resp_calc_analysis_RMD

Samuel Gurr

9/22/2021

CALCULATE RESPIRATION RATES

(1) summary table of blanks to normalize respiration rates

- we will use 'Lpc' value (or L%) was chosen to generally choose the local regression with the majority of observations in the truly linear subset of data
- review Olito et al. 2017 Estimating monotonic rates from biological data using local linear regression

```
##        Date                     Notes BLANK.mean.Lpc BLANK.mean.Leq
## 1 4/30/2021     20210319_new_sensor_7   0.0007473465   0.0006079785
## 2 4/30/2021  20210430_LOWtemp_HIGHsal   0.0189874195   0.0197653280
## 3 4/30/2021   20210430_LOWtemp_LOWsal   0.0013211745   0.0012705960
## 4 4/30/2021              20210430_raw   0.0005703090   0.0005747545
## 5  5/7/2021 20210507_HIGHtemp_HIGHsal   0.0011802100   0.0015554425
## 6  5/7/2021  20210507_HIGHtemp_LOWsal   0.0000902505   0.0001397385
##   BLANK.mean.Lz
## 1  0.0006814560
## 2  0.0197631395
## 3  0.0012566205
## 4  0.0005747545
## 5  0.0013648970
## 6  0.0003289235
```

(2) Calculate respiration rates as ng O2 L-1 indiv-1 hr-1

- normalize by mean value of blanks for each treatment, grouped by 'Date' and 'Notes' (as the run wihtin day)
- omit rows resprenting blank resp > sample resp - likely bad data, view to troubleshoot if needed

```
## [1] "number of blank resp > sample resp ="
## [2] "0"
```

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

```
## # A tibble: 15 x 3
## # Groups:   Date [2]
##    Date      TempCarbSal     n
##    <chr>     <chr>       <int>
##  1 4/30/2021 HHH             3
##  2 4/30/2021 HHL             3
```

```
##  3 4/30/2021 HLH             3
##  4 4/30/2021 HLL             3
##  5 4/30/2021 LHH             3
##  6 4/30/2021 LHL             3
##  7 4/30/2021 LLH             3
##  8 4/30/2021 LLL             3
##  9 5/7/2021  HHH             3
## 10 5/7/2021  HHL             1
## 11 5/7/2021  HLH             3
## 12 5/7/2021  LHH             6
## 13 5/7/2021  LHL             3
## 14 5/7/2021  LLH             4
## 15 5/7/2021  LLL             3
```

- calculate resp rates

```
##        Date Chamber_tank Channel resp_ng_L_indiv_hr
## 1 4/30/2021            1     CH1         0.01037379
## 2 4/30/2021           10     CH5         0.05641173
## 3 4/30/2021           11     CH6         0.08542498
## 4 4/30/2021           12     CH7         0.15307783
## 5 4/30/2021           13     CH1         0.04543331
## 6 4/30/2021           14     CH2         0.03518799
```

ANALYSIS

Day 1 Respirometry (20210430)

- a. filter the dataset for taget data
- b. add column for mean aragonite sat. (note: this is from the entire experiment, not just day 1 chemistry!)
- c. run moels and diagnostics
- d. plot (reconfigure data as needed for plotting; i.e. melt/dcast, etc.)

