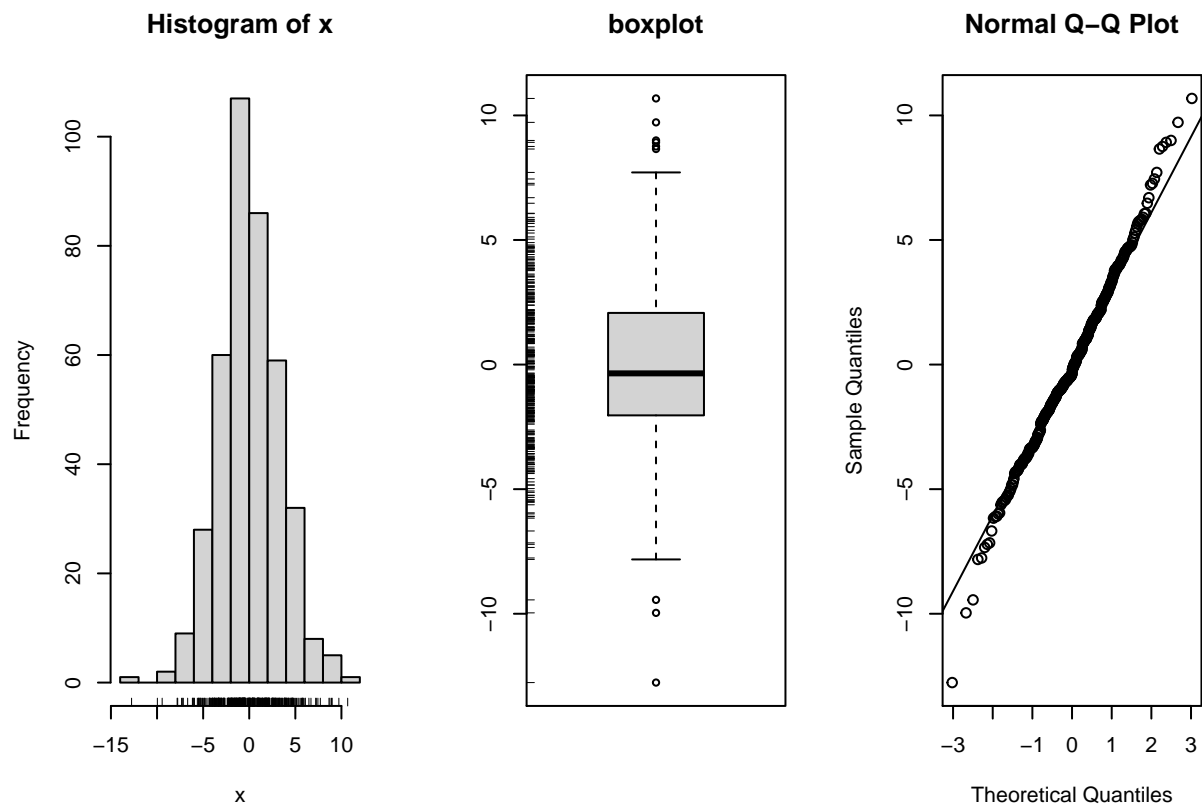
Assumptions

Check linear regression assumptions/conditions:

```
plot(hct_mod2b)
```



```
simple.eda(residuals(hct_mod2b))
```



```
shapiro.test(residuals(hct_mod2b))
```

```
##
## Shapiro-Wilk normality test
##
## data: residuals(hct_mod2b)
## W = 0.9946, p-value = 0.1756
```

All assumptions are satisfied (and beautiful).

Save

Save full model stats with p-values:

```
hct_mod_full <- lmerTest::lmer(data = dat_no_rehab,
                             hematocrit_percent ~
                               day_n +
                               day_n:humidity_tmt + day_n:temp_tmt +
                               day_n:humidity_tmt:temp_tmt +
                               (1|trial_number/individual_ID))
summary(hct_mod_full)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: hematocrit_percent ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
##       day_n:humidity_tmt:temp_tmt + (1 | trial_number/individual_ID)
## Data: dat_no_rehab
##
## REML criterion at convergence: 2386.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.3033 -0.5422 -0.0534  0.5507  2.7645
##
## Random effects:
## Groups              Name                Variance Std.Dev.
## individual_ID:trial_number (Intercept) 12.258   3.501
## trial_number           (Intercept)    7.004   2.647
## Residual                          14.923   3.863
## Number of obs: 398, groups: individual_ID:trial_number, 134; trial_number, 5
##
## Fixed effects:
##              Estimate Std. Error      df t value
## (Intercept)   38.8414    1.2604   4.2442  30.817
## day_n         -1.6014    0.1072  365.2333 -14.932
## day_n:humidity_tmtDry    0.2226    0.1423  387.7685  1.565
## day_n:temp_tmtCool      0.4835    0.1414  387.7848  3.419
## day_n:humidity_tmtDry:temp_tmtCool -0.2278    0.1991  387.7880 -1.144
##              Pr(>|t|)
## (Intercept)   3.74e-06 ***
## day_n         < 2e-16 ***
## day_n:humidity_tmtDry    0.118463
## day_n:temp_tmtCool      0.000695 ***
## day_n:humidity_tmtDry:temp_tmtCool 0.253206
## ---
```



```

## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) day_n  dy_:_D dy_:_C
## day_n      -0.104
## dy_n:hmdt_D  0.000 -0.684
## dy_n:tmp_tC  0.001 -0.688  0.518
## dy_n:h_D:_C  0.000  0.488 -0.713 -0.710
hct_stats_full <- broom.mixed::tidy(hct_mod_full) %>%
  mutate(response = "Hematocrit (%)")

Save best model stats (2b) with p-values:
hct_mod_best1 <- lmerTest::lmer(data = dat_no_rehab,
                               hematocrit_percent ~
                                day_n + day_n:temp_tmt +
                                (1|trial_number/individual_ID))
summary(hct_mod_best1)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## hematocrit_percent ~ day_n + day_n:temp_tmt + (1 | trial_number/individual_ID)
## Data: dat_no_rehab
##
## REML criterion at convergence: 2384.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.3033 -0.5249 -0.0914  0.5361  2.7645
##
## Random effects:
## Groups              Name                Variance Std.Dev.
## individual_ID:trial_number (Intercept) 12.302   3.507
## trial_number              (Intercept)  7.127   2.670
## Residual                  14.926   3.863
## Number of obs: 398, groups: individual_ID:trial_number, 134; trial_number, 5
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   38.84092    1.27026   4.24212  30.577 3.88e-06 ***
## day_n         -1.48653    0.07824  330.30598  -19.001 < 2e-16 ***
## day_n:temp_tmtCool  0.36596    0.09961  389.73358   3.674 0.000272 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) day_n
## day_n      -0.141
## dy_n:tmp_tC  0.001 -0.650

hct_stats <- broom.mixed::tidy(hct_mod_best1) %>%
  mutate(response = "Hematocrit (%)")

```

Osmolality

Building

```
osml_mod4 <- lme4::lmer(data = dat_no_rehab,
                        osmolality_mmol_kg_mean ~
                          day_n +
                          day_n:humidity_tmt + day_n:temp_tmt +
                          day_n:humidity_tmt:temp_tmt +
                          (1|trial_number/individual_ID))
car::vif(osml_mod4) # check collinearity

##              day_n              day_n:humidity_tmt
##              2.233263              2.789683
##              day_n:temp_tmt day_n:humidity_tmt:temp_tmt
##              2.789038              3.339336

summary(osml_mod4)

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## osmolality_mmol_kg_mean ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
##   day_n:humidity_tmt:temp_tmt + (1 | trial_number/individual_ID)
##   Data: dat_no_rehab
##
## REML criterion at convergence: 3568.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.4598 -0.7223 -0.1609  0.6130  4.0596
##
## Random effects:
##   Groups                Name            Variance Std.Dev.
## individual_ID:trial_number (Intercept)    11.7      3.420
## trial_number              (Intercept)    11.9      3.449
## Residual                      535.3     23.136
## Number of obs: 391, groups:  individual_ID:trial_number, 134; trial_number, 5
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)    351.0936    2.4146 145.403
## day_n           0.4282    0.5394   0.794
## day_n:humidity_tmtDry    1.0186    0.6613   1.540
## day_n:temp_tmtCool    -0.6326    0.6609  -0.957
## day_n:humidity_tmtDry:temp_tmtCool -0.9411    0.9401  -1.001
##
## Correlation of Fixed Effects:
##              (Intr) day_n  dy_:_D dy_:_C
## day_n          -0.385
## dy_n:hmdt_D    -0.004 -0.607
## dy_n:tmp_tC    -0.004 -0.606  0.496
## dy_n:h_D:_C    0.002  0.426 -0.702 -0.703

drop1(osml_mod4)

## Single term deletions
```

```
##
## Model:
## osmolality_mmol_kg_mean ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
##   day_n:humidity_tmt:temp_tmt + (1 | trial_number/individual_ID)
##               npar      AIC
## <none>                3590.1
## day_n:humidity_tmt:temp_tmt    1 3589.2
```

Dropping the three-way interaction slightly improves AIC, so test that:

```
osml_mod3a <- lme4::lmer(data = dat_no_rehab,
                        osmolality_mmol_kg_mean ~
                          day_n +
                          day_n:humidity_tmt + day_n:temp_tmt +
                          (1|trial_number/individual_ID))
car::vif(osml_mod3a) # check collinearity
```

```
##               day_n day_n:humidity_tmt    day_n:temp_tmt
##             1.831913             1.415181             1.412337
```

```
anova(osml_mod4, osml_mod3a)
```

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

```
## Data: dat_no_rehab
```

```
## Models:
```

```
## osml_mod3a: osmolality_mmol_kg_mean ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
## osml_mod3a:      (1 | trial_number/individual_ID)
## osml_mod4: osmolality_mmol_kg_mean ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
## osml_mod4:      day_n:humidity_tmt:temp_tmt + (1 | trial_number/individual_ID)
##               npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## osml_mod3a      7 3589.2 3616.9 -1787.6   3575.2
## osml_mod4      8 3590.1 3621.9 -1787.1   3574.1 1.0118  1    0.3145
```

We do NOT lose significant predictive power, so the three-way interaction should be dropped.

Test whether we should drop any remaining variables:

```
osml_mod2b <- lme4::lmer(data = dat_no_rehab,
                        osmolality_mmol_kg_mean ~
                          day_n +
                          day_n:temp_tmt +
                          (1|trial_number/individual_ID))
anova(osml_mod3a, osml_mod2b)
```

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

```
## Data: dat_no_rehab
```

```
## Models:
```

```
## osml_mod2b: osmolality_mmol_kg_mean ~ day_n + day_n:temp_tmt + (1 | trial_number/individual_ID)
## osml_mod3a: osmolality_mmol_kg_mean ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
## osml_mod3a:      (1 | trial_number/individual_ID)
##               npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## osml_mod2b      6 3588.5 3612.3 -1788.3   3576.5
## osml_mod3a      7 3589.2 3616.9 -1787.6   3575.2 1.3627  1    0.2431
```

We do NOT lose significant predictive power if we drop humidity.

```
osml_mod2c <- lme4::lmer(data = dat_no_rehab,
                        osmolality_mmol_kg_mean ~
```

```

        day_n +
        day_n:humidity_tmt +
        (1|trial_number/individual_ID))
anova(osml_mod3a, osml_mod2c)

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

## Data: dat_no_rehab
## Models:
## osml_mod2c: osmolality_mmol_kg_mean ~ day_n + day_n:humidity_tmt + (1 | trial_number/individual_ID)
## osml_mod3a: osmolality_mmol_kg_mean ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
## osml_mod3a:      (1 | trial_number/individual_ID)
##           npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## osml_mod2c      6 3592.6 3616.4 -1790.3   3580.6
## osml_mod3a      7 3589.2 3616.9 -1787.6   3575.2 5.4558  1    0.0195 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

We lose significant predictive power if we drop temperature.

Selection

Compare with/without three-way interaction and humidity:

```

osml_models <- list(osml_mod4, osml_mod3a, osml_mod2b, osml_mod2c)

EXP_mod_names2 <- data.frame(predictors =
  c('Day, Day:Humd, Day:Temp, Day:Humd:Temp', #4
    'Day, Day:Humd, Day:Temp', #3a
    'Day, Day:Temp', #2b
    'Day, Day:Humd' #2c
  ))
rownames(EXP_mod_names2) <- c('4', '3a', '2b', '2c')
#calculate AIC of each model
osml_AICc <- data.frame(aictab(cand.set = osml_models,
  modnames = EXP_mod_names2)) %>%
  mutate(response = "Plasma Osmolality (mmol/kg)")

```

```

## Warning in aictab.AIClmerMod(cand.set = osml_models, modnames = EXP_mod_names2):
## Model selection for fixed effects is only appropriate with ML estimation:
## REML (default) should only be used to select random effects for a constant set of fixed effects

```

```

osml_AICc

##           predictors K      AICc Delta_AICc Modellik
## 1 Day, Day:Humd, Day:Temp, Day:Humd:Temp 8 3585.291  0.0000000 1.0000000
## 2           Day, Day:Temp 6 3585.562  0.2703096 0.8735807
## 3           Day, Day:Humd, Day:Temp 7 3585.921  0.6301657 0.7297284
## 4           Day, Day:Humd 6 3589.605  4.3142933 0.1156547
##      AICcWt  Res.LL  Cum.Wt      response
## 1 0.3677872 -1784.457 0.3677872 Plasma Osmolality (mmol/kg)
## 2 0.3212918 -1786.671 0.6890789 Plasma Osmolality (mmol/kg)
## 3 0.2683848 -1785.814 0.9574637 Plasma Osmolality (mmol/kg)
## 4 0.0425363 -1788.693 1.0000000 Plasma Osmolality (mmol/kg)

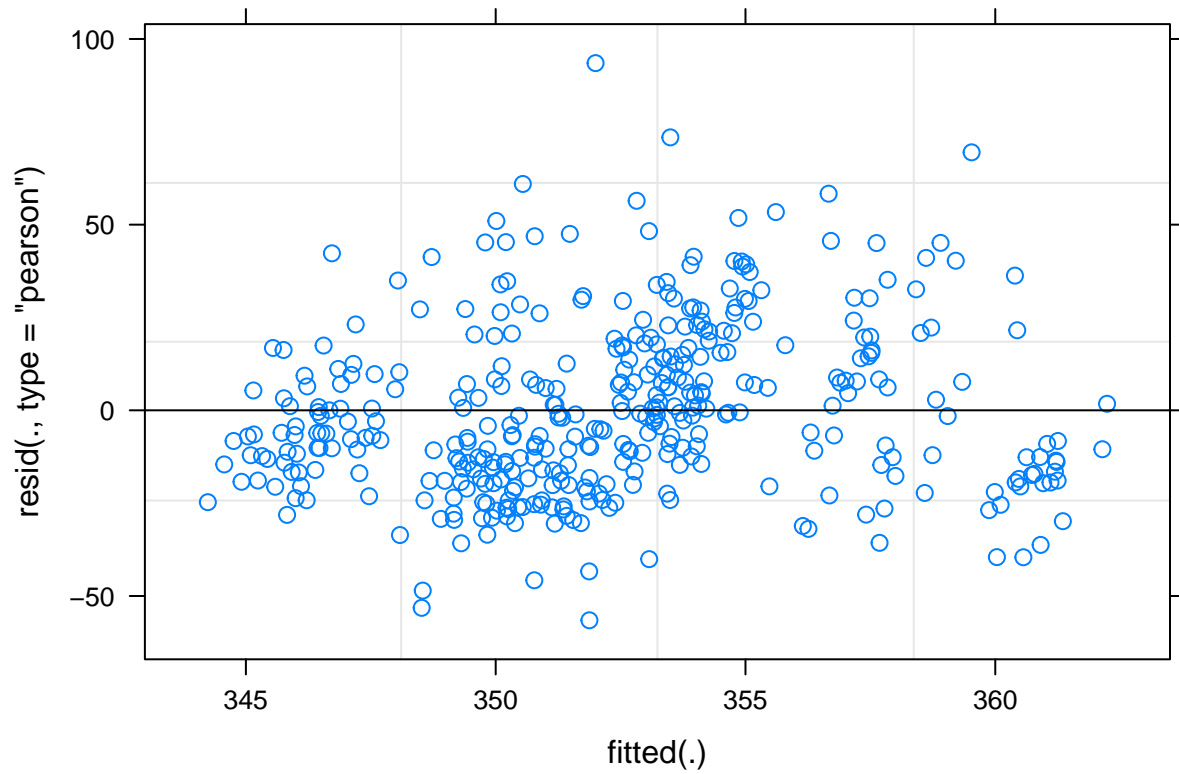
```

They're all exactly equal, with the full model having SLIGHTLY lower AIC. Since they're all equal, we will use the model with the fewest predictors (2b with day and temp only).

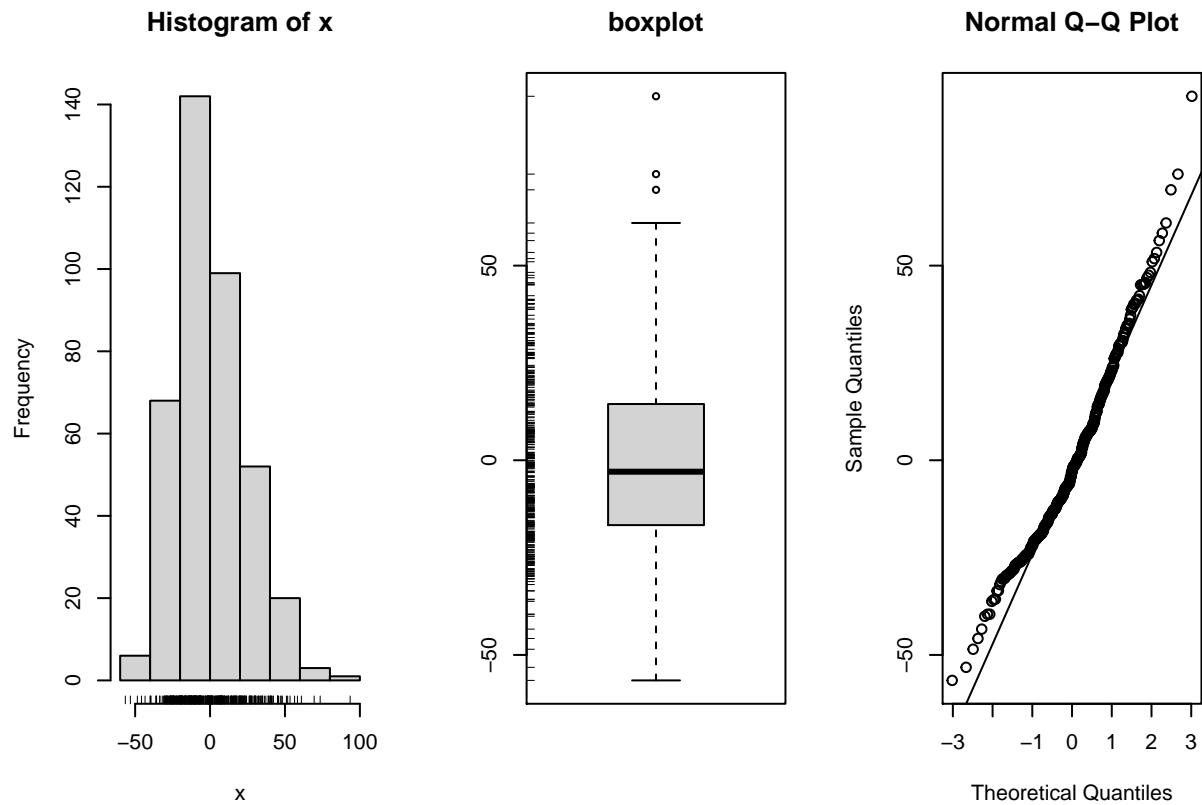
Assumptions

Check linear regression assumptions/conditions:

```
plot(osml_mod2b)
```



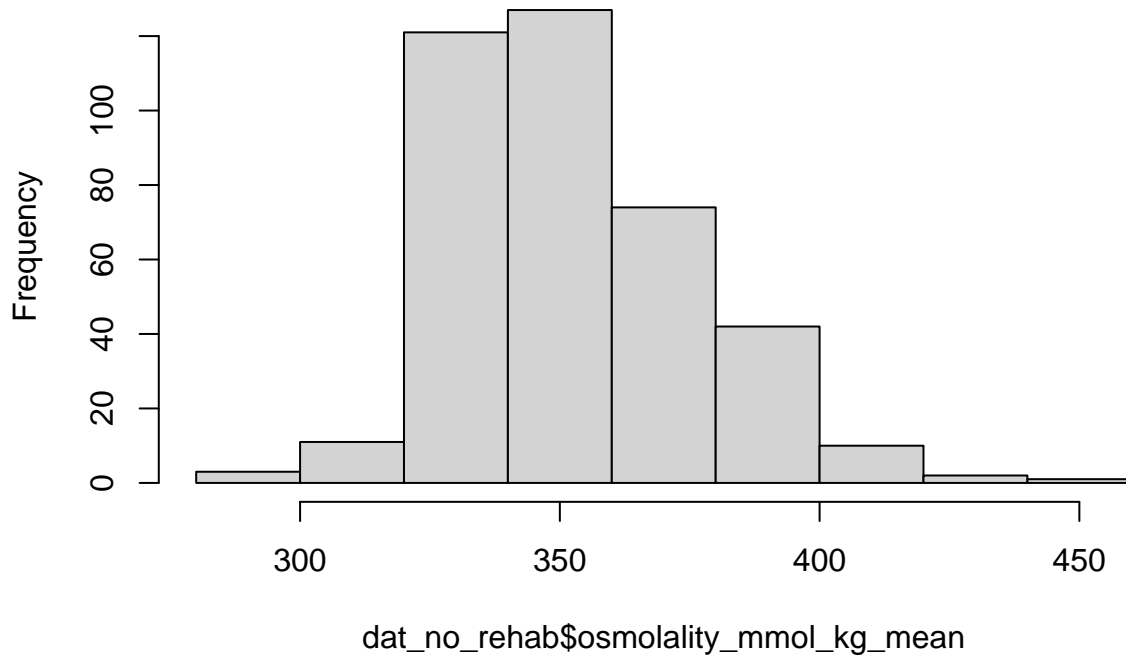
```
simple.eda(residuals(osml_mod2b))
```



```
shapiro.test(residuals(osml_mod2b))
```

```
##
## Shapiro-Wilk normality test
##
## data: residuals(osml_mod2b)
## W = 0.97335, p-value = 1.375e-06
hist(dat_no_rehab$osmolality_mmol_kg_mean)
```

Histogram of dat_no_rehab\$osmolality_mmol_kg_mean



All assumptions are OKAY. Normality of the residuals is statistically, but not necessarily visually, violated. Osmolality values are slightly skewed to the right, but not enough to transform the variable.

Save

Save full model stats with p-values:

```
osml_mod_full <- lmerTest::lmer(data = dat_no_rehab,
                                osmolality_mmol_kg_mean ~
                                  day_n +
                                  day_n:humidity_tmt + day_n:temp_tmt +
                                  day_n:humidity_tmt:temp_tmt +
                                  (1|trial_number/individual_ID))
summary(osml_mod_full)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## osmolality_mmol_kg_mean ~ day_n + day_n:humidity_tmt + day_n:temp_tmt +
##   day_n:humidity_tmt:temp_tmt + (1 | trial_number/individual_ID)
## Data: dat_no_rehab
##
## REML criterion at convergence: 3568.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.4598 -0.7223 -0.1609  0.6130  4.0596
##
## Random effects:
## Groups              Name                Variance Std.Dev.
```

```
## individual_ID:trial_number (Intercept) 11.7 3.420
## trial_number (Intercept) 11.9 3.449
## Residual 535.3 23.136
## Number of obs: 391, groups: individual_ID:trial_number, 134; trial_number, 5
##
## Fixed effects:
##
## Estimate Std. Error df t value
## (Intercept) 351.0936 2.4146 9.6731 145.403
## day_n 0.4282 0.5394 383.2904 0.794
## day_n:humidity_tmtDry 1.0186 0.6613 291.6737 1.540
## day_n:temp_tmtCool -0.6326 0.6609 290.9840 -0.957
## day_n:humidity_tmtDry:temp_tmtCool -0.9411 0.9401 290.6609 -1.001
##
## Pr(>|t|)
## (Intercept) <2e-16 ***
## day_n 0.428
## day_n:humidity_tmtDry 0.125
## day_n:temp_tmtCool 0.339
## day_n:humidity_tmtDry:temp_tmtCool 0.318
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
## (Intr) day_n dy_:_D dy_:_C
## day_n -0.385
## dy_n:hmdt_D -0.004 -0.607
## dy_n:tmp_tC -0.004 -0.606 0.496
## dy_n:h_D:_C 0.002 0.426 -0.702 -0.703
```

```
osml_stats_full <- broom.mixed::tidy(osml_mod_full) %>%
  mutate(response = "Plasma Osmolality (mmol/kg)")
```

Save best model stats with p-values:

```
osml_mod_best1 <- lmerTest::lmer(data = dat_no_rehab,
  osmolality_mmol_kg_mean ~
    day_n +
    day_n:temp_tmt +
    (1|trial_number/individual_ID))
summary(osml_mod_best1)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## osmolality_mmol_kg_mean ~ day_n + day_n:temp_tmt + (1 | trial_number/individual_ID)
## Data: dat_no_rehab
##
## REML criterion at convergence: 3573.3
##
## Scaled residuals:
## Min 1Q Median 3Q Max
## -2.4432 -0.7196 -0.1269 0.6244 4.0401
##
## Random effects:
## Groups Name Variance Std.Dev.
## individual_ID:trial_number (Intercept) 12.13 3.482
```



```
## trial_number          (Intercept) 11.32    3.364
## Residual              535.60    23.143
## Number of obs: 391, groups:  individual_ID:trial_number, 134; trial_number, 5
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   351.1010     2.3914    9.8996 146.815  <2e-16 ***
## day_n         0.9324      0.4289   347.3548   2.174  0.0304 *
## day_n:temp_tmtCool -1.1010     0.4705   293.2875  -2.340  0.0199 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) day_n
## day_n        -0.492
## dy_n:tmp_tC -0.003 -0.540

osml_stats <- broom.mixed::tidy(osml_mod_best1) %>%
  mutate(response = "Plasma Osmolality (mmol/kg)")
```

CEWL

Building

```
# get df with only CEWL values in it
dat_no_rehab_rm <- dat_no_rehab %>%
  dplyr::filter(complete.cases(CEWL_g_m2h_mean))
CEWL_mod4 <- lme4::lmer(data = dat_no_rehab_rm,
  CEWL_g_m2h_mean ~ cloacal_temp_C +
    day_n +
    day_n:humidity_tmt + day_n:temp_tmt +
    day_n:humidity_tmt:temp_tmt +
    (1|trial_number/individual_ID))

## boundary (singular) fit: see ?isSingular
summary(CEWL_mod4)

## Linear mixed model fit by REML ['lmerMod']
## Formula: CEWL_g_m2h_mean ~ cloacal_temp_C + day_n + day_n:humidity_tmt +
##          day_n:temp_tmt + day_n:humidity_tmt:temp_tmt + (1 | trial_number/individual_ID)
## Data: dat_no_rehab_rm
##
## REML criterion at convergence: 1666
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.7253 -0.6161 -0.0815  0.5676  3.9063
##
## Random effects:
##      Groups              Name      Variance Std.Dev.
## individual_ID:trial_number (Intercept)  0.000    0.000
## trial_number              (Intercept)  7.839    2.800
## Residual                  28.402    5.329
## Number of obs: 267, groups:  individual_ID:trial_number, 134; trial_number, 5
```

```
##
## Fixed effects:
##
##               Estimate Std. Error t value
## (Intercept)      55.9225    9.5233   5.872
## cloacal_temp_C    -1.3274    0.3564  -3.724
## day_n              1.7338    0.1451  11.952
## day_n:humidity_tmtDry -1.3979    0.1653  -8.456
## day_n:temp_tmtCool  -0.7447    0.1643  -4.532
## day_n:humidity_tmtDry:temp_tmtCool  0.6603    0.2312   2.856
##
## Correlation of Fixed Effects:
##      (Intr) clc__C day_n  dy_:_D dy_:_C
## clocl_tmp_C -0.990
## day_n        -0.465  0.450
## dy_n:hmdt_D  0.165 -0.167 -0.638
## dy_n:tmp_tC  0.059 -0.060 -0.591  0.506
## dy_n:h_D:_C -0.008  0.008  0.405 -0.695 -0.709
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

We can and should drop the random effect of individual ID, since it accounts for zero variation.

```
CEWL_mod4.2 <- lme4::lmer(data = dat_no_rehab_rm,
  CEWL_g_m2h_mean ~ cloacal_temp_C +
    day_n +
    day_n:humidity_tmt + day_n:temp_tmt +
    day_n:humidity_tmt:temp_tmt +
    (1|trial_number))
summary(CEWL_mod4.2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: CEWL_g_m2h_mean ~ cloacal_temp_C + day_n + day_n:humidity_tmt +
##      day_n:temp_tmt + day_n:humidity_tmt:temp_tmt + (1 | trial_number)
##      Data: dat_no_rehab_rm
##
## REML criterion at convergence: 1666
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.7253 -0.6161 -0.0815  0.5676  3.9063
##
## Random effects:
##      Groups       Name             Variance Std.Dev.
## trial_number (Intercept)  7.839      2.800
## Residual                28.402      5.329
## Number of obs: 267, groups: trial_number, 5
##
## Fixed effects:
##
##               Estimate Std. Error t value
## (Intercept)      55.9226    9.5233   5.872
## cloacal_temp_C    -1.3274    0.3564  -3.724
## day_n              1.7338    0.1451  11.952
## day_n:humidity_tmtDry -1.3979    0.1653  -8.456
## day_n:temp_tmtCool  -0.7447    0.1643  -4.532
## day_n:humidity_tmtDry:temp_tmtCool  0.6603    0.2312   2.856
```

```
##
## Correlation of Fixed Effects:
##          (Intr) clc__C day_n dy_:_D dy_:_C
## clocl_tmp_C -0.990
## day_n       -0.465  0.450
## dy_n:hmdt_D 0.165 -0.167 -0.638
## dy_n:tmp_tC 0.059 -0.060 -0.591  0.506
## dy_n:h_D:_C -0.008  0.008  0.405 -0.695 -0.709
drop1(CEWL_mod4.2)

## Single term deletions
##
## Model:
## CEWL_g_m2h_mean ~ cloacal_temp_C + day_n + day_n:humidity_tmt +
##   day_n:temp_tmt + day_n:humidity_tmt:temp_tmt + (1 | trial_number)
##               npar      AIC
## <none>                1674.8
## cloacal_temp_C         1 1686.1
## day_n:humidity_tmt:temp_tmt 1 1681.0
```

```
car::vif(CEWL_mod4.2) # check collinearity
```

```
##               cloacal_temp_C               day_n
##               1.390398               3.164979
##   day_n:humidity_tmt               day_n:temp_tmt
##               3.080182               3.022880
## day_n:humidity_tmt:temp_tmt
##               3.473830
```

All variables have really big t-values and AIC cannot be improved by dropping anything.

But, check 1-variable drop:

```
CEWL_mod3a <- lme4::lmer(data = dat_no_rehab_rm,
  CEWL_g_m2h_mean ~ cloacal_temp_C +
    day_n +
    day_n:humidity_tmt + day_n:temp_tmt +
    (1|trial_number))
anova(CEWL_mod4.2, CEWL_mod3a)
```

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

```
## Data: dat_no_rehab_rm
## Models:
## CEWL_mod3a: CEWL_g_m2h_mean ~ cloacal_temp_C + day_n + day_n:humidity_tmt +
## CEWL_mod3a:   day_n:temp_tmt + (1 | trial_number)
## CEWL_mod4.2: CEWL_g_m2h_mean ~ cloacal_temp_C + day_n + day_n:humidity_tmt +
## CEWL_mod4.2:   day_n:temp_tmt + day_n:humidity_tmt:temp_tmt + (1 | trial_number)
##               npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## CEWL_mod3a      7 1681.0 1706.1 -833.47   1667.0
## CEWL_mod4.2     8 1674.8 1703.5 -829.38   1658.8 8.179  1  0.004238 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
car::vif(CEWL_mod3a) # check collinearity
```

```
##       cloacal_temp_C       day_n day_n:humidity_tmt       day_n:temp_tmt
```

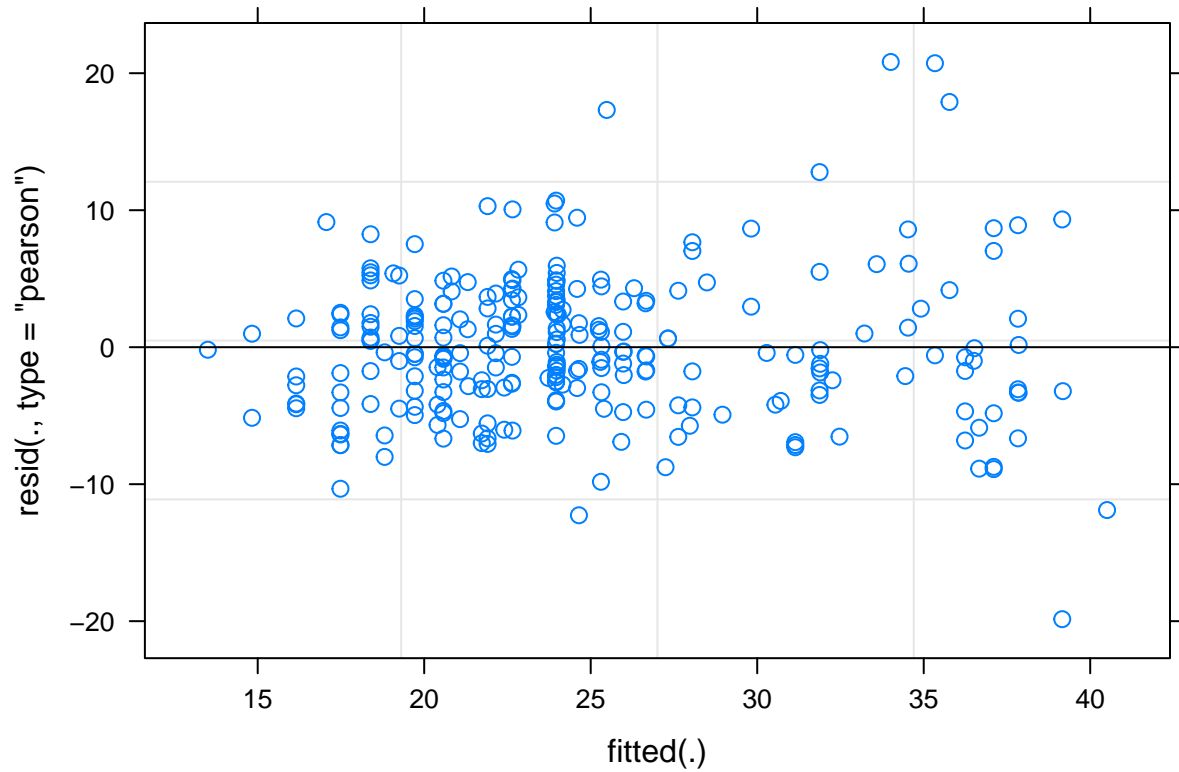
```
##          1.390133          2.646403          1.592525          1.503385
```

We lose significant predictive power, so we should NOT drop the three-way interaction.

