

```

#load in libraries

library(csv)
library(readr)
library(tidyverse)
library(janitor)
library(ggpubr)
library(ggplot2)
library(FactoMineR)
library(factoextra) #for viewing plots in FactoMineR
library(corrplot) #for viewing correlation plots


rm(list = ls()) #clears working environment

##### Loading and Manipulating the data #####

path = '/Users/tessavery/Dropbox/jobs/Lauder Lab/Denticles/Sleeper
Shark Project/Morphotype Final Figures/PCA'
setwd(path)

denticles_raw <- read_csv("denticle_data_FINAL_march-20-2023.csv") %>%
  janitor::clean_names() #this cleans/synchronizes all column names

#View(denticles_raw) #this views the whole dataset

#selecting for the columns we want
denticles <- denticles_raw %>%
  select(updated_file_names, denticle_group, morphotype,
sample_number, total_body_legnth_mm, surface_area_mm2, volume_mm3,
crown_length_um, crown_width_um, central_cusp_length_um,
central_cusp_width_um, lateral_cusp_length_um, lateral_cusp_width_um,
total_base_legnth_um, total_base_width_um, number_longitudinal_ridges,
crown_angle_degrees)

#selecting the rows we want
denticles <- denticles %>%
  filter(sample_number == 1 | sample_number == 2 | sample_number == 3
| sample_number == 4 | sample_number == 5 | sample_number == 6)

#view(denticles)

#converts measurements in um to new columns in mm
denticles_clean <- denticles %>%
  mutate(crown_length_mm = denticles$crown_length_um/1000) %>%
  mutate(crown_width_mm = denticles$crown_width_um/1000)%>%
  mutate(central_cusp_length_mm = denticles$central_cusp_length_um/
1000)%>%
  mutate(central_cusp_width_mm = denticles$central_cusp_width_um/
1000)%>%

```

```

    mutate(lateral_cusp_length_mm = denticles$lateral_cusp_length_um/
1000)%>%
    mutate(lateral_cusp_width_mm = denticles$lateral_cusp_width_um/
1000)%>%
    mutate(base_length_mm = denticles$total_base_legnth_um/1000)%>%
    mutate(base_width_mm = denticles$total_base_width_um/1000)

#view(denticles_clean)

#making the active data to run through the PCA-- only the columns with
measurements
denticles_final <- denticles_clean %>%
    select(!c(updated_file_names, denticle_group, morphotype,
sample_number, total_body_legnth_mm, crown_length_um, crown_width_um,
central_cusp_length_um, central_cusp_width_um, lateral_cusp_length_um,
lateral_cusp_width_um, total_base_legnth_um, total_base_width_um))
#selects all except these ones

#view(denticles_final)

##### Running the Principal Component Analysis #####

#calculating the PCA of the active data
res2.pca <- PCA(denticles_final, graph = FALSE) #NA's are imputed by
the mean of the variable

#determine the number of eigenvalues to use in the PCA
eig.val <- get_eigenvalue(res2.pca)
eig.val #with this data, we will have 12 eigenvalues because we have
12 variables, so 12 dimensions

#scree plot
fviz_eig(res2.pca, addlabels = TRUE, ylim = c(0,50)) #Extract and
visualize the eigenvalues/variances of dimensions

#extracting PCA results
var <- get_pca_var(res2.pca) #creates a data frame called var that has
all the results (coordinates, squared cosine, contributions) for the
active variables from our Principal Component Analysis (PCA) output.

#### Visualizations of the PCA data ####

#Correlation circle
fviz_pca_var(res2.pca, col.var = "black", geom = c("arrow", "text"))