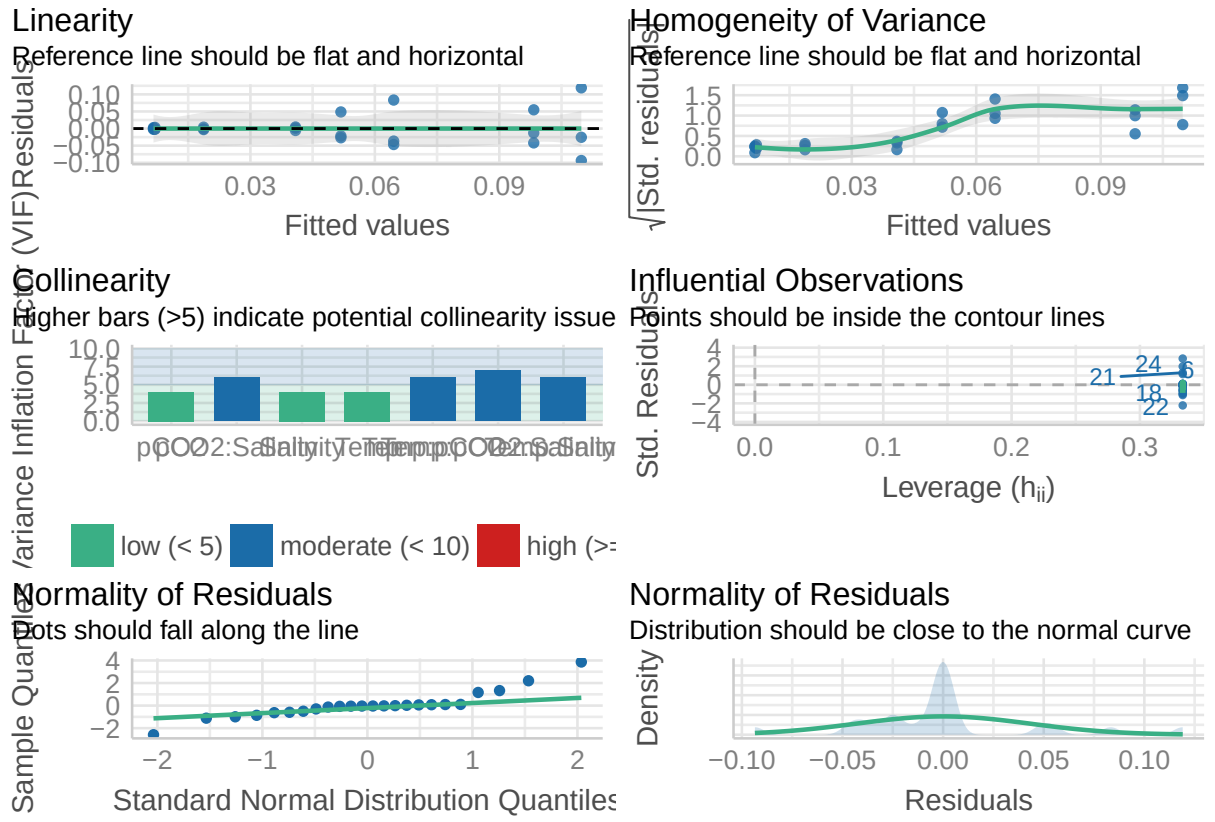
```
# a.
Resp_APRIL <- Resp.Master %>%
  dplyr::filter(Date %in% '4/30/2021')

# b.
Resp_APRIL <- merge(Resp_APRIL, aragonite_dat, by = 'TempCarbSal') # merge with the mean aragonite satu

# c.
LMmod.APRIL   <- aov(lm( resp_ng_L_indiv_hr~Temp*pCO2*Salinity,data=Resp_APRIL))
print('LM model  == aov( lm ( resp_ng_L_indiv_hr ~ Temp * pCO2 * Salinity, data = Resp_APRIL ) )')
```

```
## [1] "LM model  == aov( lm ( resp_ng_L_indiv_hr ~ Temp * pCO2 * Salinity, data = Resp_APRIL ) )"
```

```
summary(LMmod.APRIL)
```

```
##                     Df  Sum Sq  Mean Sq F value Pr(>F)
## Temp                 1 0.01558 0.015583   5.849 0.0279 *
## pCO2                 1 0.00079 0.000786   0.295 0.5946
## Salinity             1 0.00173 0.001727   0.648 0.4326
## Temp:pCO2            1 0.01284 0.012835   4.817 0.0433 *
## Temp:Salinity        1 0.00020 0.000199   0.075 0.7880
## pCO2:Salinity        1 0.00078 0.000781   0.293 0.5956
## Temp:pCO2:Salinity   1 0.00082 0.000818   0.307 0.5872
## Residuals           16 0.04263 0.002664
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
check_model(LMmod.APRIL) # observe the diagnostics of the model
```




```
shapiro.test(residuals(LMmod.APRIL)) # non normal
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(LMmod.APRIL)
## W = 0.88542, p-value = 0.0107
```

```
leveneTest(LMmod.APRIL) # good
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  7  1.2514 0.3331
##       16
```

```
# LME model here however we do only have one rep per chamber_tank in some cases, thus the lm (above is
# MEmod.APRIL    <- lmer(resp_ng_L_indiv_hr~Temp*pCO2*Salinity + (1|Chamber_tank),REML=F, data=Resp_APR
# summary(MEmod.APRIL)
# check_model(MEmod.APRIL)
# shapiro.test(residuals(MEmod.APRIL)) # non normal
# leveneTest(MEmod.APRIL) # good

# POst-hoc tests and exploration of sig effects
TukeyHSD(LMmod.APRIL, 'Temp:pCO2')
```

