Tall stature and small leaves: ecological strategies that enhance tree growth across the subtropical Brazilian Atlantic Forest
README file to conduct the analyses presented in the manuscript

Kauane Maiara Bordin

2025-02-22

This file describes the content of the folder to conduct the analyses of the above-titled manuscript.

The folder contains three folders: `data`, `results`, and `scripts`. In the data folder you will find the `species.information.csv` file, `metadata` from the `species.information.csv` file, and `plot` metadata. In the scripts folder there is a `code.R` file, and the `results` folder contains the main PCAs presented in the manuscript.

In the `code.R` script file, you will find all the functions and procedures to conduct the data analysis; in the `species.information.csv` file you will find the processed data for species used in the analysis.

The columns of species.information.csv file are the following:

[1] Species = Species scientific name

[2] Family = Species' family

[3] Relative.Growth.Rate: Relative Growth Rate (RGR), obtained from the following equation: (dbh1/dbh0)/time, where dbh1 means dbh in mm in the second census, dbh0 is the dbh in mm in the first census, and time is the time census interval in years

[4] Mean.absolute.growth.rates: Absolute Growth Rate (AGR) per species, obtained from the following equation: (dbh1-dbh0)/time, where dbh1 means dbh in mm in the second census, dbh0 is the dbh in mm in the first census, and time is the time census interval in years

[5] SD.growth.rate: Standard deviation from the mean of the AGR per species

[6] SLA: Specific Leaf Area, in cm2/g

[7] LA: Leaf Area, in cm2

[8] LDMC: Leaf Dry-Matter Content, in mg/g

[9] LNC: Leaf Nitrogen Content, in %

[10] LPC: Leaf Phosphorus Content, in %

[11] WD: Wood Density, in g/cm3

[12] Height: Height, in meters

[13] Number.of.stems: Number of stems per species across the dataset

[14] Median.height: Species median height across the dataset

[15] Category: Categorisation as canopy or understory"