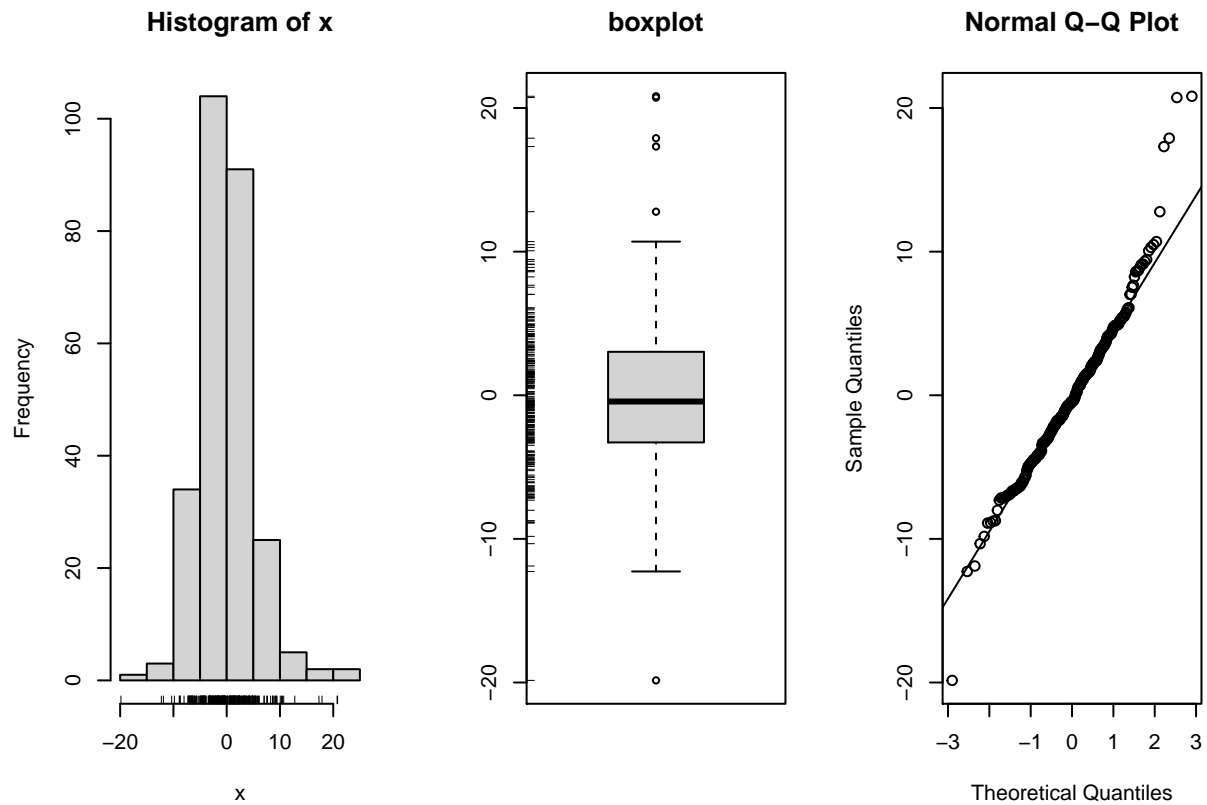
Assumptions

Check linear regression assumptions/conditions:

```
plot(CEWL_mod4.2)
```



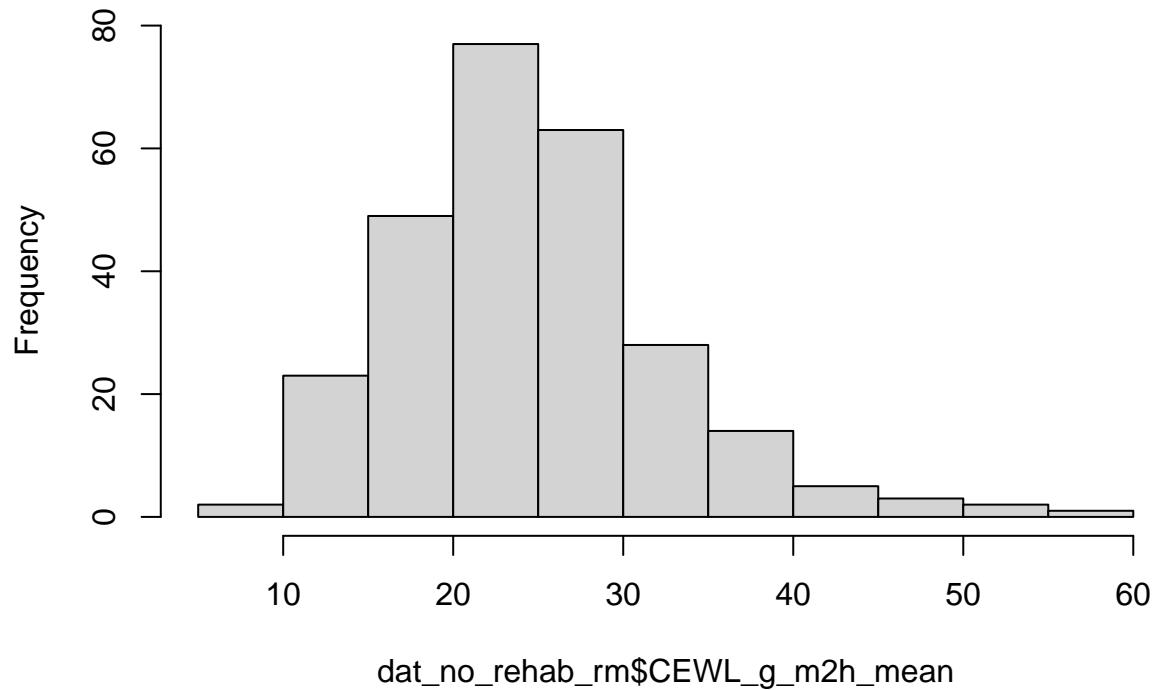
```
simple.eda(residuals(CEWL_mod4.2))
```



```
shapiro.test(residuals(CEWL_mod4.2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(CEWL_mod4.2)
## W = 0.96724, p-value = 8.744e-06
hist(dat_no_rehab_rm$CEWL_g_m2h_mean)
```

Histogram of dat_no_rehab_rm\$CEWL_g_m2h_mean



normality is not good, both other assumptions OKAY enough.

Save

Save best model stats with p-values:

```
CEWL_mod_best <- lmerTest::lmer(data = dat_no_rehab,
                                CEWL_g_m2h_mean ~ cloacal_temp_C +
                                day_n +
                                day_n:humidity_tmt + day_n:temp_tmt +
                                day_n:humidity_tmt:temp_tmt +
                                (1|trial_number))
summary(CEWL_mod_best)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: CEWL_g_m2h_mean ~ cloacal_temp_C + day_n + day_n:humidity_tmt +
##          day_n:temp_tmt + day_n:humidity_tmt:temp_tmt + (1 | trial_number)
## Data: dat_no_rehab
##
## REML criterion at convergence: 1666
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.7253 -0.6161 -0.0815  0.5676  3.9063
##
## Random effects:
##  Groups       Name             Variance Std.Dev.
##  trial_number (Intercept)  7.839    2.800
##  Residual                28.402    5.329
```

```
## Number of obs: 267, groups: trial_number, 5
##
## Fixed effects:
##
##              Estimate Std. Error      df t value
## (Intercept)    55.9226    9.5233 257.7583   5.872
## cloacal_temp_C   -1.3274    0.3564 260.6308  -3.724
## day_n           1.7338    0.1451 258.1416  11.952
## day_n:humidity_tmtDry -1.3979    0.1653 257.2239  -8.456
## day_n:temp_tmtCool -0.7447    0.1643 256.9944  -4.532
## day_n:humidity_tmtDry:temp_tmtCool  0.6603    0.2312 257.0031   2.856
##
##              Pr(>|t|)
## (Intercept)    1.32e-08 ***
## cloacal_temp_C  0.00024 ***
## day_n          < 2e-16 ***
## day_n:humidity_tmtDry 2.09e-15 ***
## day_n:temp_tmtCool  8.98e-06 ***
## day_n:humidity_tmtDry:temp_tmtCool 0.00464 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) clc__C day_n  dy_:_D dy_:_C
## clocl_tmp_C -0.990
## day_n       -0.465  0.450
## dy_n:hmdt_D  0.165 -0.167 -0.638
## dy_n:tmp_tC  0.059 -0.060 -0.591  0.506
## dy_n:h_D:_C -0.008  0.008  0.405 -0.695 -0.709

CEWL_stats <- broom.mixed::tidy(CEWL_mod_best) %>%
  mutate(response = "CEWL (g/m2h)")
```

Group Export

Put all the best models and AICc rankings into one csv each

```
experiment_stats <- CEWL_stats %>%
  rbind(osml_stats) %>%
  rbind(hct_stats) %>%
  rbind(SMI_stats)
write.csv(experiment_stats,
  "./results_statistics/exp_models_BEST.csv")

exp_stats_full_secondary <- osml_stats_full %>% rbind(hct_stats_full)
write.csv(exp_stats_full_secondary,
  "./results_statistics/exp_models_full_additional.csv")

AICc_ranks <- osml_AICc %>%
  rbind(hct_AICc)
write.csv(AICc_ranks, "./results_statistics/exp_model_rankings.csv")
```

CEWL and body condition were best-predicted by the FULL model. Hematocrit was best-predicted by a reduced model, with delta-AIC of the other models >3.7 (AICc_ranks / exp_model_rankings). Osmolality was predicted equally well by the full and reduced models, so we present the reduced model with the least number of variables (AICc_ranks / exp_model_rankings). The full models for hct and osml are exported in a separate csv.

Effect Estimates (from coefficients)

I want to know what the change in the response variables was per day for each treatment group, on average.

So, first make a matrix expressing the 4 treatment groups with 0s (is not) and 1s (it is). For example, if the baseline states (1s) express Cool and Dry, then hot-dry would have a 0 for temperature and 1 for humidity.

```
tmt_matrix <- data.frame(tmt = c("Hot Humid", "Hot Dry",  
                                "Cool Humid", "Cool Dry"),  
                        temp_state = c(0,0,1,1),  
                        humid_state = c(0,1,0,1))  
  
tmt_matrix
```

```
##           tmt temp_state humid_state  
## 1  Hot Humid          0           0  
## 2   Hot Dry          0           1  
## 3 Cool Humid          1           0  
## 4   Cool Dry          1           1
```

Next, get all the coefficients:

```
# CEWL  
CEWL_day_n <- experiment_stats %>%  
  dplyr::filter(response == "CEWL (g/m2h)") %>%  
  dplyr::filter(term == "day_n") %>%  
  dplyr::select(estimate)  
CEWL_day_n_humidity_tmtDry <- experiment_stats %>%  
  dplyr::filter(response == "CEWL (g/m2h)") %>%  
  dplyr::filter(term == "day_n:humidity_tmtDry") %>%  
  dplyr::select(estimate)  
CEWL_day_n_temp_tmtCool <- experiment_stats %>%  
  dplyr::filter(response == "CEWL (g/m2h)") %>%  
  dplyr::filter(term == "day_n:temp_tmtCool") %>%  
  dplyr::select(estimate)  
CEWL_day_n_humidity_tmtDry_temp_tmtCool <- experiment_stats %>%  
  dplyr::filter(response == "CEWL (g/m2h)") %>%  
  dplyr::filter(term == "day_n:humidity_tmtDry:temp_tmtCool") %>%  
  dplyr::select(estimate)  
  
# plasma osmolality  
osml_day_n <- experiment_stats %>%  
  dplyr::filter(response == "Plasma Osmolality (mmol/kg)") %>%  
  dplyr::filter(term == "day_n") %>%  
  dplyr::select(estimate)  
osml_day_n_temp_tmtCool <- experiment_stats %>%  
  dplyr::filter(response == "Plasma Osmolality (mmol/kg)") %>%  
  dplyr::filter(term == "day_n:temp_tmtCool") %>%  
  dplyr::select(estimate)  
  
# hematocrit  
hct_day_n <- experiment_stats %>%  
  dplyr::filter(response == "Hematocrit (%)") %>%  
  dplyr::filter(term == "day_n") %>%  
  dplyr::select(estimate)  
hct_day_n_temp_tmtCool <- experiment_stats %>%  
  dplyr::filter(response == "Hematocrit (%)") %>%
```



```

dplyr::filter(term == "day_n:temp_tmtCool") %>%
dplyr::select(estimate)

# body condition
SMI_day_n <- experiment_stats %>%
  dplyr::filter(response == "Body Condition (g)") %>%
  dplyr::filter(term == "day_n") %>%
  dplyr::select(estimate)
SMI_day_n_humidity_tmtDry <- experiment_stats %>%
  dplyr::filter(response == "Body Condition (g)") %>%
  dplyr::filter(term == "day_n:humidity_tmtDry") %>%
  dplyr::select(estimate)
SMI_day_n_temp_tmtCool <- experiment_stats %>%
  dplyr::filter(response == "Body Condition (g)") %>%
  dplyr::filter(term == "day_n:temp_tmtCool") %>%
  dplyr::select(estimate)
SMI_day_n_humidity_tmtDry_temp_tmtCool <- experiment_stats %>%
  dplyr::filter(response == "Body Condition (g)") %>%
  dplyr::filter(term == "day_n:humidity_tmtDry:temp_tmtCool") %>%
  dplyr::select(estimate)

```

Now, use the character states to calculate estimated rates of change per day for each tmt group:

```

response_per_day <- tmt_matrix %>%
  mutate(CEWL_per_day = CEWL_day_n[[1]] +
    CEWL_day_n_humidity_tmtDry[[1]]*humid_state +
    CEWL_day_n_temp_tmtCool[[1]]*temp_state +
    CEWL_day_n_humidity_tmtDry_temp_tmtCool[[1]]*humid_state*temp_state,
    osml_per_day = osml_day_n[[1]] +
    osml_day_n_temp_tmtCool[[1]]*temp_state,
    hct_per_day = hct_day_n[[1]] +
    hct_day_n_temp_tmtCool[[1]]*temp_state,
    SMI_per_day = SMI_day_n[[1]] +
    SMI_day_n_humidity_tmtDry[[1]]*humid_state +
    SMI_day_n_temp_tmtCool[[1]]*temp_state +
    SMI_day_n_humidity_tmtDry_temp_tmtCool[[1]]*humid_state*temp_state)
response_per_day

```

```

##          tmt temp_state humid_state CEWL_per_day osml_per_day hct_per_day
## 1 Hot Humid          0           0  1.7337987   0.9323703 -1.486534
## 2 Hot Dry           0           1  0.3359483   0.9323703 -1.486534
## 3 Cool Humid        1           0  0.9891075  -0.1686352 -1.120572
## 4 Cool Dry          1           1  0.2515608  -0.1686352 -1.120572
## SMI_per_day
## 1 -0.2386335
## 2 -0.3036765
## 3 -0.1230815
## 4 -0.1562379

```

```

write.csv(response_per_day, "./results_statistics/exp_change_per_day.csv")

```

Rehydration Models

SMI

```
SMI_rmod_best1 <- lmerTest::lmer(data = dat_for_rehab,
                                SMI ~ type +
                                (1|trial_number/individual_ID))
summary(SMI_rmod_best1)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: SMI ~ type + (1 | trial_number/individual_ID)
## Data: dat_for_rehab
##
## REML criterion at convergence: 765.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.76930 -0.51420  0.00365  0.49102  2.29491
##
## Random effects:
## Groups              Name      Variance Std.Dev.
## individual_ID:trial_number (Intercept) 1.4856   1.2189
## trial_number              (Intercept) 0.3707   0.6089
## Residual                  0.3004   0.5481
## Number of obs: 266, groups: individual_ID:trial_number, 134; trial_number, 5
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  10.11742    0.29624   3.99894  34.152 4.40e-06 ***
## typerehab      0.45207    0.06742  131.48613   6.705 5.41e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr)
## typerehab -0.112
```

Hematocrit

```
hct_rmod_best1 <- lmerTest::lmer(data = dat_for_rehab,
                                hematocrit_percent ~
                                type +
                                (1|trial_number/individual_ID))
summary(hct_rmod_best1)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: hematocrit_percent ~ type + (1 | trial_number/individual_ID)
## Data: dat_for_rehab
##
## REML criterion at convergence: 1609.1
##
## Scaled residuals:
```

```
##      Min      1Q   Median      3Q      Max
## -3.03283 -0.44653  0.00249  0.42448  2.45818
##
## Random effects:
##   Groups                Name      Variance Std.Dev.
## individual_ID:trial_number (Intercept) 16.156   4.019
## trial_number              (Intercept)  9.184   3.031
## Residual                  15.310   3.913
## Number of obs: 261, groups: individual_ID:trial_number, 134; trial_number, 5
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   28.381      1.442    4.193  19.679 2.73e-05 ***
## typerehab     -2.051      0.488  128.191  -4.203 4.91e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr)
## typerehab -0.167
```

Osmolality

```
osml_rmod_best1 <- lmerTest::lmer(data = dat_for_rehab,
                                osmolality_mmol_kg_mean ~
                                  type +
                                  (1|trial_number/individual_ID))
summary(osml_rmod_best1)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: osmolality_mmol_kg_mean ~ type + (1 | trial_number/individual_ID)
##   Data: dat_for_rehab
##
## REML criterion at convergence: 2355.4
##
## Scaled residuals:
##      Min      1Q   Median      3Q      Max
## -2.0006 -0.6411 -0.0988  0.5203  3.8059
##
## Random effects:
##   Groups                Name      Variance Std.Dev.
## individual_ID:trial_number (Intercept) 91.53   9.567
## trial_number              (Intercept) 159.65  12.635
## Residual                  513.33  22.657
## Number of obs: 255, groups: individual_ID:trial_number, 133; trial_number, 5
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  350.470      6.079    4.526  57.650 1.16e-07 ***
## typerehab      8.469      2.858  125.916   2.964 0.00364 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Correlation of Fixed Effects:
##           (Intr)
## typerehab -0.244
```