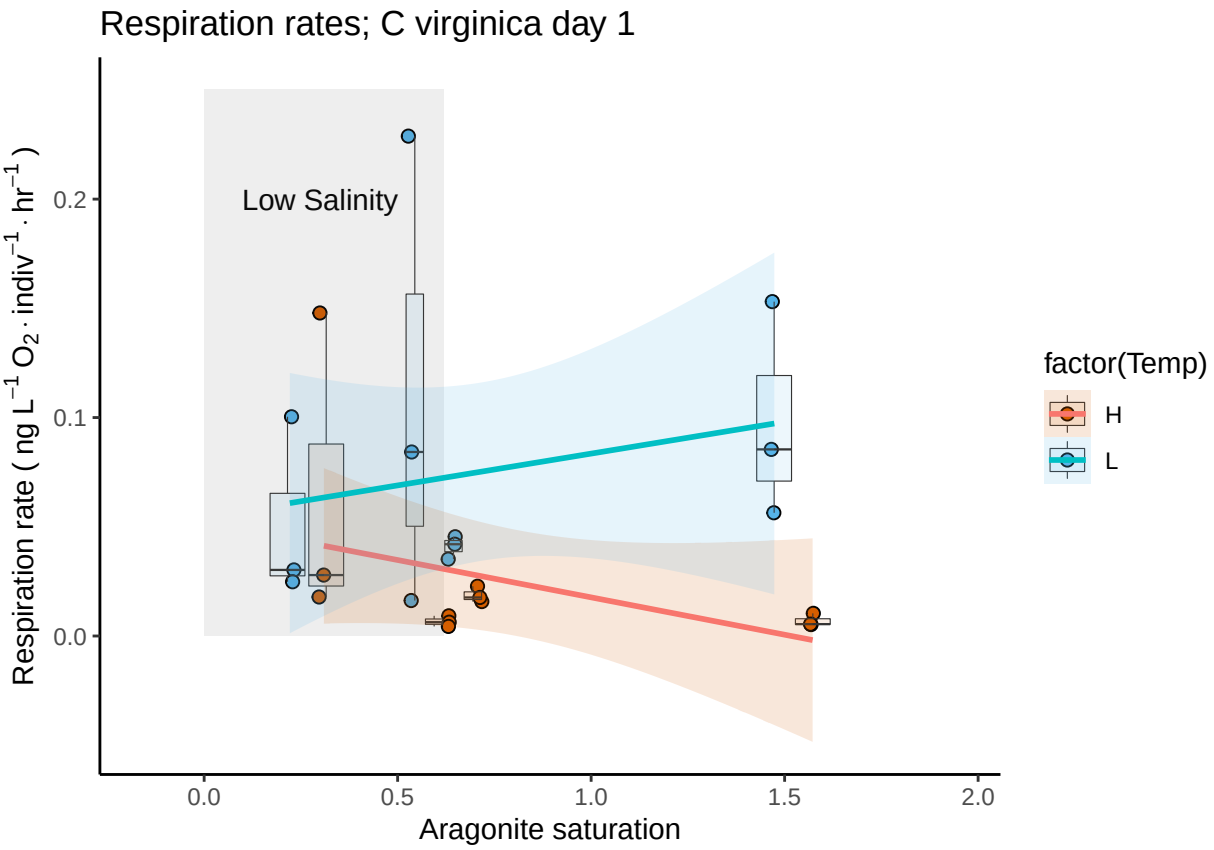
```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = lm(resp_ng_L_indiv_hr ~ Temp * pCO2 * Salinity, data = Resp_APRIL))
##
## $`Temp:pCO2`
##                 diff         lwr        upr     p adj
## L:H-H:H  0.004711293 -0.08055106 0.08997364 0.9985320
## H:L-H:H -0.034809121 -0.12007147 0.05045323 0.6545779
## L:L-H:H  0.062404535 -0.02285782 0.14766689 0.1970100
## H:L-L:H -0.039520413 -0.12478276 0.04574194 0.5605082
## L:L-L:H  0.057693242 -0.02756911 0.14295559 0.2526911
## L:L-H:L  0.097213656  0.01195130 0.18247601 0.0227961
```

```
# d.
Resp_APRIL_select  <- Resp_APRIL %>% dplyr::select(c('resp_ng_L_indiv_hr', 'Temp', 'pCO2', 'Salinity'))
Resp_APRIL_melt    <- tidyr::gather(Resp_APRIL_select, variable, value, -resp_ng_L_indiv_hr)

# MAIN EFFECT PLOT
# ggplot(Resp_APRIL_melt, aes(value , resp_ng_L_indiv_hr , fill = factor(value ))) +
#   theme(panel.grid=element_blank()) +
#   scale_color_manual(values=c("#56B4E9","#D55E00")) +
#   geom_point(shape = 21, size = 2, position = position_jitterdodge(jitter.width = 0.5))+
#   geom_boxplot(size=0.2, alpha=0.1) +
#   theme_bw() +
#   facet_wrap(~variable, scales = "free_y")

# INTERACTION PLOT
Resp_APRIL %>%