The R script contains the code below. To run the code, please remember to load the `species.information.csv` file.

```r
### PLEASE NOTE that PC2 for understory was reversed for visualisation porpuses in the main file.

# load packages and functions -----
library(here)
library(tidyverse)
library(corrplot)
library(MuMIn)
library(ggfortify)
library(lmerTest)
library(visreg)
library(ggpubr)
library(gridExtra)
library(vegan)
library(dplyr)

# function to estimate the effect of ecological strategies on tree growth
models <- function(data.models, variable){
  #creading a list to save each model
  model0 <- list()
  model1 <- list()
  model2 <- list()
  model3 <- list()
  model4 <- list()
  model5 <- list()
  model6 <- list()
  model7 <- list()

  model.selection <- NULL

  if(variable == "rgr"){
    m0 <- lm(log(rgr) ~1, data = data.models) #null model
    m1 <- lm(log(rgr)~PC1, data=data.models)
    m2 <- lm(log(rgr)~PC2, data=data.models)
    m3 <- lm(log(rgr)~PC3, data=data.models)
    m4 <- lm(log(rgr)~PC1+PC2, data=data.models)
    m5 <- lm(log(rgr)~PC1+PC2+PC3, data=data.models)
    m6 <- lm(log(rgr)~PC1+PC3, data = data.models)
    m7 <- lm(log(rgr)~PC2+PC3, data = data.models)
  }
  model.selection = as.data.frame(model.sel(m0, m1,m2,m3,m4,m5,m6,m7))
  model0 <- m0
  model1 <- m1
  model2 <- m2
  model3 <- m3
  model4 <- m4
  model5 <- m5
  model6 <- m6
  model7 <- m7

  return(list(model.selection = model.selection,
              m0=model0,
```

```r
            m1=model1,
            m2=model2,
            m3=model3,
            m4=model4,
            m5=model5,
            m6=model6,
            m7=model7))
}

# function to generate plots for the PCAs generated
pca.vis <- function(data,pca,axis1,axis2,name){
  autoplot(pca,
           label= F, # keep as false
           data=data, # original data (csv file)
           colour="rgr", # colour of the shapes
           label.size=4, # label sizes
           shape="Forest.stratum", # distincts canopy from understory species
           size="rgr", # size of the shapes
           loadings=TRUE, # includes PCA loads
           loadings.label.repel = TRUE, # repel labels within the PCA
           loadings.colour="black", # arrow colour
           loadings.label=TRUE, # variables names
           loadings.label.size=4, # size of labels
           loadings.label.colour="black", # label colours
           x=axis1, y=axis2) + #PC axes to be plotted
    my_theme+
    theme(aspect.ratio=1)+
    theme(text = element_text(size=15))+
    scale_colour_gradient(low = "#F0E442",high = "#009E73")+ # gradient of colours
    geom_vline(xintercept=0, color="gray80", linetype="dotted") +
    geom_hline(yintercept=0, color="gray80", linetype="dotted") +
    xlim(-0.4,0.4)+
    theme(plot.title=element_text(size=14,face="bold",hjust = 0.5))+
    labs(title=name)

}

# theme for plots
my_theme <- theme_light() +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank()) +
  theme(legend.position = "none",
        text = element_text(size=15),
        plot.margin=unit(c(0.1, 1, 0.1 , 0.1), "cm"),
        axis.text.x = element_text(size=15),
        axis.text.y = element_text(size = 15))

# load dataset -----
data <- read.csv(here::here("data", "species.information.csv"))

data <- data %>%
  mutate(rgr = Relative.Growth.Rate,
```

```
        LA = log(LA))

# data filter -----
dat.total = data %>%
  mutate(Forest.stratum=Category)
dat.underst = data %>%
  filter(Category == "Understory") %>%
  mutate(Forest.stratum=Category)
dat.canop = data %>%
  filter(Category == "Canopy") %>%
  mutate(Forest.stratum=Category)

# PCAs -----
pca.tot <- dat.total %>%
  select(WD,SLA,LNC,LPC,LDMC,Height,LA)
pca.tot <- prcomp((pca.tot), center=T, scale=T)
pca.tot$rotation; summary(pca.tot)
```

```
##                   PC1         PC2         PC3          PC4        PC5          PC6
## WD         0.2443495 -0.29307647  0.78038647 -0.287917314  0.3125059 -0.02353600
## SLA       -0.4974965  0.07092055  0.07561376 -0.369482234  0.4345247 -0.10701346
## LNC       -0.4714853 -0.28457746  0.15637094 -0.171583319 -0.4334161  0.67294985
## LPC       -0.4838034 -0.33479691  0.00909808 -0.043383558 -0.2857066 -0.69749982
## LDMC       0.4159151 -0.49595346  0.04626948  0.003559261 -0.4037945 -0.14531742
## Height    -0.1813472 -0.52212582 -0.05537034  0.682470385  0.4546754  0.13399879
## LA        -0.1747977  0.44431740  0.59627019  0.532433127 -0.2725266 -0.09774379
##                   PC7
## WD         0.25354718
## SLA       -0.63637202
## LNC        0.04597306
## LPC        0.28938918
## LDMC      -0.62827068
## Height    -0.03131706
## LA        -0.22176578

## Importance of components:
##                            PC1    PC2    PC3    PC4     PC5     PC6     PC7
## Standard deviation      1.5832 1.1647 0.9771 0.9446 0.76846 0.61574 0.56583
## Proportion of Variance  0.3581 0.1938 0.1364 0.1275 0.08436 0.05416 0.04574
## Cumulative Proportion   0.3581 0.5519 0.6883 0.8157 0.90010 0.95426 1.00000
```

```
pca.und <- dat.underst %>%
  select(WD,SLA,LNC,LPC,LDMC,Height,LA)
pca.und <- prcomp((pca.und), center=T, scale=T)
pca.und$rotation; summary(pca.und)
```

```
##                   PC1         PC2         PC3          PC4         PC5          PC6
## WD         0.15353194  0.43534695 -0.73994011  0.13784604 -0.35949634  0.05326392
## SLA       -0.50006311  0.06793126  0.06298567 -0.09351331 -0.70344925 -0.14203316
## LNC       -0.50266312  0.24588652 -0.06499206 -0.11090288  0.25823898  0.77604463
## LPC       -0.41170420  0.39237501 -0.15931452 -0.29855704  0.45324491 -0.59613678
## LDMC       0.47672209  0.36271694 -0.09619232 -0.07884483  0.18981548  0.07434413
## Height    -0.06943068  0.52363809  0.44482015  0.71606291  0.01177756 -0.07176604
## LA        -0.26853250 -0.43090558 -0.46021564  0.59317399  0.26006649 -0.09320841
##                   PC7
```