```

```

#this plots the PCA graph of the variables and where they push shape
space

#quality of representation
colnames(var$cos2) <- c("Dim 1", "Dim 2", "Dim 3", "Dim 4", "Dim 5")
#renames the columns for ease
rownames(var$cos2) <- c("Surface Area", "Volume", "Number Longitudinal
Ridges", "Crown Angle", "Crown Length", "Crown Width", "Central Cusp
Length", "Central Cusp Width", "Lateral Cusp Length", "Lateral Cusp
Width", "Base Length", "Base Width") #renames the rows for ease
corrplot(var$cos2, #visualization of a correlation matrix of cos2.
          method = 'circle', #makes circles
          is.corr=FALSE, #sets scale to 0-1
          cl.pos="r", cl.ratio= 0.3, #sets position and width of the
legend
          tl.col = "black",
          col=COL1('Blues', 10)) #sets the color palette

#Total cos2 of variables on Dim1 and Dim2
fviz_cos2(res2.pca, choice="var", axes = 1:2, color="black") #bar
graph of our variables and their cos2 number. Higher cos2 numbers mean
that that variable is better represented on the PC axis. So in this
case, the surface area is the variable most represented by PC1 and 2

#Color by cos2 values:quality on the factor map
fviz_pca_var(res2.pca, col.var = "cos2", gradient.cols = c("#00AFBB",
"#E7B800", "#FC4E07"), repel=TRUE)

#PCA breakdown using factoextra package
#fviz_eig(res2.pca) #we already saw this
fviz_pca_ind(res2.pca,
             col.ind = "cos2", gradient.cols =
c("#00AFBB", "#E7B800", "#FC4E07"), repel = TRUE)

fviz_pca_var(res2.pca,
             col.var = "contrib",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE)

fviz_pca_biplot(res2.pca, repel = TRUE,
               col.var = "#2E9FDF",
               col.ind = "#696969")

##### PCA plots that sort by color #####

```

```

#PCA's where color shows body length
groups <- as.factor(denticles_clean$total_body_length_mm)

fviz_pca_ind(res2.pca,
             geom.ind = "point",
             col.ind = groups,
             palette = c("#1B9E77", "#D95F02", "#7570B3", "#E7298A",
"#66A61E"),
             addEllipse = TRUE,
             ellipse.type = "confidence",
             mean.point=FALSE,
             legend.title = "Body Length (mm)",
             repel = TRUE)

fviz_pca_biplot(res2.pca,
               geom.ind = "point",
               col.ind = groups, palette = c("#1B9E77", "#D95F02",
"#7570B3", "#E7298A", "#66A61E"),
               addEllipse = TRUE,
               ellipse.type = "confidence",
               mean.point=FALSE,
               label = "var",
               col.var = "black", repel = TRUE,
               legend.title = "Body Length (mm)")

# PCA's where color shows morphotype

#reorder for legend
denticles_clean$morphotype <- factor(denticles_clean$morphotype,
levels = c("SN1", "SN2", "SN3", "SP1", "SP2", "SP3", "SP4", "T1",
"T2", "T3", "B4"))

#group by morphotype
groups <- as.factor(denticles_clean$morphotype)

fviz_pca_ind(res2.pca,
             geom.ind = "point",
             col.ind = groups,
             palette = c("#ffe767", "orange", "coral2", "turquoise",
'mediumspringgreen', '#A3DE34', '#D9E650', 'skyblue',
'cornflowerblue', 'slateblue1', "darkorchid1"),
             addEllipse = TRUE,
             ellipse.type = "confidence",
             mean.point=FALSE,
             legend.title = "Morphotype",
             repel = TRUE)

```

```

fviz_pca_biplot(res2.pca,
                geom.ind = "point",
                col.ind = groups,
                palette = c("#ffe767", "orange", "coral2",
"turquoise", 'mediumspringgreen', '#A3DE34', '#D9E650', 'skyblue',
'cornflowerblue', 'slateblue1', "darkorchid1"),
                addEllipse = TRUE,
                ellipse.type = "confidence",
                mean.point=FALSE,
                label = "var",
                col.var = "grey40", repel = TRUE,
                legend.title = "Morphotype")

```

#Running the PCA without any of the columns that have missing data

#select just columns we want for pca -- so in this case, we want
volume, surface area, CW, CL, BL, BW, crown angle, longituinal ridges

```

nomissing_denticles_pca <- denticles_clean %>%
  select(c(volume_mm3, surface_area_mm2, crown_width_mm,
crown_length_mm, base_length_mm, base_width_mm, crown_angle_degrees,
number_longitudinal_ridges))