Group Export

```
rehydration_stats <- broom.mixed::tidy(osml_rmod_best1) %>%
  rbind(broom.mixed::tidy(hct_rmod_best1)) %>%
  rbind(broom.mixed::tidy(SMI_rmod_best1)) %>%
  mutate(response = c(rep("Plasma Osmolality (mmol/kg)", 5),
                      rep("Hematocrit (%)", 5),
                      rep("Body Condition (g)", 5)))
write.csv(rehydration_stats,
          "./results_statistics/rehydration_best_models.csv")
```

Figures

Means to Overlay

mean values per day per tmt group:

```
means <- dat %>% # use whole dat because want for both exp and rehyd
  group_by(day_n, tmt) %>%
  summarise(n_lizards = n(),
            mean_CEWL = mean(CEWL_g_m2h_mean, na.rm = T),
            sd_CEWL = sd(CEWL_g_m2h_mean, na.rm = T),
            mean_osml = mean(osmolality_mmol_kg_mean, na.rm = T),
            sd_osml = sd(osmolality_mmol_kg_mean, na.rm = T),
            mean_hct = mean(hematocrit_percent, na.rm = T),
            sd_hct = sd(hematocrit_percent, na.rm = T),
            mean_SMI = mean(SMI, na.rm = T),
            sd_SMI = sd(SMI, na.rm = T)) %>%
  mutate(se_CEWL = (sd_CEWL/sqrt(n_lizards)),
         se_osml = (sd_osml/sqrt(n_lizards)),
         se_hct = (sd_hct/sqrt(n_lizards)),
         se_SMI = (sd_SMI/sqrt(n_lizards)))

## `summarise()` regrouping output by 'day_n' (override with `.groups` argument)

means$mean_CEWL[is.nan(means$mean_CEWL)] <- NA
means$mean_osml[is.nan(means$mean_osml)] <- NA
means$mean_hct[is.nan(means$mean_hct)] <- NA
means$mean_SMI[is.nan(means$mean_SMI)] <- NA
means
```

```
## # A tibble: 28 x 15
## # Groups:   day_n [7]
##   day_n tmt      n_lizards mean_CEWL sd_CEWL mean_osml sd_osml mean_hct sd_hct
##   <dbl> <fct>         <int>    <dbl>  <dbl>    <dbl>  <dbl>  <dbl>  <dbl>
## 1     0 Hot Dry           34     20.9    5.93     347.   17.9    38.9    5.05
## 2     0 Hot Humid        33     21.4    4.85     347.   18.7    37.9    5.46
## 3     0 Cool Dry         33     20.0    6.08     346.   20.6    39.3    5.96
## 4     0 Cool Hum~       34     20.9    4.78     351.   20.3    39.6    5.30
## 5     4 Hot Dry          34      NA     NA      361.   25.8    33.1    5.00
## 6     4 Hot Humid       33      NA     NA      359.   22.5    32.0    5.38
```

```
## 7      4 Cool Dry      33      NA      NA      355.      27.0      35.0      7.02
## 8      4 Cool Hum~    34      NA      NA      356.      24.5      34.5      5.34
## 9      5 Hot Dry      34      NA      NA      NA      NA      NA      NA
## 10     5 Hot Humid    33      NA      NA      NA      NA      NA      NA
## # ... with 18 more rows, and 6 more variables: mean_SMI <dbl>, sd_SMI <dbl>,
## #   se_CEWL <dbl>, se_osml <dbl>, se_hct <dbl>, se_SMI <dbl>

# make only for days 0,8,10 for points to plot
pt_means <- means %>%
  dplyr::filter(day_n %in% c(0,4,8,10))

end_means <- means %>%
  dplyr::filter(day_n == 8)
write.csv(end_means, "./results_statistics/exp_end_means.csv")
```

SMI

Ind + Means

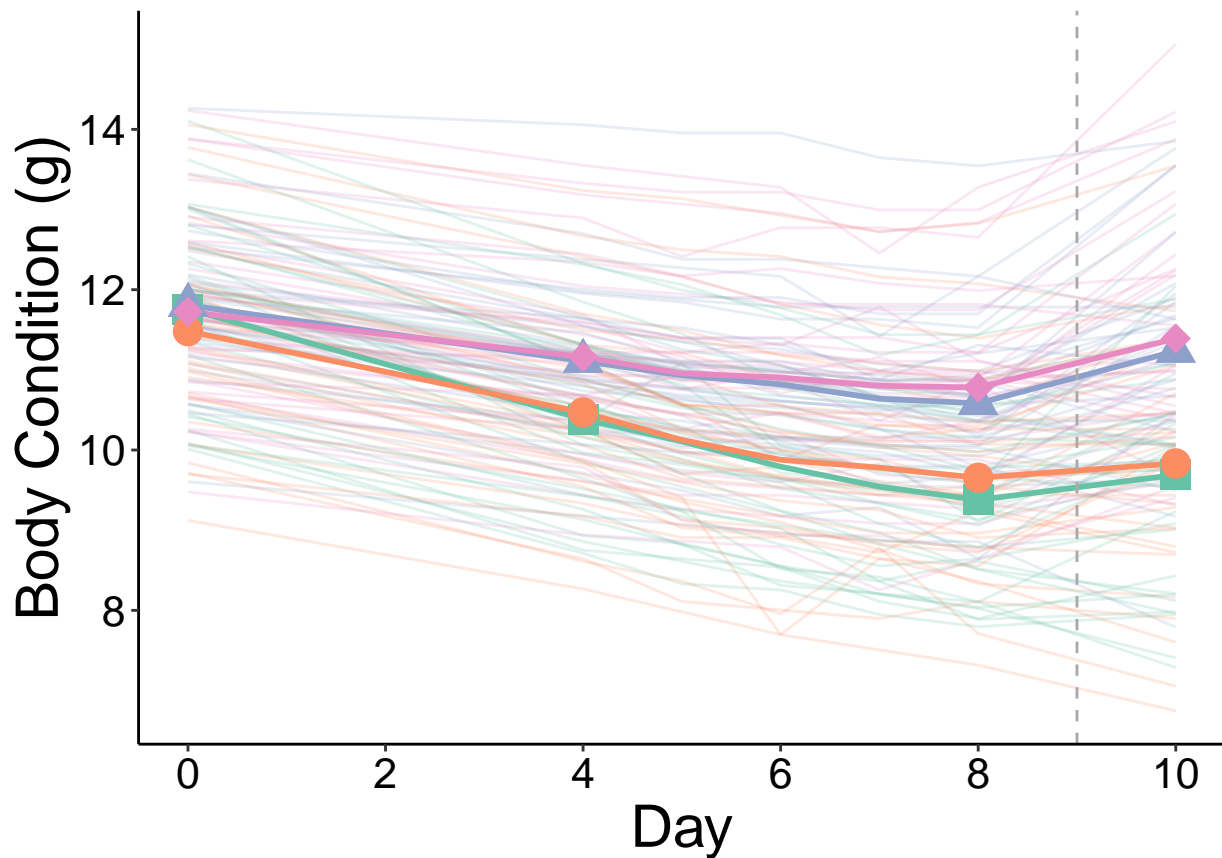
```
ggplot() +
  geom_line(data = dat,
    aes(x = day_n,
        y = SMI,
        color = tmt,
        group = individual_ID),
    alpha = 0.2) +
  geom_line(data = means,
    aes(x = day_n,
        y = mean_SMI,
        color = tmt,
        group = tmt),
    alpha = 1,
    size = 1) +
  geom_point(data = pt_means,
    aes(x = day_n,
        y = mean_SMI,
        color = tmt,
        shape = tmt),
    alpha = 1,
    size = 5) +
  geom_vline(xintercept = 9,
    linetype = "dashed",
    color = "darkgrey") +
  theme_classic() +
  scale_shape_manual(values = c(15:18), name = "") +
  scale_x_continuous(breaks = c(0, 2, 4, 6, 8, 10)) +
  scale_color_brewer(palette = "Set2", name = "") +
  xlab("Day") +
  ylab("Body Condition (g)") +
  guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
  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 = 22),
  legend.text.align = 0,
  legend.position = "none",
  plot.margin = unit(c(0.1, #top
                      0.1, #right
                      0.1, #bottom
                      0.41 #left
                      ), "cm")) -> SMI_fig

```

SMI_fig



```

#ggsave(filename = "SMI_fig1.jpeg",
#        plot = SMI_fig,
#        path = "./results_figures",
#        device = "jpeg",
#        dpi = 1200,
#        width = 6, height = 6)

```

Means Only

```

ggplot() +
  #plot these first so they end up on the "bottom"
  geom_errorbar(data = means,

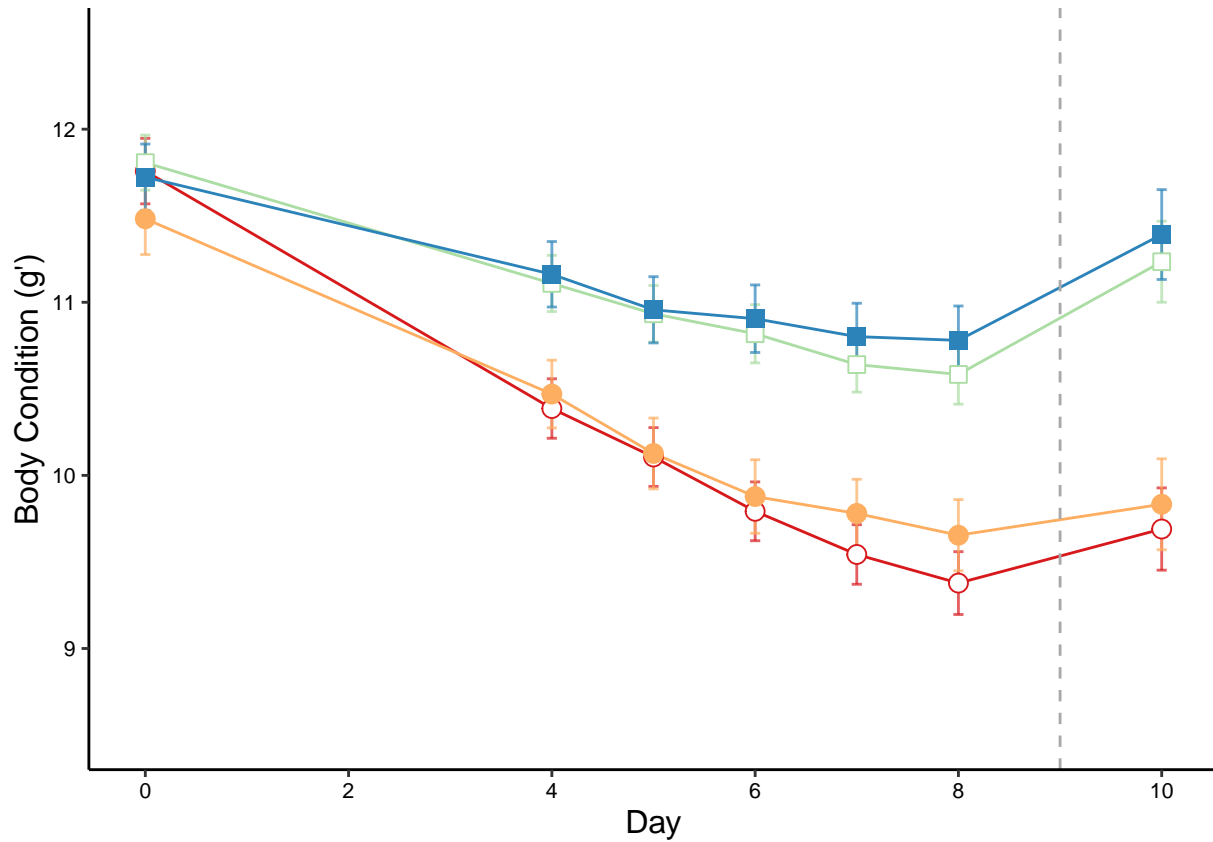
```

```

      aes(x = day_n,
        y = mean_SMI,
        color = tmt,
        group = tmt,
        ymin = mean_SMI-se_SMI,
        ymax = mean_SMI+se_SMI),
        width = .1,
        #position=position_dodge(.01),
        alpha = 0.7) +
geom_line(data = means,
  aes(x = day_n,
    y = mean_SMI,
    color = tmt,
    #linetype = tmt,
    group = tmt),
  alpha = 1,
  size = 0.5) +
geom_point(data = means,
  aes(x = day_n,
    y = mean_SMI,
    color = tmt,
    #fill = tmt,
    shape = tmt),
  fill = "white",
  alpha = 1,
  size = 3) +
theme_classic() +
scale_shape_manual(values = c(21,19, 22,15), name = "") +
#scale_linetype_manual(values = c("solid", "longdash", "solid", "longdash"),
#  name = "") +
scale_fill_brewer(palette = "Spectral", name = "") +
scale_color_brewer(palette = "Spectral", name = "") +
scale_x_continuous(breaks = c(0, 2, 4, 6, 8, 10)) +
scale_y_continuous(breaks = c(seq(8,12)),
  limits = c(8.5,12.5)) +
xlab("Day") +
ylab("Body Condition (g')") +
guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
geom_vline(xintercept = 9,
  linetype = "dashed",
  color = "darkgrey") +
theme(text = element_text(color = "black",
  family = "sans",
  size = 12),
  axis.text = element_text(color = "black",
    family = "sans",
    size = 8),
  legend.text = element_text(color = "black",
    family = "sans",
    size = 12),
  legend.text.align = 0,
  legend.position = "none",
  plot.margin = margin(t = 6, r = 6, b = 6, l = 10.8, unit = "pt"))

```

```
) -> SMI_fig_min
SMI_fig_min
```



LM + SE

```
ggplot() +
  geom_smooth(data = dat_no_rehab,
    aes(x = day_n,
        y = mass_g,
        color = tmt,
        group = tmt),
    formula = y ~ x,
    method = "lm",
    se = T,
    size = 1.2,
    alpha = 0.2) +
  theme_classic() +
  #scale_shape_manual(values = c(15:18), name = "") +
  scale_x_continuous(limits = c(0,8),
    breaks = c(0, 2, 4, 6, 8)) +
  scale_color_brewer(palette = "Set2", name = "") +
  xlab("Acclimation Day") +
  ylab("Lizard Mass (g)") +
  guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
  theme(text = element_text(color = "black",
    family = "sans",
```

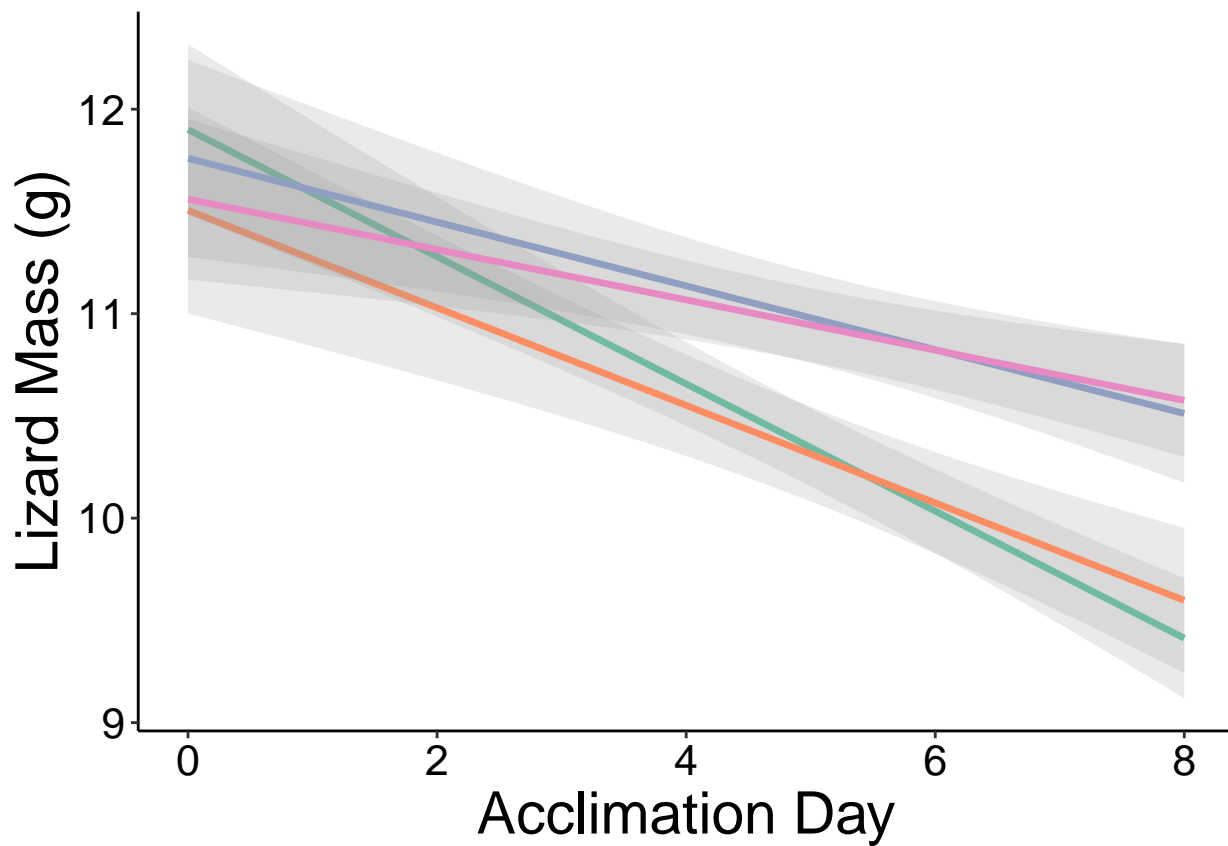


```

        size = 22),
axis.text = element_text(color = "black",
                          family = "sans",
                          size = 16),
legend.text = element_text(color = "black",
                           family = "sans",
                           size = 22),

legend.text.align = 0,
legend.position = "none"
) -> SMI_lm_fig
SMI_lm_fig