  dplyr::mutate(full.treatment = (paste(Temp, pCO2, Salinity,sep=''))) %>%
  dplyr::mutate(full.treatment = fct_relevel(full.treatment,
                             "HHH", "HLH", "LHH",'LLH',
                             "HHL", "HLL", "LHL", 'LLL')) %>%
  ggplot(aes(meanAragonite, resp_ng_L_indiv_hr , group =full.treatment, fill = factor(Temp))) +
  geom_point(shape = 21, size = 2, position = position_jitterdodge(jitter.width = 0.05))+
  geom_boxplot(size=.2, alpha=0.1, width =0.2) +
  scale_fill_manual(values=c("#D55E00","#56B4E9", "#D55E00","#56B4E9",
                             "#D55E00","#56B4E9", "#D55E00","#56B4E9")) +
```

```
  labs(title = "Respiration rates; C virginica day 1",
       y = expression(Respiration~rate~"("~ng~L^{-1}~O[2]%.%indiv^{-1}%.% hr^{-1}~")"),
       x = "Aragonite saturation") +
  annotate("text", x=0.3, y=0.2, label = "Low Salinity") +
  annotate("rect", xmin = 0, xmax = 0.62, ymin = 0, ymax = 0.25, alpha = .1) +
  geom_smooth(method = "lm", level = 0.95, alpha = .15, aes(group=Temp,colour = factor(Temp))) +
  theme_classic()
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: position_jitterdodge requires non-overlapping x intervals
```