```

#run the PCA

```

res2.pca.nomissing <- PCA(nomissing_denticles_pca, graph = FALSE)

```

#group by morphotype

```

groups <- as.factor(denticles_clean$morphotype)

```

#rename columns for pretty text

```

rownames(res2.pca.nomissing$var$coord) <- c("Volume", "Surface Area",
"Crown Width", "Crown Length", "Base Length", "Base Width", "Crown
Angle", "Number Longitudinal Ridges")

```

```

fviz_pca_ind(res2.pca.nomissing,
            geom.ind = "point",
            col.ind = groups,
            palette = c("#ffe767", "orange", "coral2", "turquoise",
'mediumspringgreen', '#A3DE34', '#D9E650', 'skyblue',
'cornflowerblue', 'slateblue1', "darkorchid1"),
            addEllipse = TRUE,
            ellipse.type = "confidence",
            mean.point=FALSE,

```

```

        legend.title = "Morphotype",
        title = "Denticles PCA (no missing values)",
        repel = TRUE)

#this one colors the lines depending on cos2 value
fviz_pca_biplot(res2.pca.nomissing,
                geom.ind = "point",
                habillage = groups,
                palette = c("#ffe767", "orange", "coral2",
"turquoise", 'mediumspringgreen', '#A3DE34', '#D9E650', 'skyblue',
'cornflowerblue', 'slateblue1', "darkorchid1"),
                addEllipse = TRUE,
                ellipse.type = "confidence",
                mean.point=FALSE,
                label = "var",
                alpha.var = "cos2",
                col.var = "black",
                labelsize=3,
                legend.title = "Morphotype",
                title = "Denticles PCA biplot (no missing values)")

##### PCA plots separated by body location #####

#### Snout ####

#create data set with just snout denticles
snout_denticles <- filter(denticles_clean, morphotype == "SN1" |
morphotype == "SN2" | morphotype == "SN3")
#view(snout_denticles)

#select just active data in columns to run through the PCA

snout_denticles_pca <- snout_denticles %>%
  select(c(volume_mm3, surface_area_mm2, crown_width_mm,
crown_length_mm, base_length_mm, base_width_mm, crown_angle_degrees,
number_longitudinal_ridges))
#view(snout_denticles_pca)

#run the PCA on snout denticles
res2.pca.snout <- PCA(snout_denticles_pca, graph = FALSE)

#group by morphotype
groups <- as.factor(snout_denticles$morphotype)

#renames the variable columns for graphing
rownames(res2.pca.snout$var$coord) <- c("Volume", "Surface Area",
"Crown Width", "Crown Length", "Base Length", "Base Width", "Crown
Angle", "Number Longitudinal Ridges")

```

```

fviz_pca_ind(res2.pca.snout,
             geom.ind = "point",
             col.ind = groups,
             palette = c("#ffe767", "orange", "coral2"),
             addEllipse = TRUE,
             ellipse.type = "confidence",
             mean.point=FALSE,
             legend.title = "Morphotypes",
             repel = TRUE,
             title = "Snout denticles PCA")

#this one colors the lines depending on cos2 value
fviz_pca_biplot(res2.pca.snout,
               geom.ind = "point",
               habillage = groups,
               palette = c("#ffe767", "orange", "coral2"),
               addEllipse = TRUE,
               ellipse.type = "confidence",
               mean.point=FALSE,
               label = "var",
               #select.var = list(cos2 = 0.9),
               alpha.var = "cos2",
               col.var = "black",
               labelsize=3,
               legend.title = "Morphotype",
               title = "Snout denticles PCA biplot")

```

Spiracle

```

#create data set with just spiracle denticles
spiracle_denticles <- filter(denticles_clean, morphotype == "SP1" |
morphotype == "SP2" | morphotype == "SP3" | morphotype == "SP4")
#view(spiracle_denticles)

#select just active data to run through the PCA

spiracle_denticles_pca <- spiracle_denticles %>%
  select(c(volume_mm3, surface_area_mm2, crown_width_mm,
crown_length_mm, base_length_mm, base_width_mm, crown_angle_degrees,
number_longitudinal_ridges, central_cusp_length_mm,
central_cusp_width_mm, lateral_cusp_length_mm, lateral_cusp_width_mm))
#view(spiracle_denticles_pca)

#run the PCA on spiracle denticles
res2.pca.spiracle <- PCA(spiracle_denticles_pca, graph = FALSE)

#reorder for legend