```



Ending Values Boxplot

```

end_vals_SMI <- dat %>%
  dplyr::filter(day_n == 8) %>%
  dplyr::filter(complete.cases(SMI))

SMI_aov <- summary(aov(data = end_vals_SMI,
  SMI ~ temp_tmt * humidity_tmt))
SMI_tukey <- TukeyHSD(aov(data = end_vals_SMI,
  SMI ~ temp_tmt * humidity_tmt))

ggplot(data = end_vals_SMI) +
  geom_boxplot(aes(x = tmt,
    y = SMI,

```

```

        color = tmt),
        size = 0.5,
        outlier.shape = NA) +
geom_jitter(aes(x = tmt,
                y = SMI,
                color = tmt,
                fill = tmt,
                shape = tmt),
            #fill = "white",
            size = 1,
            alpha = 0.4,
            position = position_jitter(height = 0, width = 0.2)) +
geom_point(data = end_means,
           aes(x = tmt,
               y = mean_SMI,
               color = tmt,
               fill = tmt),
           size = 3,
           shape = 23) +
theme_classic() +
scale_shape_manual(values = c(21,19, 22,15), name = "") +
scale_fill_brewer(palette = "Spectral", name = "") +
scale_color_brewer(palette = "Spectral", name = "") +
scale_y_continuous(limits = c(7,15),
                   breaks = c(seq(7,15, by = 2)),
                   labels = c(seq(7,15, by = 2))) +
xlab("") +

annotate(geom = "text", x = 1, y = 12.7, label = "A",
         size = 3) +
annotate(geom = "text", x = 1, y = 12.2, label = "9.4",
         size = 3) +
annotate(geom = "text", x = 2, y = 14.2, label = "A",
         size = 3) +
annotate(geom = "text", x = 2, y = 13.7, label = "9.7",
         size = 3) +
annotate(geom = "text", x = 3, y = 14.8, label = "B",
         size = 3) +
annotate(geom = "text", x = 3, y = 14.3, label = "10.6",
         size = 3) +
annotate(geom = "text", x = 4, y = 14.6, label = "B",
         size = 3) +
annotate(geom = "text", x = 4, y = 14.1, label = "10.8",
         size = 3) +

ylab("Body Condition (g')") +
theme(text = element_text(color = "black",
                          family = "sans",
                          size = 12),
      axis.text = element_text(color = "black",
                              family = "sans",
                              size = 8),
      #axis.text.x = element_blank(),

```

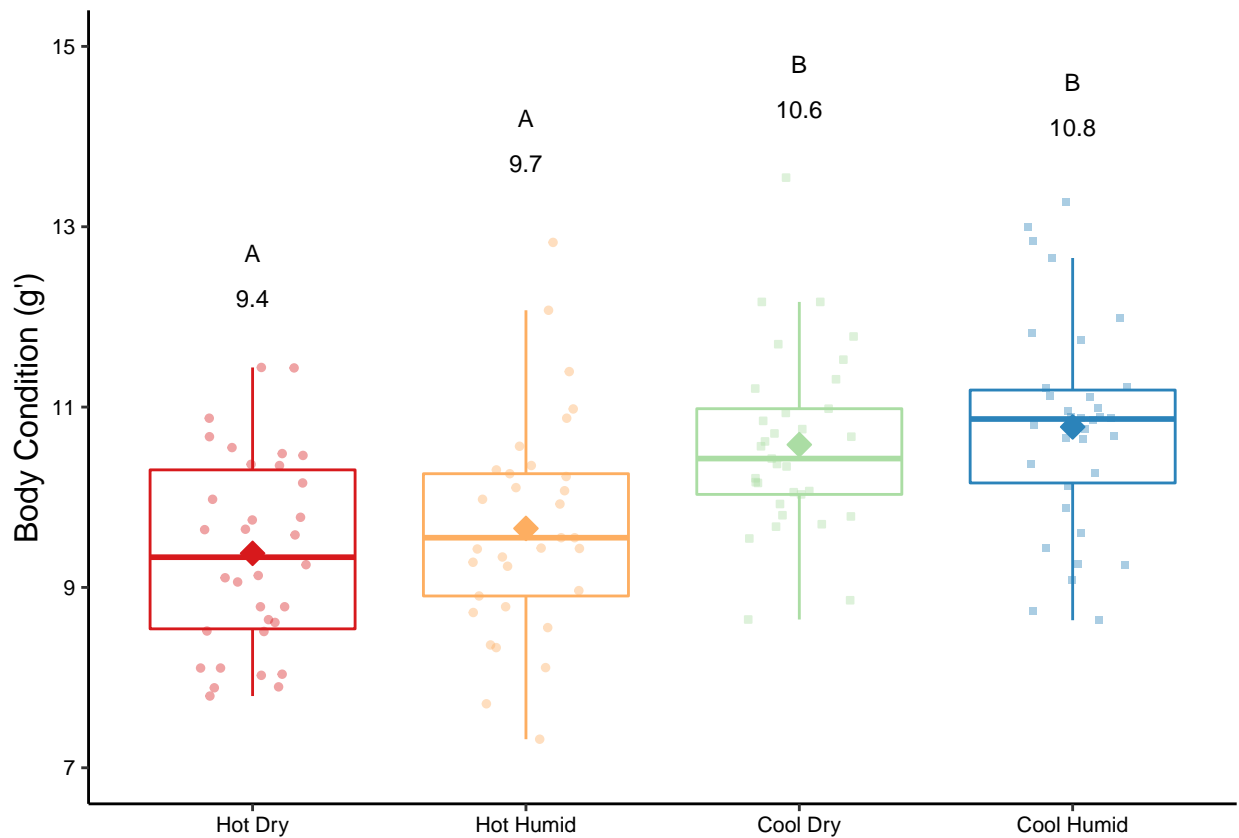
```

legend.text = element_text(color = "black",
                             family = "sans",
                             size = 12),

legend.text.align = 0,
legend.position = "none",
plot.margin = unit(c(0, #top
                     0, #right
                     0, #bottom
                     3.4), "mm")

) -> SMI_end_boxplot
SMI_end_boxplot

```



Hct

Ind + Means

```

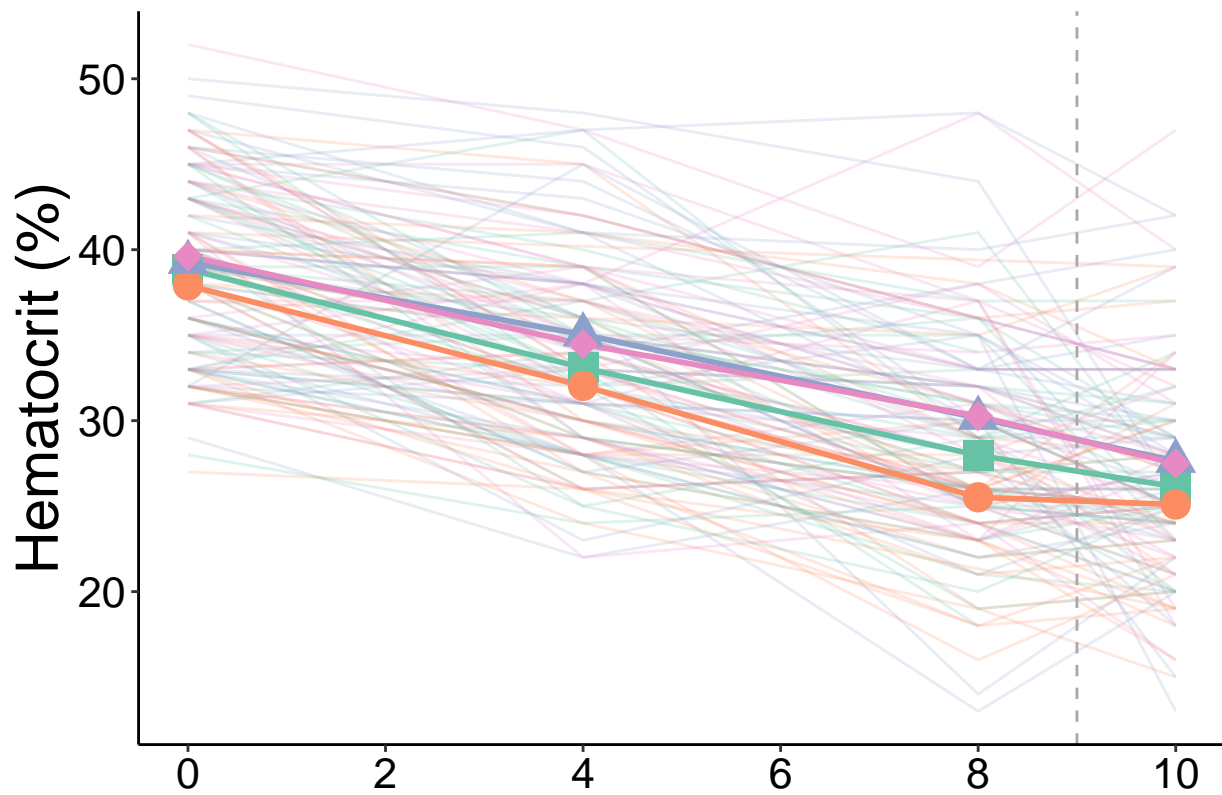
ggplot() +
  geom_line(data = dat[complete.cases(dat$hematocrit_percent),],
            aes(x = day_n,
                y = hematocrit_percent,
                color = tmt,
                group = individual_ID),
            alpha = 0.2) +
  geom_line(data = means[complete.cases(means$mean_hct),],
            aes(x = day_n,
                y = mean_hct,
                color = tmt,

```

```

      group = tmt),
      alpha = 1,
      size = 1) +
geom_point(data = pt_means,
      aes(x = day_n,
          y = mean_hct,
          color = tmt,
          shape = tmt),
      alpha = 1,
      size = 5) +
geom_vline(xintercept = 9,
      linetype = "dashed",
      color = "darkgrey") +
theme_classic() +
scale_shape_manual(values = c(15:18), name = "") +
scale_x_continuous(breaks = c(0, 2, 4, 6, 8, 10)) +
scale_color_brewer(palette = "Set2", name = "") +
xlab("") +
ylab("Hematocrit (%)") +
guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
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 = 22),
      legend.text.align = 0,
      legend.position = "none",
      plot.margin = unit(c(0.1, #top
      0.1, #right
      0.1, #bottom
      0.41 #left
      ), "cm")
) -> hct_fig
hct_fig

```



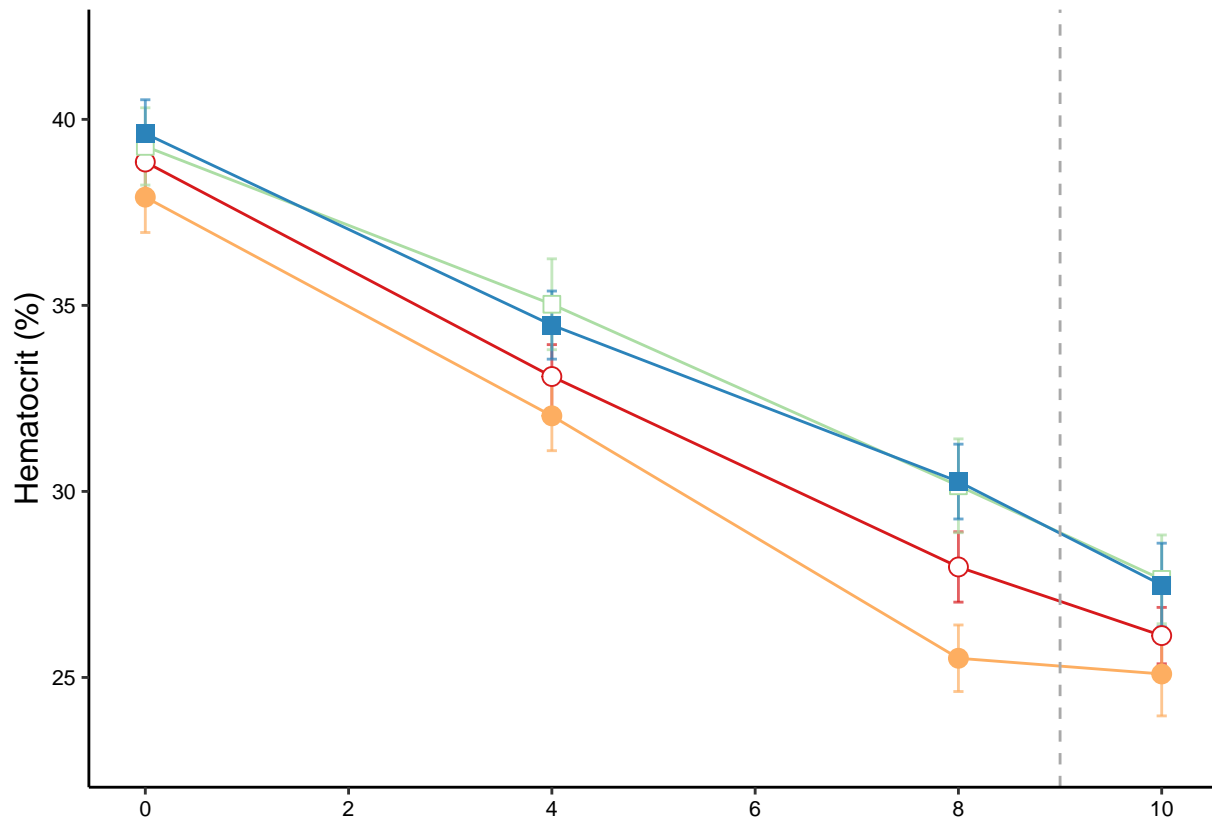
Means Only

```
ggplot() +
  geom_errorbar(data = means[complete.cases(means$mean_hct),],
    aes(x = day_n,
        y = mean_hct,
        color = tmt,
        group = tmt,
        ymin = mean_hct-se_hct,
        ymax = mean_hct+se_hct),
    width = .1,
    #position=position_dodge(.1),
    alpha = 0.7) +
  geom_line(data = means[complete.cases(means$mean_hct),],
    aes(x = day_n,
        y = mean_hct,
        color = tmt,
        #linetype = tmt,
        group = tmt),
    alpha = 1,
    size = 0.5) +
  geom_point(data = pt_means,
    aes(x = day_n,
        y = mean_hct,
        color = tmt,
        #fill = tmt,
        shape = tmt),
    alpha = 1,
```

```

        fill = "white",
        size = 3) +
theme_classic() +
scale_shape_manual(values = c(21,19, 22,15), name = "") +
#scale_linetype_manual(values = c("solid", "longdash", "solid", "longdash"),
#                          name = "") +
scale_fill_brewer(palette = "Spectral", name = "") +
scale_color_brewer(palette = "Spectral", name = "") +
scale_x_continuous(breaks = c(0, 2, 4, 6, 8, 10)) +
scale_y_continuous(breaks = c(25, 30, 35, 40),
                   limits = c(23,42)) +
xlab("") +
ylab("Hematocrit (%)") +
guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
geom_vline(xintercept = 9,
           linetype = "dashed",
           color = "darkgrey") +
theme(text = element_text(color = "black",
                           family = "sans",
                           size = 12),
      axis.text = element_text(color = "black",
                                family = "sans",
                                size = 8),
      legend.text = element_text(color = "black",
                                  family = "sans",
                                  size = 12),
      legend.text.align = 0,
      legend.position = "none",
      plot.margin = margin(t = 6, r = 6, b = 0, l = 10.8, unit = "pt")
) -> hct_fig_min
hct_fig_min