Day 8 Respirometry (20210507)

```
# a.
Resp_MAY <- Resp.Master %>%
  dplyr::filter(Date %in% '5/7/2021')
# b.
Resp_MAY <- merge(Resp_MAY, aragonite_dat, by = 'TempCarbSal') # merge with the mean aragonite saturati
# c.
LMmod.MAY   <- aov(lm(resp_ng_L_indiv_hr~Temp*pCO2*Salinity,data=Resp_MAY))
print('LM model  == aov( lm ( resp_ng_L_indiv_hr ~ Temp * pCO2 * Salinity, data = Resp_APRIL ) )')
```

```
## [1] "LM model  == aov( lm ( resp_ng_L_indiv_hr ~ Temp * pCO2 * Salinity, data = Resp_APRIL ) )"
```

```
summary(LMmod.MAY)
```

```
##               Df  Sum Sq  Mean Sq F value Pr(>F)
## Temp           1 0.00296 0.002962   0.314  0.583
## pCO2           1 0.00478 0.004776   0.507  0.487
## Salinity       1 0.02777 0.027769   2.947  0.105
## Temp:pCO2      1 0.02652 0.026520   2.815  0.113
## Temp:Salinity  1 0.00008 0.000080   0.008  0.928
## pCO2:Salinity  1 0.01576 0.015758   1.673  0.214
## Residuals     16 0.15075 0.009422
```

```
# check_model(LMmod.MAY) # observe the diagnostics of the model
shapiro.test(residuals(LMmod.MAY)) # non normal
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(LMmod.MAY)
## W = 0.82009, p-value = 0.0008179
```

```
leveneTest(LMmod.MAY) # good
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  6  1.3835 0.2802
##       16
```

```
# LME model here however we do only have one rep per chamber_tank in some cases, thus the lm (above is
# MEmod.APRIL    <- lmer(resp_ng_L_indiv_hr~Temp*pCO2*Salinity + (1|Chamber_tank),REML=F, data=Resp_APR
# summary(MEmod.APRIL)
# check_model(MEmod.APRIL)
# shapiro.test(residuals(MEmod.APRIL)) # non normal
# leveneTest(MEmod.APRIL) # good


# d.
Resp_MAY_select  <- Resp_MAY %>% dplyr::select(c('resp_ng_L_indiv_hr', 'Temp', 'pCO2', 'Salinity'))
Resp_MAY_melt    <- tidyr::gather(Resp_MAY_select, variable, value, -resp_ng_L_indiv_hr)

# MAIN EFFECT PLOT
# ggplot(Resp_APRIL_melt, aes(value , resp_ng_L_indiv_hr , fill = factor(value ))) +
#   theme(panel.grid=element_blank()) +
#   scale_color_manual(values=c("#56B4E9","#D55E00")) +
#   geom_point(shape = 21, size = 2, position = position_jitterdodge(jitter.width = 0.5))+
#   geom_boxplot(size=0.2, alpha=0.1) +
#   theme_bw() +
#   facet_wrap(~variable, scales = "free_y")

# INTERACTION PLOT
```

```
Resp_MAY %>%
  dplyr::mutate(full.treatment = (paste(Temp, pCO2, Salinity,sep=''))) %>%
  dplyr::mutate(full.treatment = fct_relevel(full.treatment,
                              "HHH", "HLH", "LHH",'LLH',
                              "HHL", "LHL", 'LLL')) %>%
  ggplot(aes(meanAragonite, resp_ng_L_indiv_hr , group =full.treatment, fill = factor(Temp))) +
  geom_point(shape = 21, size = 2, position = position_jitterdodge(jitter.width = 0.05))+
  geom_boxplot(size=.2, alpha=0.1, width =0.2) +
  scale_fill_manual(values=c("#D55E00","#56B4E9", "#D55E00","#56B4E9",
                             "#D55E00","#56B4E9", "#D55E00","#56B4E9")) +
  labs(title = "Respiration rates; C virginica day 8",
       y = expression(Respiration~rate~"("~ng~L^{-1}~O[2]%.%indiv^{-1}%.% hr^{-1}~")"),
       x = "Aragonite saturation") +
  annotate("text", x=0.3, y=0.55, label = "Low Salinity") +
  annotate("rect", xmin = 0, xmax = 0.62, ymin = 0, ymax = 0.6, alpha = .1) +
  geom_smooth(method = "lm", level = 0.95, alpha = .15, aes(group=Temp,colour = factor(Temp))) +
  theme_classic()
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: position_jitterdodge requires non-overlapping x intervals
```