```

```
spiracle_denticles$morphotype <- factor(spiracle_denticles$morphotype,  
levels = c("SP1", "SP2", "SP3", "SP4"))
```

```
#group by morphotype  
groups_spiracle <- as.factor(spiracle_denticles$morphotype)
```

```
#rename columns for ease  
rownames(res2.pca.spiracle$var$coord) <- c("Volume", "Surface Area",  
"Crown Width", "Crown Length", "Base Length", "Base Width", "Crown  
Angle", "Number Longitudinal Ridges", "Central Cusp Length", "Central  
Cusp Width", "Lateral Cusp Length", "Lateral Cusp Width")
```

```
fviz_pca_ind(res2.pca.spiracle,  
             geom.ind = "point",  
             col.ind = groups_spiracle,  
             palette = c("turquoise", 'mediumspringgreen', '#A3DE34',  
                         '#D9E650'),  
             addEllipse = TRUE,  
             ellipse.type = "confidence",  
             mean.point=FALSE,  
             legend.title = "Morphotype",  
             title = "Spiracle denticles PCA",  
             repel = TRUE)
```

```
#this one colors the lines depending on cos2 value
```

```
fviz_pca_biplot(res2.pca.spiracle,  
               geom.ind = "point",  
               habillage = groups_spiracle,  
               palette = c("turquoise", 'mediumspringgreen',  
                           '#A3DE34', '#D9E650'),  
               addEllipse = TRUE,  
               ellipse.type = "confidence",  
               mean.point=FALSE,  
               label = "var",  
               alpha.var = "cos2",  
               col.var = "black",  
               labelsize=3,  
               legend.title = "Morphotype",  
               title = "Spiracle denticles PCA biplot")
```

```
#### Trunk ####
```

```
#create dataset with just trunk denticles  
trunk_denticles <- filter(denticles_clean, morphotype == "T1" |  
morphotype == "T2" | morphotype == "T3" | morphotype == "B4")  
#view(trunk_denticles)
```



```

#select just columns with active data to run in the PCA

trunk_denticles_pca <- trunk_denticles %>%
  select(c(volume_mm3, surface_area_mm2, crown_width_mm,
crown_length_mm, base_length_mm, base_width_mm, crown_angle_degrees,
number_longitudinal_ridges, central_cusp_length_mm,
central_cusp_width_mm, lateral_cusp_length_mm, lateral_cusp_width_mm))
#view(trunk_denticles_pca)

#run the PCA on snout denticles
res2.pca.trunk <- PCA(trunk_denticles_pca, graph = FALSE)

#reorder morphotypes for legend
trunk_denticles$morphotype <- factor(trunk_denticles$morphotype,
levels = c("T1", "T2", "T3", "B4"))
#group by morphotype
groups <- as.factor(trunk_denticles$morphotype)

#rename columns for ease
rownames(res2.pca.trunk$var$coord) <- c("Volume", "Surface Area",
"Crown Width", "Crown Length", "Base Length", "Base Width", "Crown
Angle", "Number Longitudinal Ridges", "Central Cusp Length", "Central
Cusp Width", "Lateral Cusp Length", "Lateral Cusp Width")

fviz_pca_ind(res2.pca.trunk,
  geom.ind = "point",
  col.ind = groups,
  palette = c('skyblue', 'cornflowerblue', 'slateblue1',
"darkorchid1"),
  addEllipse = TRUE,
  ellipse.type = "confidence",
  mean.point=FALSE,
  legend.title = "Morphotype",
  title = "Trunk denticles PCA",
  repel = TRUE)

#this one colors the lines depending on cos2 value
fviz_pca_biplot(res2.pca.trunk,
  geom.ind = "point",
  habillage = groups,
  palette = c('skyblue', 'cornflowerblue', 'slateblue1',
"darkorchid1"),
  addEllipse = TRUE,
  ellipse.type = "confidence",
  mean.point=FALSE,
  label = "var",
  alpha.var = "cos2",
  col.var = "black",
  labelsize=3,