```



Ending Values Boxplot

```
end_vals_hct <- dat %>%
  dplyr::filter(day_n == 8) %>%
  dplyr::filter(complete.cases(hematocrit_percent))

hct_aov <- summary(aov(data = end_vals_hct,
  hematocrit_percent ~ temp_tmt * humidity_tmt))
hct_tukey <- TukeyHSD(aov(data = end_vals_hct,
  hematocrit_percent ~ temp_tmt * humidity_tmt))

ggplot(data = end_vals_hct) +
  geom_boxplot(aes(x = tmt,
    y = hematocrit_percent,
    color = tmt),
    size = 0.5,
    outlier.shape = NA) +
  geom_jitter(aes(x = tmt,
    y = hematocrit_percent,
    color = tmt,
    fill = tmt,
    shape = tmt),
    size = 1,
    alpha = 0.4,
    position = position_jitter(height = 0, width = 0.2)) +
  geom_point(data = end_means,
    aes(x = tmt,
```

```

        y = mean_hct,
        color = tmt,
        fill = tmt),
        size = 3,
        shape = 23) +
theme_classic() +
scale_shape_manual(values = c(21,19, 22,15), name = "") +
scale_fill_brewer(palette = "Spectral", name = "") +
scale_color_brewer(palette = "Spectral", name = "") +
xlab("") +
scale_y_continuous(limits = c(10,55),
                    breaks = c(seq(10,50, by = 10)),
                    labels = c(seq(10,50, by = 10))) +

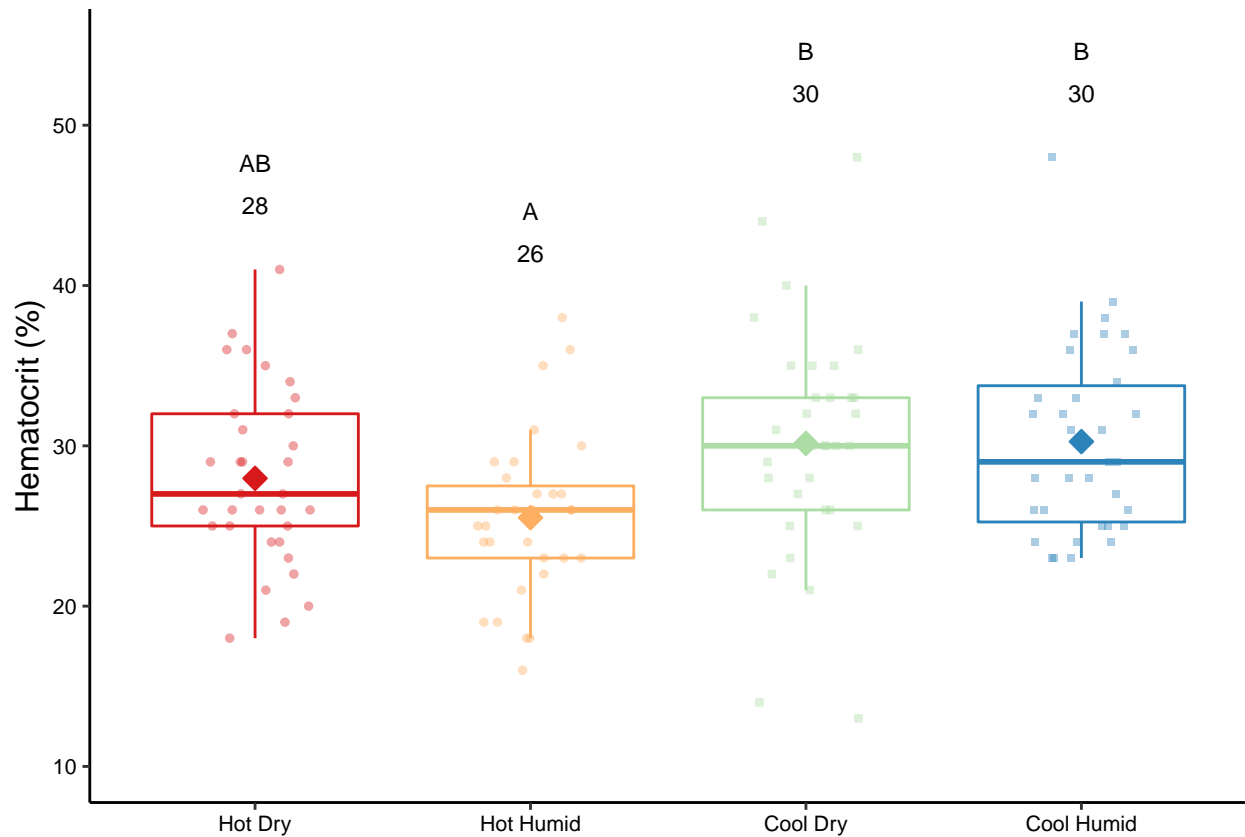
annotate(geom = "text", x = 1, y = 47.6, label = "AB",
         size = 3) +
annotate(geom = "text", x = 1, y = 45, label = "28",
         size = 3) +
annotate(geom = "text", x = 2, y = 44.6, label = "A",
         size = 3) +
annotate(geom = "text", x = 2, y = 42, label = "26",
         size = 3) +
annotate(geom = "text", x = 3, y = 54.6, label = "B",
         size = 3) +
annotate(geom = "text", x = 3, y = 52, label = "30",
         size = 3) +
annotate(geom = "text", x = 4, y = 54.6, label = "B",
         size = 3) +
annotate(geom = "text", x = 4, y = 52, label = "30",
         size = 3) +

ylab("Hematocrit (%)") +
theme(text = element_text(color = "black",
                           family = "sans",
                           size = 12),
      axis.text = element_text(color = "black",
                                family = "sans",
                                size = 8),
      legend.text = element_text(color = "black",
                                  family = "sans",
                                  size = 12),

      legend.text.align = 0,
      legend.position = "none",
      plot.margin = unit(c(0, #top
                           0, #right
                           0, #bottom
                           2.24), "mm"))

) -> hct_end_boxplot
hct_end_boxplot

```

Osml

Ind + Means

```
ggplot() +
  geom_line(data = dat[complete.cases(dat$osmolality_mmol_kg_mean),],
    aes(x = day_n,
        y = osmolality_mmol_kg_mean,
        color = tmt,
        group = individual_ID),
    alpha = 0.2) +
  geom_line(data = means[complete.cases(means$mean_osml),],
    aes(x = day_n,
        y = mean_osml,
        color = tmt,
        group = tmt),
    alpha = 1,
    size = 1) +
  geom_point(data = pt_means,
    aes(x = day_n,
        y = mean_osml,
        color = tmt,
        shape = tmt),
    alpha = 1,
    size = 5) +
  geom_vline(xintercept = 9,
    linetype = "dashed",
```

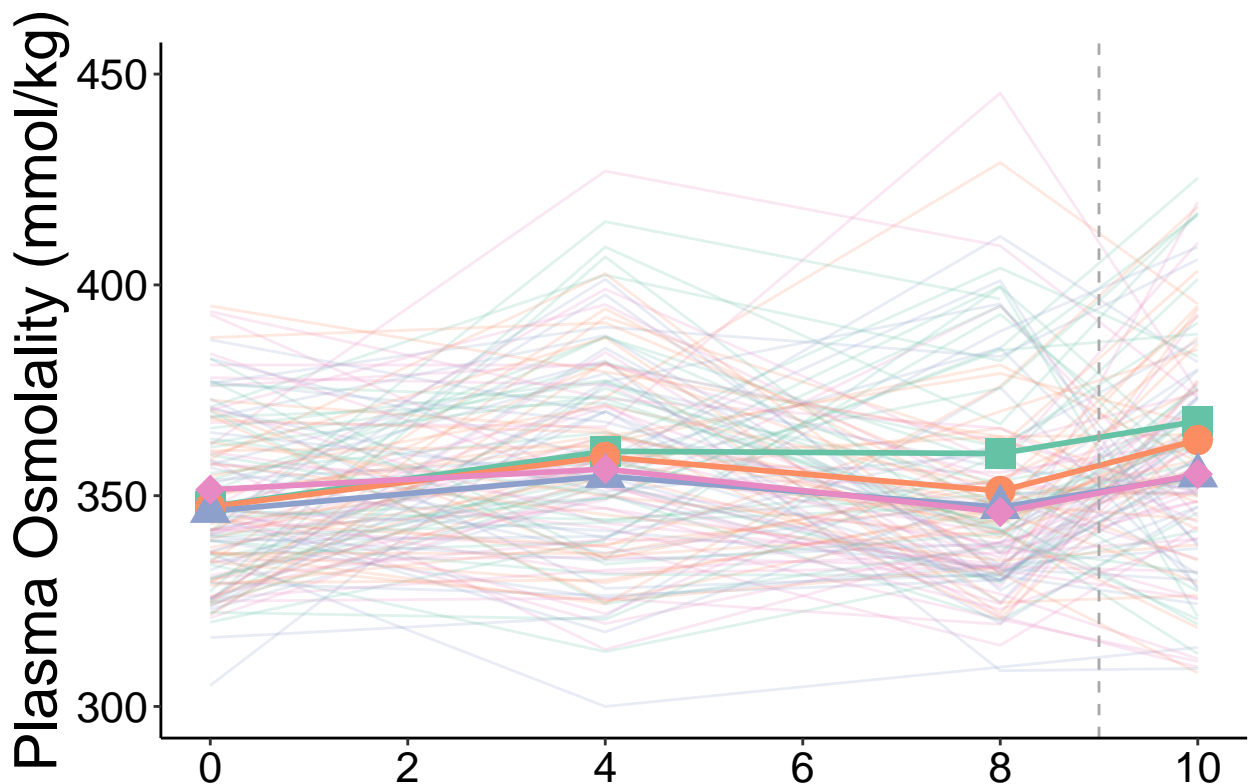
```

    color = "darkgrey") +
  scale_shape_manual(values = c(15:18), name = "") +
  scale_x_continuous(breaks = c(0, 2, 4, 6, 8, 10)) +
  ylim(300,450) +
  scale_color_brewer(palette = "Set2", name = "") +
  xlab("") +
  ylab("Plasma Osmolality (mmol/kg)") +
  guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
  theme_classic() +
  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 = 22),
        legend.text.align = 0,
        legend.position = "none",
        plot.margin = unit(c(0.6, #top
                             0.1, #right
                             0.1, #bottom
                             0.1 #left
                             ), "cm"))

) -> osml_fig
osml_fig

```

Warning: Removed 1 row(s) containing missing values (geom_path).



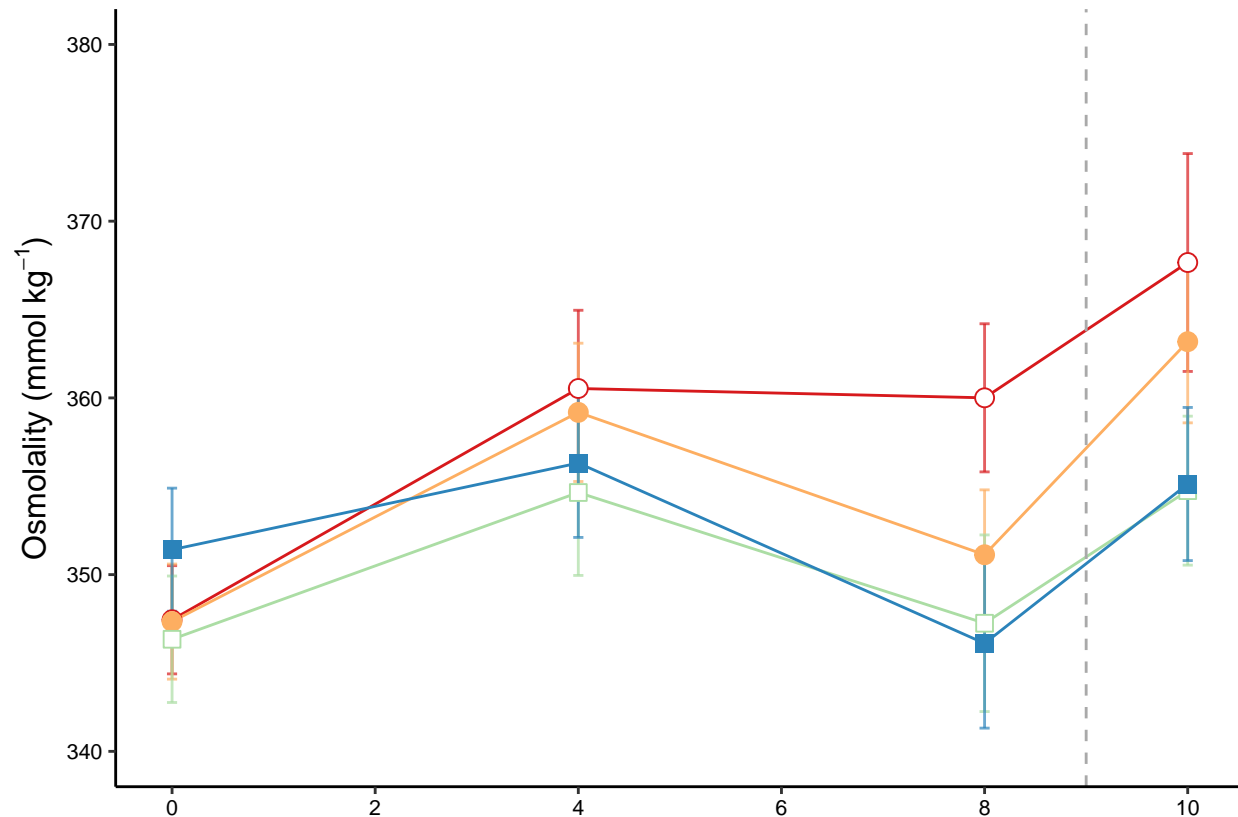
Means Only

```
ggplot() +
  geom_errorbar(data = means,
               aes(x = day_n,
                   y = mean_osml,
                   color = tmt,
                   group = tmt,
                   ymin = mean_osml-se_osml,
                   ymax = mean_osml+se_osml),
               width = .1,
               #position=position_dodge(.1),
               alpha = 0.7) +
  geom_line(data = means[complete.cases(means$mean_osml),],
            aes(x = day_n,
                y = mean_osml,
                color = tmt,
                #linetype = tmt,
                group = tmt),
            alpha = 1,
            size = 0.5) +
  geom_point(data = pt_means,
             aes(x = day_n,
                 y = mean_osml,
                 color = tmt,
                 #fill = tmt,
                 shape = tmt),
             fill = "white",
             alpha = 1,
             size = 3) +
  theme_classic() +
  scale_shape_manual(values = c(21,19, 22,15), name = "") +
  #scale_linetype_manual(values = c("solid", "longdash", "solid", "longdash"),
  #                        name = "") +
  scale_fill_brewer(palette = "Spectral", name = "") +
  scale_color_brewer(palette = "Spectral", name = "") +
  scale_x_continuous(breaks = c(0, 2, 4, 6, 8, 10)) +
  scale_y_continuous(breaks = c(seq(320,400, by = 10)),
                     limits = c(340,380)) +
  xlab("") +
  ylab(bquote('Osmolality (mmol '*kg-1*)')) +
  guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
  geom_vline(xintercept = 9,
             linetype = "dashed",
             color = "darkgrey") +
  theme(text = element_text(color = "black",
                             family = "sans",
                             size = 12),
        axis.text = element_text(color = "black",
                                   family = "sans",
                                   size = 8),
        legend.text = element_text(color = "black",
                                     family = "sans",
                                     size = 12),
```

```

legend.text.align = 0,
legend.position = "none",
plot.margin = margin(t = 6, r = 6, b = 0, l = 1, unit = "pt")
) -> osml_fig_min
osml_fig_min

```



Stats! Check Pairwise Diffs ~ Time

Since Plasma osmolality has such a nonlinear trend, we need to test whether the elevated values in the middle of the experiment are significantly different than the values taken before and/or after.

```

# first make sub-dfs for each tmt group
HH <- dat_no_rehab %>%
  dplyr::filter(tmt == "Hot Humid")
HD <- dat_no_rehab %>%
  dplyr::filter(tmt == "Hot Dry")
CH <- dat_no_rehab %>%
  dplyr::filter(tmt == "Cool Humid")
CD <- dat_no_rehab %>%
  dplyr::filter(tmt == "Cool Dry")

# next do pairwise tests for osml on the diff exp days, for each tmt group
pair_HH <- TukeyHSD(aov(data = HH, osmolality_mmol_kg_mean ~ day_factor)) #nonsig
pair_HD <- TukeyHSD(aov(data = HD, osmolality_mmol_kg_mean ~ day_factor)) #nonsig
pair_CH <- TukeyHSD(aov(data = CH, osmolality_mmol_kg_mean ~ day_factor)) #nonsig
pair_CD <- TukeyHSD(aov(data = CD, osmolality_mmol_kg_mean ~ day_factor)) #nonsig

# put into a df and export

```

```

osml_pairwise_df <- as.data.frame(pair_HD[[1]]) %>%
  rbind(as.data.frame(pair_HH[[1]])) %>%
  rbind(as.data.frame(pair_CD[[1]])) %>%
  rbind(as.data.frame(pair_CH[[1]])) %>%
  mutate(day_diff = paste("day", substr(rownames(.), 1, 3)),
         tmt = c(rep("Hot Dry",3),
                  rep("Hot Humid",3),
                  rep("Cool Dry",3),
                  rep("Cool Humid",3)),
         CI_95 = paste(round(lwr, digits = 2), round(upr, digits = 2), sep = ", "),
         diff = round(diff, digits = 2)) %>%
  dplyr::select(tmt, day_diff, diff, CI_95, p_adj = "p adj")
write.csv(osml_pairwise_df, "./results_statistics/osmolality_pairwise_diffs.csv")

```

nope, none of the differences between days within tmt groups are significantly different