4

```
## WD     -0.29717574
## SLA     0.46647190
## LNC     0.03759733
## LPC    -0.03484354
## LDMC    0.76429447
## Height -0.07129097
## LA      0.31974125

## Importance of components:
##                          PC1    PC2    PC3     PC4     PC5     PC6     PC7
## Standard deviation    1.6504 1.2828 0.9802 0.81784 0.70639 0.52571 0.47531
## Proportion of Variance 0.3891 0.2351 0.1373 0.09555 0.07128 0.03948 0.03227
## Cumulative Proportion  0.3891 0.6242 0.7614 0.85696 0.92824 0.96773 1.00000
```

```r
pca.canop <- dat.canop %>%
  select(WD,SLA,LNC,LPC,LDMC,Height,LA)
pca.canop <- prcomp((pca.canop), center=T, scale=T)
pca.canop$rotation; summary(pca.canop)
```

```
##                  PC1         PC2         PC3         PC4        PC5         PC6
## WD      0.354733857 -0.11271208  0.50375258 -0.17041090 -0.7082462 -0.01867387
## SLA    -0.498832414  0.03952174  0.34925095  0.08870364 -0.2306199 -0.51376889
## LNC    -0.383150666 -0.49153948  0.39480303 -0.03825775  0.1180672  0.65522607
## LPC    -0.542069584 -0.11219386  0.03232505 -0.13960989  0.1028309 -0.32813185
## LDMC    0.293363161 -0.53224808  0.02078247 -0.59713182  0.3273792 -0.36370239
## Height -0.313970559  0.12044458 -0.49362029 -0.61364276 -0.4420329  0.22499933
## LA      0.006840354  0.65858985  0.47249325 -0.45717979  0.3436823  0.12568600
##                 PC7
## WD     -0.276933199
## SLA     0.550105163
## LNC     0.104834879
## LPC    -0.744843993
## LDMC    0.184919322
## Height  0.143803227
## LA     -0.005904489

## Importance of components:
##                          PC1    PC2    PC3    PC4    PC5    PC6     PC7
## Standard deviation    1.6147 1.0859 0.9908 0.9227 0.8494 0.6199 0.52413
## Proportion of Variance 0.3725 0.1685 0.1402 0.1216 0.1031 0.0549 0.03924
## Cumulative Proportion  0.3725 0.5409 0.6812 0.8028 0.9059 0.9608 1.00000
```

```r
# data for analysis ----
dat.total <- dat.total %>%
  mutate(PC1 = (pca.tot$x[,1])) %>%
  mutate(PC2 = (pca.tot$x[,2])) %>%
  mutate(PC3 = (pca.tot$x[,3]))

dat.underst <- dat.underst %>%
  mutate(PC1 = (pca.und$x[,1])) %>%
  mutate(PC2 = (pca.und$x[,2])) %>%
  mutate(PC3 = (pca.und$x[,3]))

dat.canop <- dat.canop %>%
  mutate(PC1 = (pca.canop$x[,1])) %>%
  mutate(PC2 = (pca.canop$x[,2])) %>%
```

```
  mutate(PC3 = (pca.canop$x[,3]))

# growth models -----
set.seed(999)

# all species
data.models <- dat.total
total <- models(data.models = data.models, variable = "rgr")
total.modelselection <- total$model.selection %>% mutate(resu.from = "total.model")
total_best.model <- total$m2 # most parcimonious model
boot.total = car::Boot(total_best.model, R=9999)
CI.total = confint(boot.total); CI.total
```

```
## Bootstrap bca confidence intervals
##
##                  2.5 %      97.5 %
## (Intercept) -1.4229962 -1.33010927
## PC2         -0.1040547 -0.02874394
```

```
summary(total_best.model)
```

```