```

```
legend.title = "Morphotype",  
title = "Trunk denticles PCA biplot")
```

```
#### Statistics ####
```

```
#Question 1. Within the same morphotype, how does body length affect  
the size of the denticle?
```

```
#Use a linear regression to plot/regress body length versus some  
measure of denticle size and then compare the slopes for all the  
morphotypes
```

```
#using crown length as a proxy for "denticle size"
```

```
#first subset the data for morphotype
```

```
SN1 <- subset(snout_denticles, morphotype=="SN1")
```

```
SN2 <- subset(snout_denticles, morphotype=="SN2")
```

```
SN3 <- subset(snout_denticles, morphotype=="SN3")
```

```
T1 <- subset(trunk_denticles, morphotype=="T1")
```

```
T2 <- subset(trunk_denticles, morphotype=="T2")
```

```
T3 <- subset(trunk_denticles, morphotype=="T3")
```

```
B4 <- subset(trunk_denticles, morphotype=="B4")
```

```
SP1 <- subset(spiracle_denticles, morphotype=="SP1")
```

```
SP2 <- subset(spiracle_denticles, morphotype=="SP2")
```

```
SP3 <- subset(spiracle_denticles, morphotype=="SP3")
```

```
SP4 <- subset(spiracle_denticles, morphotype=="SP4")
```

```
#snout
```

```
#trial on a single morphoype
```

```
# Run linear regression with "size" as the dependent variable and  
total body length as the independent variable
```

```
lm_model_SN1 <- lm(crown_length_mm ~ total_body_legnth_mm, data = SN1)
```

```
summary(lm_model_SN1) #We want to look at the Pr(>|t|) value for total  
body length (this will give us significance) and the estimate gives us  
the slope (or the effect of body L on crown L in SN1)
```

```
#so for snout morphotype 1, we see a somewhat significant positive  
correlation between body length and crown length: as the body length  
increases, so does the crown length
```

```
cor(SN1$total_body_legnth_mm, SN1$crown_length_mm)
```

```
#plot just to check and see what it looks like / is working correctly
```

```
plot(crown_length_mm ~ total_body_legnth_mm, data = SN1)
```

```
abline(lm_model_SN1)
```

```
#SN2
lm_model_SN2 <- lm(crown_length_mm ~ total_body_legnth_mm, data = SN2)
summary(lm_model_SN2) #We want to look at the Pr(>|t|) value for total
body length (this will give us significance) and the estimate gives us
the slope (or the effect of body L on crown L in SN2)
cor(SN2$total_body_legnth_mm, SN2$crown_length_mm)
```

```
#SN3
lm_model_SN3 <- lm(crown_length_mm ~ total_body_legnth_mm, data = SN3)
summary(lm_model_SN3) #We want to look at the Pr(>|t|) value for total
body length (this will give us significance) and the estimate gives us
the slope (or the effect of body L on crown L in SN3)
cor(SN3$total_body_legnth_mm, SN3$crown_length_mm)
```

```
#spiracle
```

```
#SP1
lm_model_SP1 <- lm(crown_length_mm ~ total_body_legnth_mm, data = SP1)
summary(lm_model_SP1) #We want to look at the Pr(>|t|) value for total
body length (this will give us significance) and the estimate gives us
the slope (or the effect of body L on crown L in SP1)
cor(SP1$total_body_legnth_mm, SP1$crown_length_mm)
```

```
#SP2
lm_model_SP2 <- lm(crown_length_mm ~ total_body_legnth_mm, data = SP2)
summary(lm_model_SP2) #We want to look at the Pr(>|t|) value for total
body length (this will give us significance) and the estimate gives us
the slope (or the effect of body L on crown L in SP2)
cor(SP2$total_body_legnth_mm, SP2$crown_length_mm)
```

```
#SP3
lm_model_SP3 <- lm(crown_length_mm ~ total_body_legnth_mm, data = SP3)
summary(lm_model_SP3) #We want to look at the Pr(>|t|) value for total
body length (this will give us significance) and the estimate gives us
the slope (or the effect of body L on crown L in SP3)
cor(SP3$total_body_legnth_mm, SP3$crown_length_mm)
```

```
#SP4 we don't have enough data here because we only saw SP4
morphotypes in one specimen -- no results
```

```
lm_model_SP4 <- lm(crown_length_mm ~ total_body_legnth_mm, data = SP4)
summary(lm_model_SP4) #We want to look at the Pr(>|t|) value for total
body length (this will give us significance) and the estimate gives us
the slope (or the effect of body L on crown L in SP4)
#plot just to check and see what it looks like
plot(crown_length_mm ~ total_body_legnth_mm, data = SP4)