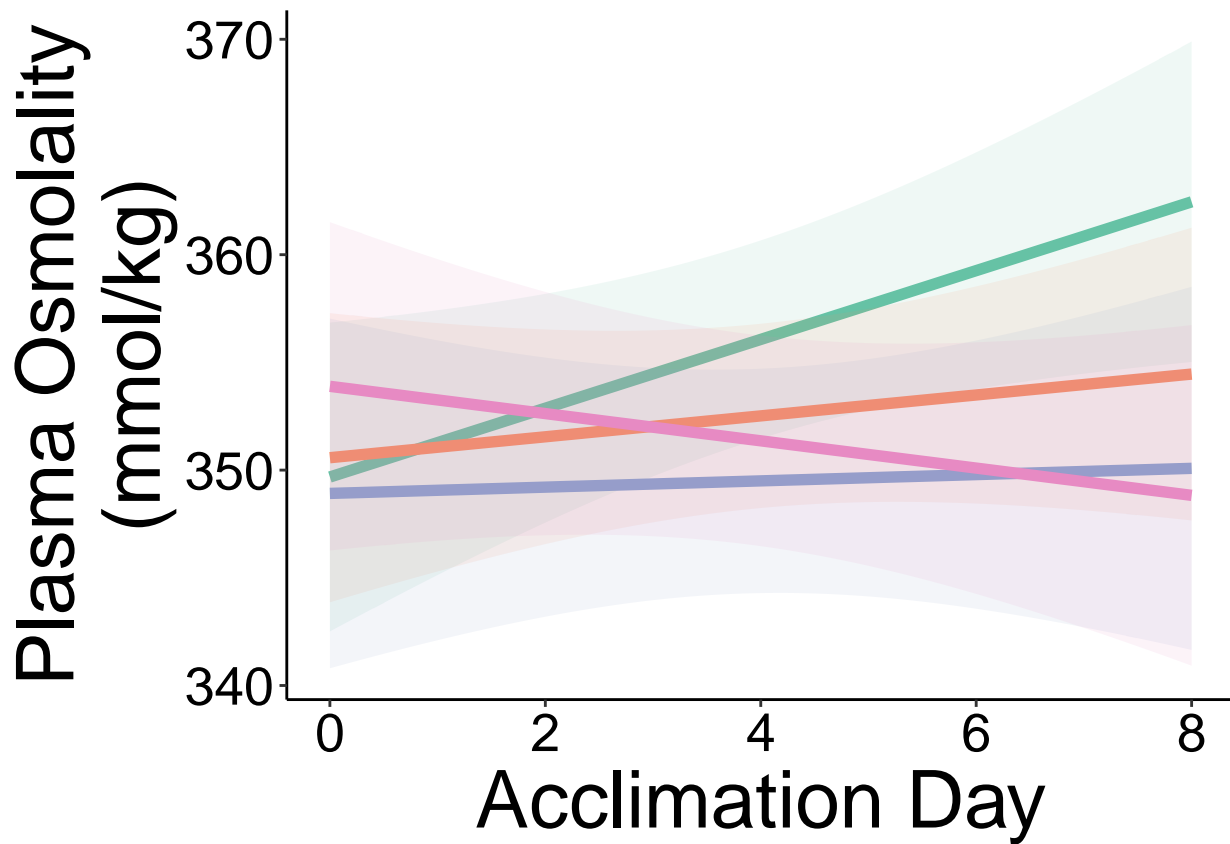
LM + SE

```

ggplot() +
  stat_smooth(data = dat_no_rehab,
             aes(x = day_n,
                 y = osmolality_mmol_kg_mean,
                 color = tmt,
                 fill = tmt,
                 group = tmt),
             formula = y ~ x,
             method = "lm",
             se = T,
             size = 2,
             alpha = 0.1) +
  theme_classic() +
  #scale_shape_manual(values = c(15:18), name = "") +
  scale_x_continuous(limits = c(0,8),
                    breaks = c(0, 2, 4, 6, 8)) +
  scale_color_brewer(palette = "Set2", name = "") +
  scale_fill_brewer(palette = "Set2", name = "") +
  xlab("Acclimation Day") +
  #ylim(340,370) + #CANNOT put ylims in bc gets rid of pts that are used to make lines
  ylab("Plasma Osmolality\n(mmol/kg)") +
  guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
  theme(text = element_text(color = "black",
                             family = "sans",
                             size = 30),
        axis.text = element_text(color = "black",
                                   family = "sans",
                                   size = 20),
        legend.text = element_text(color = "black",
                                     family = "sans",
                                     size = 22),
        legend.text.align = 0,
        legend.position = "none"
  ) -> osml_lm_fig
osml_lm_fig

```

```
## Warning: Removed 413 rows containing non-finite values (stat_smooth).
```



Ending Values Boxplot

```
end_vals_osml <- dat %>%
  dplyr::filter(day_n == 8) %>%
  dplyr::filter(complete.cases(osmolality_mmol_kg_mean))

osml_aov <- summary(aov(data = end_vals_osml,
  osmolality_mmol_kg_mean ~ temp_tmt * humidity_tmt))
osml_tukey <- TukeyHSD(aov(data = end_vals_osml,
  osmolality_mmol_kg_mean ~ temp_tmt * humidity_tmt))

ggplot(data = end_vals_osml) +
  geom_boxplot(aes(x = tmt,
    y = osmolality_mmol_kg_mean,
    color = tmt),
    size = 0.5,
    outlier.shape = NA) +
  geom_jitter(aes(x = tmt,
    y = osmolality_mmol_kg_mean,
    color = tmt,
    fill = tmt,
    shape = tmt),
    size = 1,
    alpha = 0.4,
    position = position_jitter(height = 0, width = 0.2)) +
```

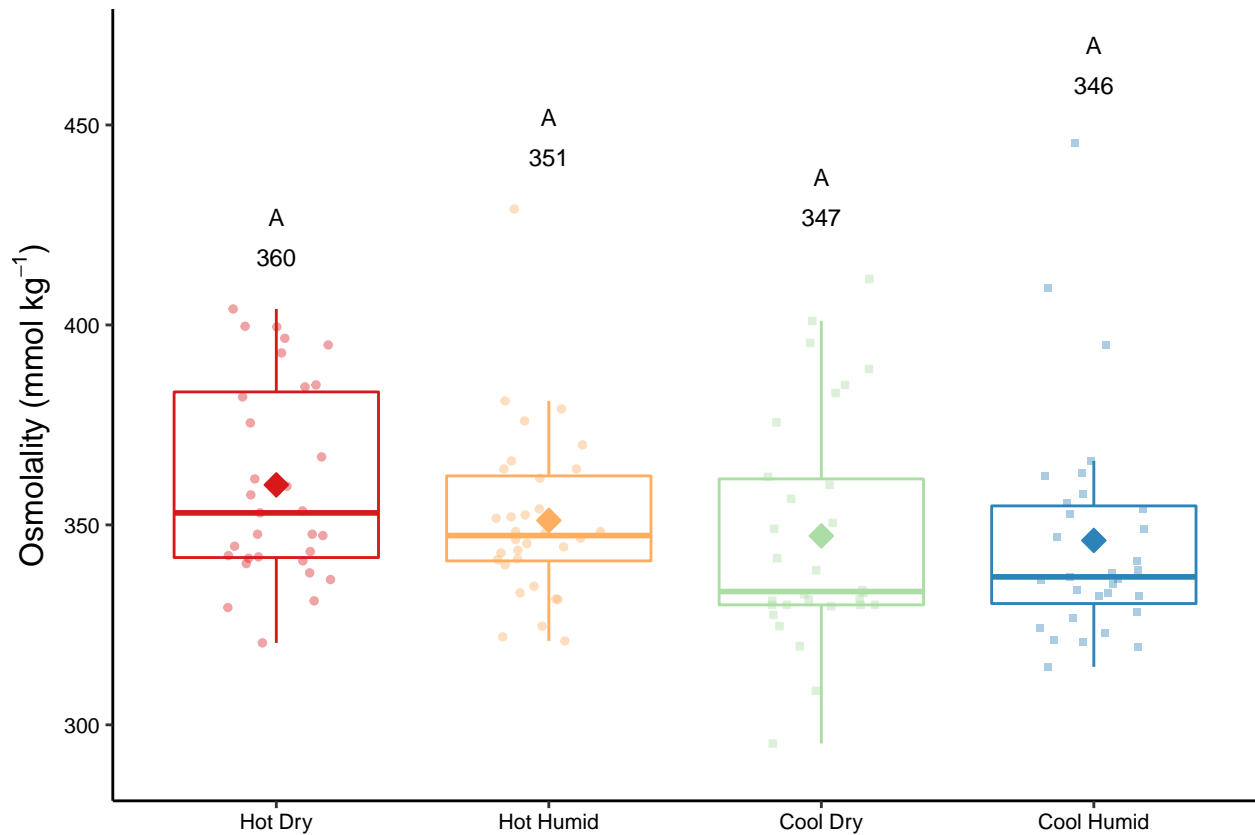
```

geom_point(data = end_means,
           aes(x = tmt,
               y = mean_osml,
               color = tmt,
               fill = tmt),
           size = 3,
           shape = 23) +
theme_classic() +
scale_shape_manual(values = c(21,19, 22,15), name = "") +
scale_fill_brewer(palette = "Spectral", name = "") +
scale_color_brewer(palette = "Spectral", name = "") +
scale_y_continuous(limits = c(290,470),
                   breaks = c(seq(300,450, by = 50)),
                   labels = c(seq(300,450, by = 50))) +
xlab("") +

annotate(geom = "text", x = 1, y = 427, label = "A",
         size = 3) +
annotate(geom = "text", x = 1, y = 417, label = "360",
         size = 3) +
annotate(geom = "text", x = 2, y = 452, label = "A",
         size = 3) +
annotate(geom = "text", x = 2, y = 442, label = "351",
         size = 3) +
annotate(geom = "text", x = 3, y = 437, label = "A",
         size = 3) +
annotate(geom = "text", x = 3, y = 427, label = "347",
         size = 3) +
annotate(geom = "text", x = 4, y = 470, label = "A",
         size = 3) +
annotate(geom = "text", x = 4, y = 460, label = "346",
         size = 3) +

ylab(bquote('Osmolality (mmol '*kg-1')))) +
theme(text = element_text(color = "black",
                          family = "sans",
                          size = 12),
      axis.text = element_text(color = "black",
                              family = "sans",
                              size = 8),
      #axis.text.x = element_blank(),
      legend.text = element_text(color = "black",
                                family = "sans",
                                size = 12),
      legend.text.align = 0,
      legend.position = "none",
      plot.margin = unit(c(0, #top
                          0, #right
                          0, #bottom
                          0), "mm"))
) -> osml_end_boxplot
osml_end_boxplot

```



CEWL

Ind + Means

```
ggplot() +
  geom_line(data = dat[complete.cases(dat$CEWL_g_m2h_mean),],
    aes(x = day_n,
        y = CEWL_g_m2h_mean,
        color = tmt,
        group = individual_ID),
    alpha = 0.2) +
  geom_line(data = means[complete.cases(means$mean_CEWL),],
    aes(x = day_n,
        y = mean_CEWL,
        color = tmt,
        group = tmt),
    alpha = 1,
    size = 1) +
  geom_point(data = pt_means[complete.cases(pt_means$mean_CEWL),],
    aes(x = day_n,
        y = mean_CEWL,
        color = tmt,
        shape = tmt),
    alpha = 1,
    size = 5) +
  theme_classic() +
  scale_shape_manual(values = c(15:18), name = "") +
```

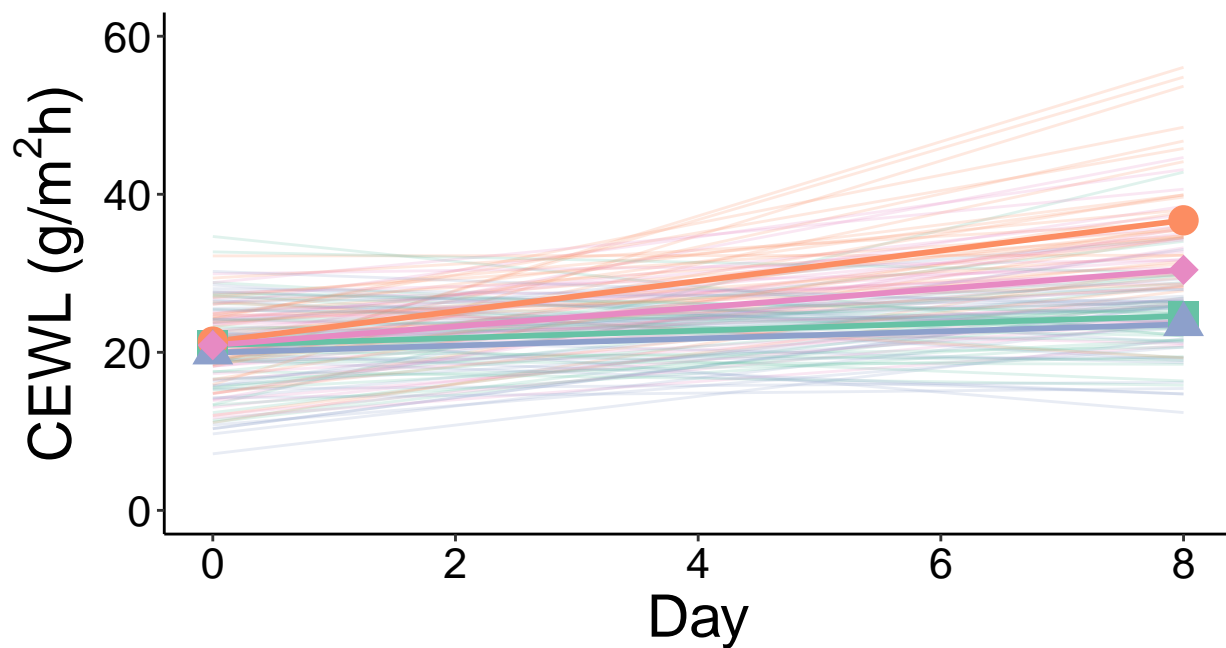


```

scale_x_continuous(breaks = c(0, 2, 4, 6, 8)) +
scale_color_brewer(palette = "Set2", name = "") +
xlab("Day") +
ylim(0,60) +
ylab(bquote('CEWL (g/m^2*h)')) +
guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
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 = 22),
      legend.text.align = 0,
      legend.position = "bottom"
      #legend.position = c(0.25,0.85)
      ) -> CEWL_fig

```

CEWL_fig



■ Hot Dry ● Hot Humid
▲ Cool Dry ◆ Cool Humid

```

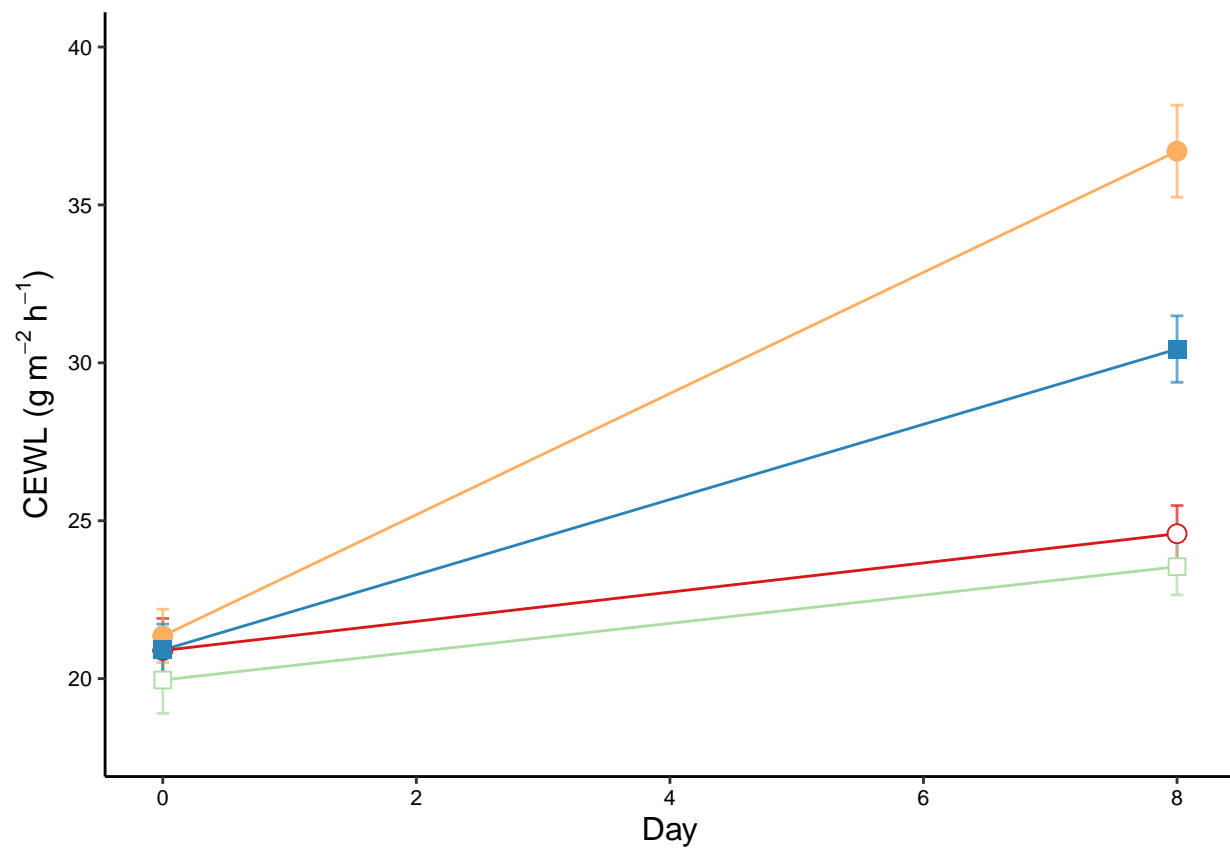
#ggsave(filename = "CEWL_fig1.jpeg",
#        plot = CEWL_fig,
#        path = "./results_figures",
#        device = "jpeg",
#        dpi = 1200,
#        width = 6, height = 6)

```

Means Only

```
ggplot() +
  geom_errorbar(data = means[complete.cases(means$mean_CEWL),],
    aes(x = day_n,
        y = mean_CEWL,
        color = tmt,
        group = tmt,
        ymin = mean_CEWL-se_CEWL,
        ymax = mean_CEWL+se_CEWL),
    width = .1,
    #position=position_dodge(.1),
    alpha = 0.7) +
  geom_line(data = means[complete.cases(means$mean_CEWL),],
    aes(x = day_n,
        y = mean_CEWL,
        color = tmt,
        #linetype = tmt,
        group = tmt),
    alpha = 1,
    size = .5) +
  geom_point(data = pt_means[complete.cases(pt_means$mean_CEWL),],
    aes(x = day_n,
        y = mean_CEWL,
        color = tmt,
        shape = tmt),
    fill = "white",
    alpha = 1,
    size = 3) +
  theme_classic() +
  scale_shape_manual(values = c(21,19, 22,15), name = "") +
  #scale_linetype_manual(values = c("solid", "longdash", "solid", "longdash"),
  #                        name = "") +
  scale_fill_brewer(palette = "Spectral", name = "") +
  scale_color_brewer(palette = "Spectral", name = "") +
  scale_x_continuous(breaks = c(0, 2, 4, 6, 8)) +
  scale_y_continuous(breaks = c(20, 25, 30, 35, 40),
    limits = c(18,40)) +
  xlab("Day") +
  ylab(bquote('CEWL (g *m^-2* ' *h^-1*)')) +
  guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
  theme(text = element_text(color = "black",
    family = "sans",
    size = 12),
    axis.text = element_text(color = "black",
    family = "sans",
    size = 8),
    legend.text = element_text(color = "black",
    family = "sans",
    size = 12),
    legend.text.align = 0,
    legend.position = "none"
    #legend.position = c(0.25,0.85)
  ) -> CEWL_fig_min
```