##
## Call:
## lm(formula = log(rgr) ~ PC2, data = data.models)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.44639 -0.16103 -0.04717  0.09694  1.21212
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.38164    0.02385 -57.924  < 2e-16 ***
## PC2         -0.06462    0.02056  -3.142  0.00212 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2624 on 119 degrees of freedom
## Multiple R-squared:  0.07662,    Adjusted R-squared:  0.06886
## F-statistic: 9.874 on 1 and 119 DF,  p-value: 0.002116
```

```
r2.total = RsquareAdj(total_best.model)
results.total <- broom::tidy(total_best.model) %>%
  mutate(`CI 2.5%` = confint(boot.total)[,1],
         `CI 97.5%` = confint(boot.total)[,2])%>%
  slice(-1) %>%
  mutate(`R2 adj` = r2.total$adj.r.squared,
         resu.from = "total.model"); results.total
```

```
## # A tibble: 1 x 9
##   term  estimate std.error statistic p.value `CI 2.5%` `CI 97.5%` `R2 adj`
##   <chr>    <dbl>     <dbl>     <dbl>   <dbl>     <dbl>      <dbl>    <dbl>
## 1 PC2    -0.0646    0.0206     -3.14 0.00212    -0.104    -0.0287   0.0689
## # i 1 more variable: resu.from <chr>
```

```
# understory species
data.models <- dat.underst
```

```
unders <- models(data.models = data.models, variable = "rgr")
unders.modelselection <- unders$model.selection%>% mutate(resu.from = "understory.model")
unders_best.model <- unders$m2  # most parcimonious model
boot.unders = car::Boot(unders_best.model, R=9999)
CI.unders = confint(boot.unders); CI.unders
```

```
## Bootstrap bca confidence intervals
##
##                  2.5 %      97.5 %
## (Intercept) -1.38927213 -1.2607656
## PC2          0.04370804  0.1377501
```

```
summary(unders_best.model)
```

```
##
## Call:
## lm(formula = log(rgr) ~ PC2, data = data.models)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37058 -0.15171 -0.06700  0.08513  1.19439
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.33800    0.03150 -42.478  < 2e-16 ***
## PC2          0.08799    0.02474   3.557 0.000694 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2616 on 67 degrees of freedom
## Multiple R-squared:  0.1589, Adjusted R-squared:  0.1463
## F-statistic: 12.65 on 1 and 67 DF,  p-value: 0.0006938
```

```
r2.und = RsquareAdj(unders_best.model)
results.und <- broom::tidy(unders_best.model) %>%
  mutate(`CI 2.5%` = confint(boot.unders)[,1],
         `CI 97.5%` = confint(boot.unders)[,2])%>%
  slice(-1) %>%
  mutate(`R2 adj` = r2.und$adj.r.squared,
         resu.from = "understory.model"); results.und
```

```
## # A tibble: 1 x 9
##   term  estimate std.error statistic  p.value `CI 2.5%` `CI 97.5%` `R2 adj`
##   <chr>    <dbl>     <dbl>     <dbl>    <dbl>     <dbl>      <dbl>    <dbl>
## 1 PC2     0.0880    0.0247      3.56 0.000694    0.0437      0.138    0.146
## # i 1 more variable: resu.from <chr>
```

```
# canopy species
data.models <- dat.canop
canop <- models(data.models = data.models, variable = "rgr")
canop.modelselection <- canop$model.selection %>% mutate(resu.from = "canopy.model")