```

```
#trunk
```

```
#T1
```

```
lm_model_T1 <- lm(crown_length_mm ~ total_body_length_mm, data = T1)
summary(lm_model_T1) #We want to look at the Pr(>|t|) value for total
body length (this will give us significance) and the estimate gives us
the slope (or the effect of body L on crown L in T1)
cor(T1$total_body_length_mm, T1$crown_length_mm)
```

```
#T2
```

```
lm_model_T2 <- lm(crown_length_mm ~ total_body_length_mm, data = T2)
summary(lm_model_T2) #We want to look at the Pr(>|t|) value for total
body length (this will give us significance) and the estimate gives us
the slope (or the effect of body L on crown L in T2)
cor(T2$total_body_length_mm, T2$crown_length_mm)
```

```
#T3
```

```
lm_model_T3 <- lm(crown_length_mm ~ total_body_length_mm, data = T3)
summary(lm_model_T3) #We want to look at the Pr(>|t|) value for total
body length (this will give us significance) and the estimate gives us
the slope (or the effect of body L on crown L in T3)
cor(T3$total_body_length_mm, T3$crown_length_mm)
```

```
#B4 same thing as spiracle 4 here-- we only have branchial
measurements from one specimen so can't fit a regression
```

```
lm_model_B4 <- lm(crown_length_mm ~ total_body_length_mm, data = B4)
summary(lm_model_B4) #We want to look at the Pr(>|t|) value for total
body length (this will give us significance) and the estimate gives us
the slope (or the effect of body L on crown L in B4)
```

```
#combine all models into a string to have all in one place
```

```
lm_models <- c(lm_model_SN1, lm_model_SN2, lm_model_SN3, lm_model_SP1,
lm_model_SP2, lm_model_SP3, lm_model_SP4, lm_model_T1, lm_model_T2,
lm_model_T3, lm_model_B4)
```

```
#Stats Question 2: statistically, are snout ventral smaller than the
dorsal ones? Compare within morphotypes.
```

```
#separate ventral and dorsal denticles within each snout morphotype
```

```
SN1_d_pre <- subset(SN1, sample_number==1)
SN1_d <- SN1_d_pre$crown_length_mm #crown_length_mm
SN1_v_pre <- subset(SN1, sample_number==6)
SN1_v <- SN1_v_pre$crown_length_mm
```

```
SN2_d_pre <- subset(SN2, sample_number==1)
SN2_d <- SN2_d_pre$crown_length_mm
SN2_v_pre <- subset(SN2, sample_number==6)
SN2_v <- SN2_v_pre$crown_length_mm
```

```
SN3_d_pre <- subset(SN3, sample_number==1)
```

```

SN3_d <- SN3_d_pre$crown_length_mm
SN3_v_pre <- subset(SN3, sample_number==6)
SN3_v <- SN3_v_pre$crown_length_mm

#unpaired t test. Want to see if there is a significant difference in
means between ventral and dorsal size of the denticles within each
morphotype
#The null hypothesis (H0) states that there is no significant
difference between the means of the two groups.The alternative
hypothesis (H1) states that there is a significant difference between
the two population means, and that this difference is unlikely to be
caused by sampling error or chance.

##testing the assumptions for a t-test

#normality and variance
# Create density plots
plot(density(SN1_d), main = "Density Plot of SN1_d") #surface area
values spread for snout morphotype 1 (dorsal)
plot(density(SN1_v), main = "Density Plot of SN1_v")

plot(density(SN2_d), main = "Density Plot of SN2_d")
plot(density(SN2_v), main = "Density Plot of SN2_v")

plot(density(SN3_d), main = "Density Plot of SN3_d")
plot(density(SN3_v), main = "Density Plot of SN3_v")

# Create density plots SN1 together ventral and dorsal
plot(density(SN1_d), xlim = range(0:2), ylim = c(0, 7),
      col = "blue", xlab = "Surface Area", ylab = "Density", main =
"Density Plot of Surface Area SN1")
lines(density(SN1_v), col = "red")
# Add a legend
legend("topright", legend = c("Dorsal Snout", "Ventral Snout"), col =
c("blue", "red"), lty = 1)

# Create density plots SN2
plot(density(SN2_d),
      col = "blue", xlim = range(0:2), ylim=range(0:3), xlab =
"Surface Area", ylab = "Density", main = "Density Plot of Surface Area
SN2")
lines(density(SN2_v), col = "red")
# Add a legend
legend("topright", legend = c("Dorsal Snout", "Ventral Snout"), col =
c("blue", "red"), lty = 1)

# Create density plots SN3
plot(density(SN3_d),
      col = "blue", xlim = range(0:2), ylim=range(0:5), xlab =
"Surface Area", ylab = "Density", main = "Density Plot of Surface Area