CEWL_fig_min



```
# use ggarrange so legend is centered
CEWL_fig_formatted <- ggarrange(CEWL_fig_min,
                                ncol = 1, nrow = 1,
                                common.legend = TRUE,
                                legend = "bottom")

# save
ggsave(filename = "experiment_CEWL_fig.pdf",
        plot = CEWL_fig_formatted,
        path = "./results_figures",
        device = "pdf",
        dpi = 600,
        units = "mm",
        width = 80, height = 80)
```

LM + SE

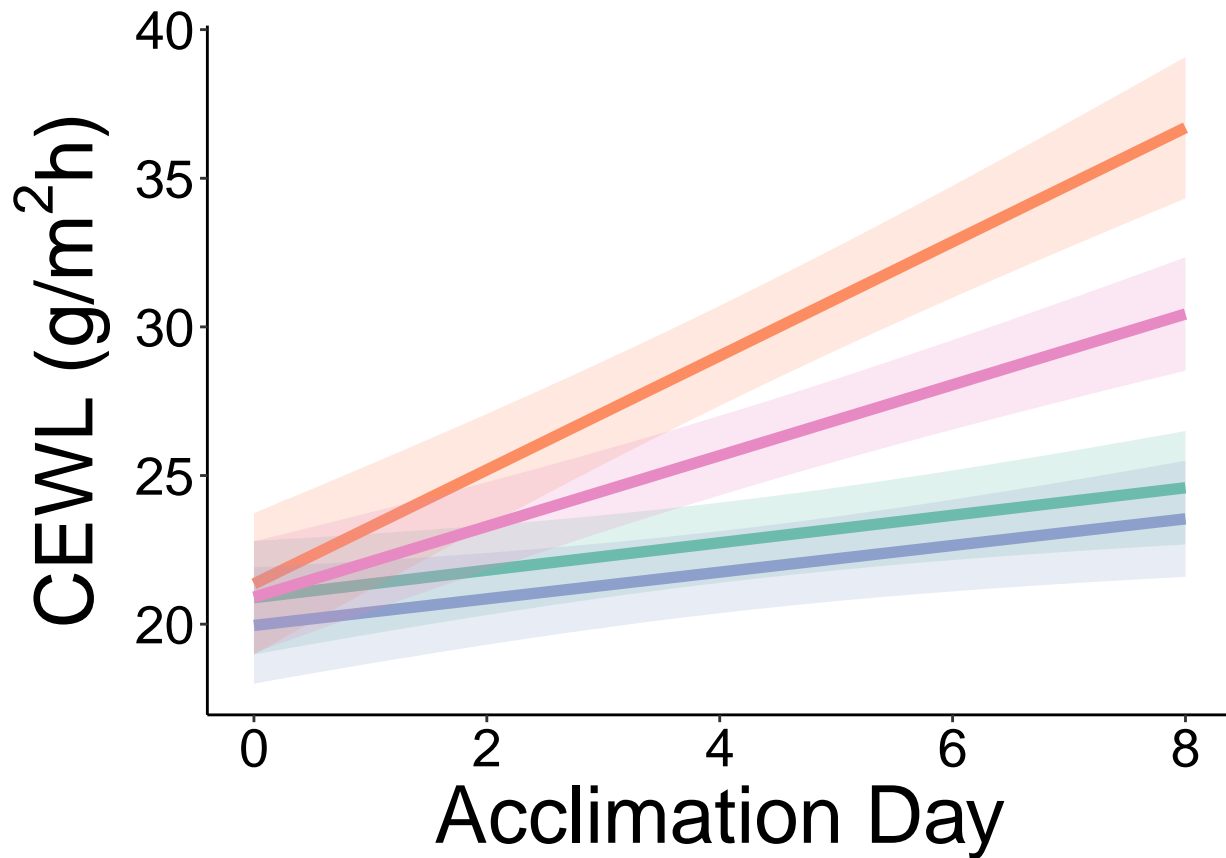
```
ggplot() +
  stat_smooth(data = dat,
             aes(x = day_n,
                 y = CEWL_g_m2h_mean,
                 color = tmt,
                 fill = tmt,
                 group = tmt),
             formula = y ~ x,
```

```

        method = "lm",
        se = T,
        size = 2,
        alpha = 0.2) +
theme_classic() +
scale_x_continuous(breaks = c(0, 2, 4, 6, 8)) +
scale_color_brewer(palette = "Set2", name = "") +
scale_fill_brewer(palette = "Set2", name = "") +
xlab("Acclimation Day") +
ylab(bquote('CEWL (g/*m2*h)')) +
guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
theme(text = element_text(color = "black",
                           family = "sans",
                           size = 30),
      axis.text = element_text(color = "black",
                                family = "sans",
                                size = 20),
      legend.text = element_text(color = "black",
                                  family = "sans",
                                  size = 22),
      legend.text.align = 0,
      legend.position = "none"
      #legend.position = c(0.25,0.85)
      ) -> CEWL_lm_fig
CEWL_lm_fig

```

```
## Warning: Removed 669 rows containing non-finite values (stat_smooth).
```



Ending Values Boxplot

```
end_vals_CEWL <- dat %>%
  dplyr::filter(day_n == 8) %>%
  dplyr::filter(complete.cases(CEWL_g_m2h_mean))

CEWL_tukey <- TukeyHSD(aov(data = end_vals_CEWL,
  CEWL_g_m2h_mean ~ temp_tmt * humidity_tmt))

ggplot(data = end_vals_CEWL) +
  geom_boxplot(aes(x = tmt,
    y = CEWL_g_m2h_mean,
    color = tmt),
    size = 0.5,
    outlier.shape = NA) +
  geom_jitter(aes(x = tmt,
    y = CEWL_g_m2h_mean,
    color = tmt,
    fill = tmt,
    shape = tmt),
    size = 1,
    alpha = 0.4,
    position = position_jitter(height = 0, width = 0.2)) +
  geom_point(data = end_means,
    aes(x = tmt,
```

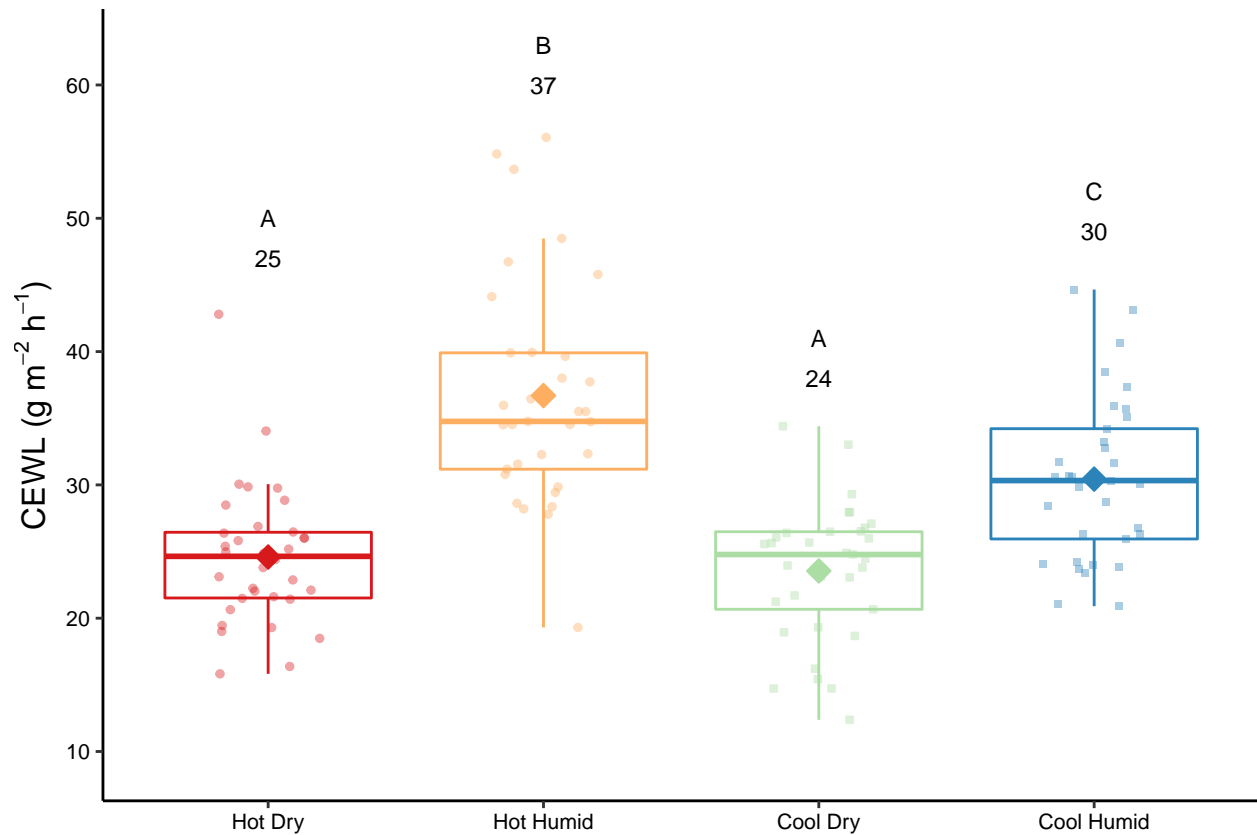
```

        y = mean_CEWL,
        color = tmt,
        fill = tmt),
        size = 3,
        shape = 23) +
theme_classic() +
scale_shape_manual(values = c(21,19, 22,15), name = "") +
scale_fill_brewer(palette = "Spectral", name = "") +
scale_color_brewer(palette = "Spectral", name = "") +
scale_y_continuous(limits = c(9,63),
                    breaks = c(seq(10,60, by = 10)),
                    labels = c(seq(10,60, by = 10))) +
xlab("") +
ylab(bquote('CEWL (g '*m^-2*' '*h^-1*')')) +

annotate(geom = "text", x = 1, y = 50, label = "A",
         size = 3) +
annotate(geom = "text", x = 1, y = 47, label = "25",
         size = 3) +
annotate(geom = "text", x = 2, y = 63, label = "B",
         size = 3) +
annotate(geom = "text", x = 2, y = 60, label = "37",
         size = 3) +
annotate(geom = "text", x = 3, y = 41, label = "A",
         size = 3) +
annotate(geom = "text", x = 3, y = 38, label = "24",
         size = 3) +
annotate(geom = "text", x = 4, y = 52, label = "C",
         size = 3) +
annotate(geom = "text", x = 4, y = 49, label = "30",
         size = 3) +

theme(text = element_text(color = "black",
                           family = "sans",
                           size = 12),
      axis.text = element_text(color = "black",
                                family = "sans",
                                size = 8),
      #axis.text.x = element_blank(),
      legend.text = element_text(color = "black",
                                  family = "sans",
                                  size = 12),
      legend.text.align = 0,
      legend.position = "none",
      plot.margin = unit(c(0, #top
                          0, #right
                          0, #bottom
                          0), "mm"))
) -> CEWL_end_boxplot
CEWL_end_boxplot

```



CEWL ~ Osmol

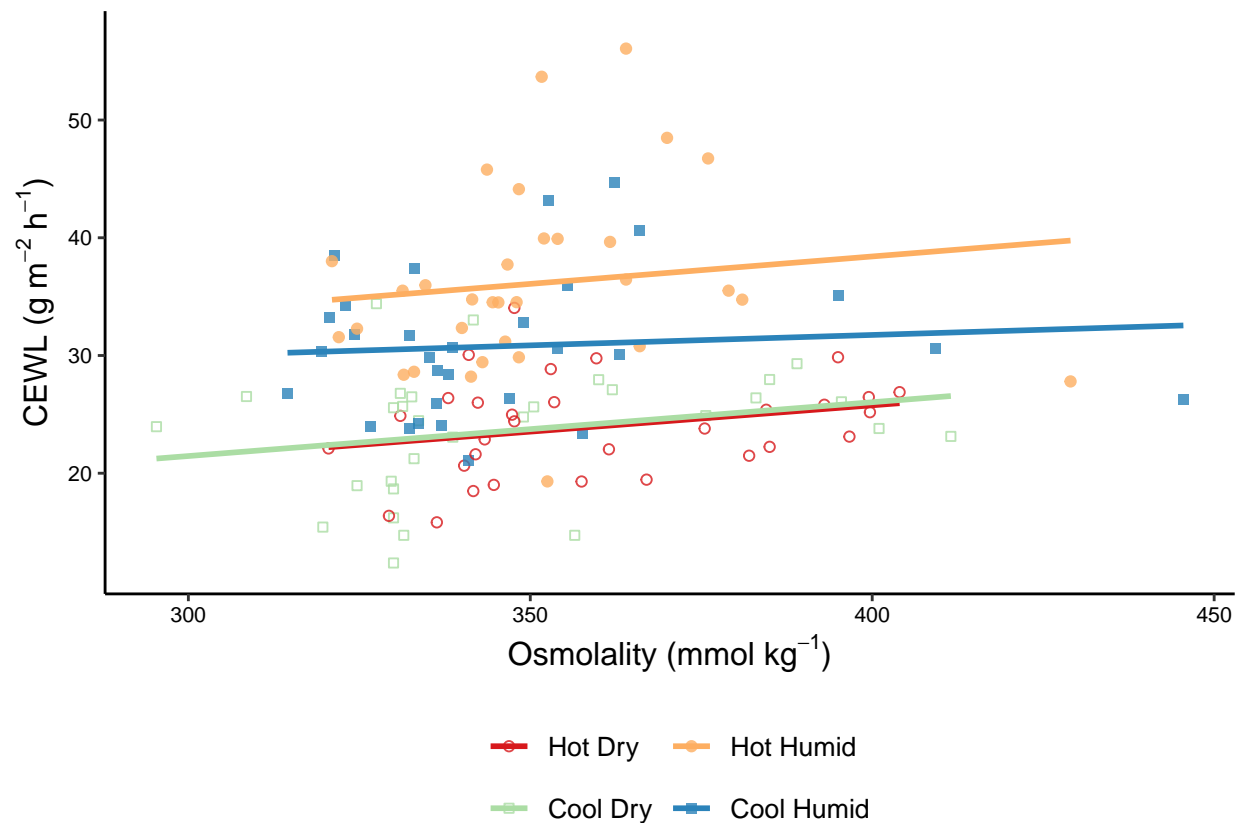
```
end_vals_CEWL_osml <- dat %>%
  dplyr::filter(day_n == 8) %>%
  dplyr::filter(complete.cases(CEWL_g_m2h_mean, osmolality_mmol_kg_mean))

ggplot(end_vals_CEWL_osml) +
  aes(x = osmolality_mmol_kg_mean,
       y = CEWL_g_m2h_mean,
       color = tmt,
       shape = tmt) +
  geom_point(size = 1.5,
             alpha = 0.8) +
  stat_smooth(formula = y ~ x,
             method = "lm",
             se = F,
             size = 1,
             alpha = 0.9) +
  theme_classic() +
  xlab(bquote('Osmolality (mmol *kg-1)')) +
  ylab(bquote('CEWL (g *m-2 *h-1)')) +
  #xlim(300, 400) +
  #ylim(0, 40) +
  scale_shape_manual(values = c(21,19, 22,15), name = "") +
  scale_fill_brewer(palette = "Spectral", name = "") +
  scale_color_brewer(palette = "Spectral", name = "") +
```

```

scale_x_continuous(breaks = c(300, 350, 400, 450)) +
scale_y_continuous(breaks = c(20, 30, 40, 50),
                  limits = c(12, 57)) +
guides(shape = guide_legend(nrow = 2, byrow = TRUE)) +
theme(text = element_text(color = "black",
                          family = "sans",
                          size = 12),
      axis.text = element_text(color = "black",
                              family = "sans",
                              size = 8),
      legend.position = "bottom"
      #axis.text.y = element_blank(),
      #plot.margin = unit(c(0.1, 0, 0.1, 0.45), "cm")
) -> exp_end_CEWL_osml_fig
exp_end_CEWL_osml_fig

```



```

# will need to save using figure arrange to make legend centered if I use that fig
ggsave(filename = "exp_CEWL_osml_fig.pdf",
        plot = exp_end_CEWL_osml_fig,
        path = "./results_figures",
        device = "pdf",
        dpi = 600,
        units = "mm",
        width = 80, height = 90)

```


Multi-Figures

over time:

```
ggarrange(osml_fig_min,
          hct_fig_min,
          SMI_fig_min,
          ncol = 1, nrow = 3,
          labels = c("A", "B", "C"),
          common.legend = TRUE,
          legend = "bottom"
          ) -> experiment_multi_fig
#experiment_multi_fig
# export figure
ggsave(filename = "experiment_multi_fig.pdf",
        plot = experiment_multi_fig,
        path = "./results_figures",
        device = "pdf",
        dpi = 600,
        units = "mm",
        width = 80, height = 210)
```

end values:

```
ggarrange(CEWL_end_boxplot,
          osml_end_boxplot,
          hct_end_boxplot,
          SMI_end_boxplot,
          ncol = 2, nrow = 2,
          labels = c("A", "B", "C", "D"),
          widths = c(2, 2.045), heights = c(2, 2),
          common.legend = FALSE
          ) -> ending_values_multi_fig
#ending_values_multi_fig

ggsave(filename = "exp_end_val_multi_fig.pdf",
        plot = ending_values_multi_fig,
        path = "./results_figures",
        device = "pdf",
        dpi = 600,
        units = "mm",
        width = 180, height = 150)
```