canop_best.model <- canop$m6  # most parcimonious model
boot.canop = car::Boot(canop_best.model, R=9999)
CI.canop = confint(boot.canop); CI.canop
```

```
## Bootstrap bca confidence intervals
```

```
##
##                    2.5 %        97.5 %
## (Intercept) -1.49175728 -1.370656229
## PC1         -0.09909675  0.005043703
## PC3         -0.17391935 -0.050602092
```
```r
summary(canop_best.model)
```
```
##
## Call:
## lm(formula = log(rgr) ~ PC1 + PC3, data = data.models)
##
## Residuals:
##      Min      1Q  Median      3Q     Max
## -0.30371 -0.13658 -0.04186  0.07318  0.58988
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.43956    0.03048 -47.231  < 2e-16 ***
## PC1         -0.03841    0.01906  -2.015  0.04940 *
## PC3         -0.10480    0.03106  -3.374  0.00146 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2198 on 49 degrees of freedom
## Multiple R-squared:  0.2396, Adjusted R-squared:  0.2086
## F-statistic: 7.721 on 2 and 49 DF,  p-value: 0.001216
```
```r
r2.can = RsquareAdj(canop_best.model)
results.can <- broom::tidy(canop_best.model) %>%
  mutate(`CI 2.5%` = confint(boot.canop)[,1],
         `CI 97.5%` = confint(boot.canop)[,2])%>%
  slice(-1) %>%
  mutate(`R2 adj` = r2.can$adj.r.squared,
         resu.from = "canopy.model"); results.can
```
```
## # A tibble: 2 x 9
##   term  estimate std.error statistic p.value `CI 2.5%` `CI 97.5%` `R2 adj`
##   <chr>    <dbl>     <dbl>     <dbl>   <dbl>     <dbl>      <dbl>    <dbl>
## 1 PC1    -0.0384    0.0191    -2.02 0.0494    -0.0991    0.00504    0.209
## 2 PC3    -0.105     0.0311    -3.37 0.00146   -0.174    -0.0506     0.209
## # i 1 more variable: resu.from <chr>
```
```r
results = bind_rows(results.total,results.und,results.can)
model.sel = bind_rows(total.modelselection, unders.modelselection,canop.modelselection)


# PCA plots ----
a <- pca.vis(data = dat.total,pca = pca.tot,axis1 = 1,axis2 = 2, name = "All")+labs(tag="a)")
b <- pca.vis(data = dat.total,pca = pca.tot,axis1 = 1,axis2 = 3, name = "All")+labs(tag="d)")
c <- pca.vis(data = dat.underst,pca = pca.und,axis1 = 1,axis2 = 2, name = "Understory")+labs(tag="c)")+
d <- pca.vis(data = dat.underst,pca = pca.und,axis1 = 1,axis2 = 3, name = "Understory")+labs(tag="f)")+
e <- pca.vis(data = dat.canop,pca = pca.canop,axis1 = 1,axis2 = 2, name = "Canopy")+labs(tag="b)")
f <- pca.vis(data = dat.canop,pca = pca.canop,axis1 = 1,axis2 = 3, name = "Canopy")+labs(tag="e)")

#png('results/principal.components.plot.png', units="in", width=13, height=8, res=300)
```
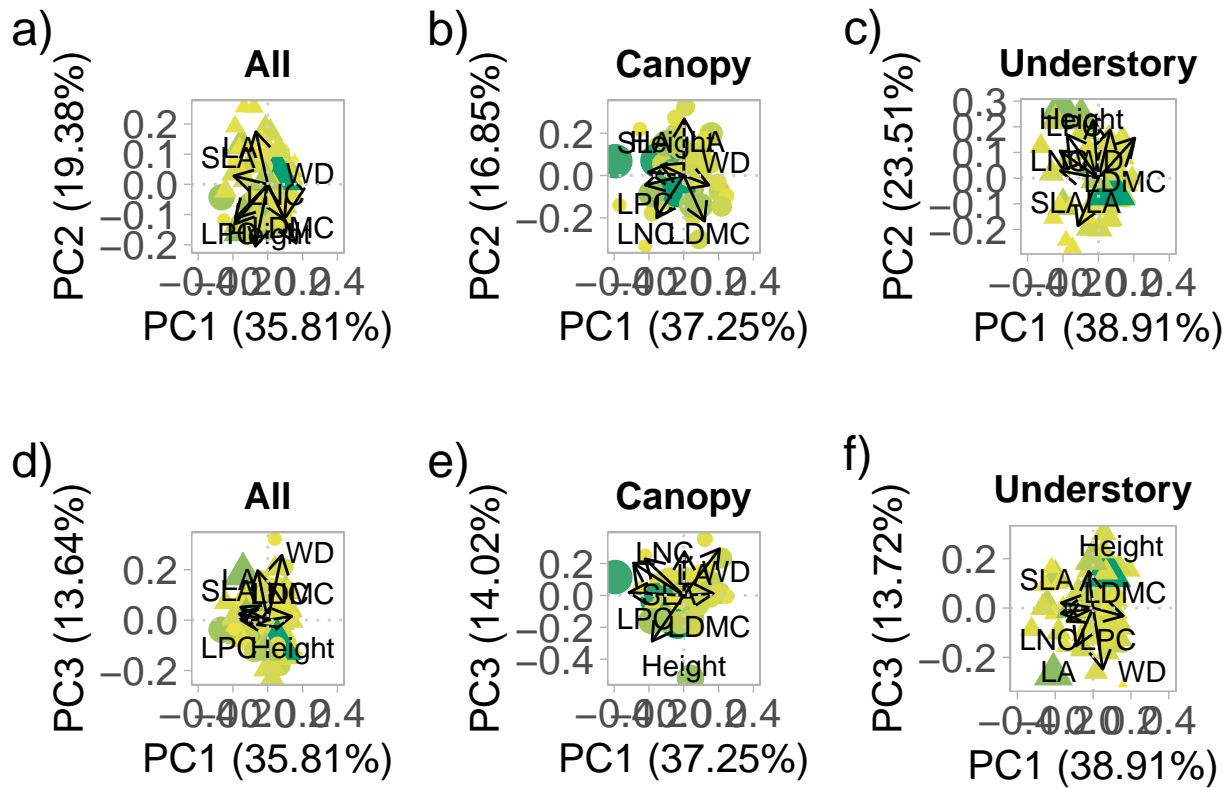
```
grid.arrange(a,e,c,b,f,d, ncol=3)
```


a) **All** — PC2 (19.38%), PC1 (35.81%)

b) **Canopy** — PC2 (16.85%), PC1 (37.25%)

c) **Understory** — PC2 (23.51%), PC1 (38.91%)

d) **All** — PC3 (13.64%), PC1 (35.81%)

e) **Canopy** — PC3 (14.02%), PC1 (37.25%)

f) **Understory** — PC3 (13.72%), PC1 (38.91%)

```
#dev.off()
```