```

```

SN3")
lines(density(SN3_v), col = "red")
# Add a legend
legend("topright", legend = c("Dorsal Snout", "Ventral Snout"), col =
c("blue", "red"), lty = 1)

# Shapiro-Wilk test
shapiro.test(SN1_d)
shapiro.test(SN1_v)
shapiro.test(SN2_d)
shapiro.test(SN2_v)
shapiro.test(SN3_d)
shapiro.test(SN3_v)
#these all look fine. For the statistical test, a value below 0.05 for
the p value would be worrisky. All values greater than 0.05 for p, so
data is normally distributed, and t-test assumptions are good.

#Snout type 1 (SN1)
t.test(SN1_d, SN1_v, paired=FALSE)

#Snout type 2 (SN2)
t.test(SN2_d, SN2_v, paired=FALSE)

#Snout type 3 (SN3)
t.test(SN3_d, SN3_v, paired=FALSE)

#### Scaling ####

#adding a column of data to denticles_clean that is the log of total
body length as we will use that to graph for all following
denticles_clean$log_body_length <-
log10(denticles_clean$total_body_length_mm)

#surface area

#calculate the log of the surface area
denticles_clean$log_sa <- log10(denticles_clean$surface_area_mm2)

#regression
sa_regression_equation <- lm(log_sa ~ log_body_length, data =
denticles_clean)
summary(sa_regression_equation)
slope_sa <- sa_regression_equation$coefficients[2]
intercept_sa <- sa_regression_equation$coefficients[1]
print(slope_sa) #check the slope

```

```

# Create the plot
ggplot(denticles_clean, aes(log_body_length, log_sa)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  ylim(-1.75, 1.3) +
  labs(x = "log( Body Length (mm) )", y = "log( Surface Area (mm2) )")
+
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_line(color = "lightgray"),
        panel.background = element_blank(),
        axis.text.x = element_text(size = 15),
        axis.text.y = element_text(size = 15),
        axis.title.x = element_text(size = 18, margin = margin(t =
10))), # Adjust the top margin
        axis.title.y = element_text(size = 18, margin = margin(r =
10)))

```

```

#volume
denticles_clean$log_v <- log10(denticles_clean$volume_mm3)

v_regression_equation <- lm(log_v ~ log_body_length, data =
denticles_clean)
summary(v_regression_equation)
slope_v <- v_regression_equation$coefficients[2]
intercept_v <- v_regression_equation$coefficients[1]
print(slope_v)

```

```

# Create the plot
ggplot(denticles_clean, aes(log_body_length, log_v)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  ylim(-1.75, 1.3) +
  labs(x = "log( Body Length (mm) )", y = "log( Volume (mm3) )") +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_line(color = "lightgray"),
        panel.background = element_blank(),
        axis.text.x = element_text(size = 15),
        axis.text.y = element_text(size = 15),
        axis.title.x = element_text(size = 18, margin = margin(t =
10))), # Adjust the top margin
        axis.title.y = element_text(size = 18, margin = margin(r =
10)))

```

```

#crown length
denticles_clean$log_crown_length <-

```

```

log10(denticles_clean$crown_length_mm)

cl_regression_equation <- lm(log_crown_length ~ log_body_length, data
= denticles_clean)
summary(cl_regression_equation)
slope_cl <- cl_regression_equation$coefficients[2]
intercept_cl <- cl_regression_equation$coefficients[1]
print(slope_cl)

# Create the plot
ggplot(denticles_clean, aes(log_body_length, log_crown_length)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  ylim(-1.75, 1.3) +
  labs(x = "log( Body Length (mm) )", y = "log( Crown Length (mm) )")
+
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_line(color = "lightgray"),
        panel.background = element_blank(),
        axis.text.x = element_text(size = 15),
        axis.text.y = element_text(size = 15),
        axis.title.x = element_text(size = 18, margin = margin(t =
10)), # Adjust the top margin
        axis.title.y = element_text(size = 18, margin = margin(r =
10)))

#crown width
denticles_clean$log_crown_width <-
log10(denticles_clean$crown_width_mm)

cw_regression_equation <- lm(log_crown_width ~ log_body_length, data =
denticles_clean)
summary(cw_regression_equation)
slope_cw <- cw_regression_equation$coefficients[2]
intercept_cw <- cw_regression_equation$coefficients[1]

# Create the plot
ggplot(denticles_clean, aes(log_body_length, log_crown_width)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  ylim(-1.75, 1.3) +
  labs(x = "log( Body Length (mm) )", y = "log( Crown Width (mm) )") +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_line(color = "lightgray"),
        panel.background = element_blank(),
        axis.text.x = element_text(size = 15),
        axis.text.y = element_text(size = 15),
        axis.title.x = element_text(size = 18, margin = margin(t =

```



```

10)), # Adjust the top margin
      axis.title.y = element_text(size = 18, margin = margin(r =
10)))

#base length
denticles_clean$log_base_length <-
log10(denticles_clean$base_length_mm)

bl_regression_equation <- lm(log_base_length ~ log_body_length, data =
denticles_clean)
summary(bl_regression_equation)
slope_bl <- bl_regression_equation$coefficients[2]
intercept_bl <- bl_regression_equation$coefficients[1]

# Create the plot
ggplot(denticles_clean, aes(log_body_length, log_base_length)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  ylim(-1.75, 1.3) +
  labs(x = "log( Body Length (mm) )", y = "log( Base Length (mm) )") +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_line(color = "lightgray"),
        panel.background = element_blank(),
        axis.text.x = element_text(size = 15),
        axis.text.y = element_text(size = 15),
        axis.title.x = element_text(size = 18, margin = margin(t =
10)), # Adjust the top margin
        axis.title.y = element_text(size = 18, margin = margin(r =
10)))

#base width
denticles_clean$log_base_width <- log10(denticles_clean$base_width_mm)

bw_regression_equation <- lm(log_base_width ~ log_body_length, data =
denticles_clean)
summary(bw_regression_equation)
slope_bw <- bw_regression_equation$coefficients[2]
intercept_bw <- bw_regression_equation$coefficients[1]

# Create the plot
ggplot(denticles_clean, aes(log_body_length, log_base_width)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  ylim(-1.75, 1.3) +
  labs(x = "log( Body Length (mm) )", y = "log( Base Width (mm) )") +
  theme(panel.grid.major = element_blank(),

```

```
    panel.grid.minor = element_line(color = "lightgray"),
    panel.background = element_blank(),
    axis.text.x = element_text(size = 15),
    axis.text.y = element_text(size = 15),
    axis.title.x = element_text(size = 18, margin = margin(t =
10)), # Adjust the top margin
    axis.title.y = element_text(size = 18, margin = margin(r =
10)))
```