```
#title: "Paleobiology of a large mammal community from the late Pleistocene of
Sonora, Mexico"
#author: "R.A. Short, L.G. Emmert, N.A. Famoso, J.M. Martin, J.I. Mead, S.L.
Swift, A. Baez"
#date: "March 2020"
#based on McHorse et al. 2016


  #This analysis is for identifying isolated equid phalanges found at Terapa,
Sonora, Mexico. We use linear discriminant analysis to determine the likely
species of the Terapa fossil horses, using a training set of identified
phalanges from contemporaneous faunas.

  #1. Setup and data cleaning
  #2. Assumption testing
  #3. Discriminant analysis for Paisley Caves phalanges
  #4. Stout vs. stilt setup, cleaning, and assumptions
  #5. Stout vs. stilt logistic regression

# Section 1: Setup and Data Cleaning
#First, set up the workspace and call relevant libraries.

rm(list=ls())
#setwd("filepath-to-data")
library(MASS)
library(plyr)
library(ggplot2)
library(GGally)
library(reshape2)
library(knitr)
library(dr)
library(cowplot)

# Bring in the data and check it out.
#rawtoes <- read.csv("supplementary_data_2.2.csv", header = T) #phalangies
rawtoes <- read.csv("combinedMCs2.csv", header = T) #metacarples
kable(head(rawtoes))
str(rawtoes)

# Do you want to save high-resolution figures for publication? Set as "yes" if
so.
plotflag <- "yes"


#Now, let's create a few levels of cleaned datasets to work with.

# Remove specimens with NA in any of the measurements
toes <- rawtoes[complete.cases(rawtoes[,10:15]),]

# Remove non-identified specimens
toescleaned <- toes[toes$Species.simplified != "sp.",]
toescleaned$Species.simplified <- droplevels(toescleaned$Species.simplified)
```

```
PChorses <- toescleaned[toescleaned$Species.simplified == "",] # Make a data
frame just for the Paisley Caves horses
squeakytoes <- toescleaned[toescleaned$Species.simplified != "",] # Remove the
Paisley Caves horses from the cleaned data frame.
squeakytoes$Species.simplified <- droplevels(squeakytoes$Species.simplified) #
Drop the empty levels.

#We now have four options for data: **rawtoes**, which is the original
dataset; **toes**, which has had incomplete specimens removed;
**toescleaned**, which has removed specimens without a species identification;
and **squeakytoes**, which is the cleaned dataset but also with the Paisley
Caves horses removed.

#To summarize our fully cleaned dataset:

# Summaries of locality and species sampling
kable(count(squeakytoes, "Locality"))
kable(count(squeakytoes, "Species.simplified"))
length(squeakytoes[,1]) - 2 # Total number of specimens sampled, excluding two
PC specimens

# Section 2: Assumption testing
#The main assumptions of discriminant analysis include normally distributed
data, equivalent covariance matrices, and independence of data points. The
data are independently sampled by default except for the possibility of
occasionally including two phalanges from the same animal, but we will test
the other assumptions quantitatively.

#We begin by testing normality.

# Within each species, is each measurement normally distributed? Use the
Shapiro test, then print the p-value, species, and column.
# p < 0.05 suggests non-normality.
# Using 'squeakytoes' data because singletons cannot be Shapiro tested.

toes.shap <- ddply(squeakytoes, "Species.simplified", summarize,
                      ddp = shapiro.test(squeakytoes$DD)$p.value,
                      bdp = shapiro.test(squeakytoes$Bd)$p.value,
                      ea10p = shapiro.test(squeakytoes$EAdGS10)$p.value,
                      ea12p =  shapiro.test(squeakytoes$EAdGS12)$p.value,
                      ea13p = shapiro.test(squeakytoes$EAdGS13)$p.value,
                      ea14p =  shapiro.test(squeakytoes$EAdGS14)$p.value)

# Round the output to three digits
toes.shap[1:nrow(toes.shap),2:ncol(toes.shap)] <-
round(toes.shap[1:nrow(toes.shap),2:ncol(toes.shap)], digits=3)

# Show the table.
kable(toes.shap)


#The measurements are generally normally distributed.

#Now let's test multivariate normality.
```

```
# Are all measurements normally distributed in multivariate space?
# QQ plot - code from http://www.statmethods.net/stats/anovaAssumptions.html
m.toes <- as.matrix(log(squeakytoes[,10:15])) # n x p numeric matrix
center <- colMeans(m.toes) # centroid
n <- nrow(m.toes); p <- ncol(m.toes); cov <- cov(m.toes);
d <- mahalanobis(m.toes, center, cov) # distances
qqplot(qchisq(ppoints(n),df=p),d,
       main="QQ Plot Assessing Multivariate Normality",
       ylab="Mahalanobis D2", xlab="Normal Quantiles")
abline(a=0,b=1, col="darkorchid")

#The log-transformed data appear to be fairly multivariately normally
distributed, with a few outliers; so, let's log-transform our datasheets.

rawtoes[,10:15] <- log(rawtoes[,10:15])
toes[,10:15] <- log(toes[,10:15])
toescleaned[,10:15] <- log(toescleaned[,10:15])
squeakytoes[,10:15] <- log(squeakytoes[,10:15])
PChorses[,10:15] <- log(PChorses[,10:15])


#If you'd like, you can go back and perform the univariate Shapiro tests again
to make sure that log-transforming the data has not somehow made any of them
significantly deviate from normal. (It has not, in this case.)

#Next, we test equivalence of variance.

# Bartlett Test of Homogeneity of Variances
bartlett.toes <- c(ddp = bartlett.test(DD ~ Species.simplified, data =
squeakytoes)$p.value,
                   bdp = bartlett.test(Bd ~ Species.simplified, data =
squeakytoes)$p.value,
                   ea10p = bartlett.test(EAdGS10 ~ Species.simplified, data =
squeakytoes)$p.value,
                   ea12p = bartlett.test(EAdGS12 ~ Species.simplified, data =
squeakytoes)$p.value,
                   ea13p = bartlett.test(EAdGS13 ~ Species.simplified, data =
squeakytoes)$p.value,
                   ea14p = bartlett.test(EAdGS14 ~ Species.simplified, data =
squeakytoes)$p.value)
print(round(bartlett.toes, digits=3))

#For some variables, the hypothesis of equal variance is rejected by the
Bartlett test (where p $< 0.05$). We will therefore use quadratic discriminant
analysis (QDA) rather than linear discriminant analysis (LDA), as the former
does not make the assumption of equal variance/covariance.



# Section 3: Discriminant analysis
```

```
#We will perform a jackknifed QDA, which uses leave-one-out cross-validation
to report a more accurate identification accuracy. We then use the
discriminant function to predict the identity of the Paisley Caves horses and
return the prior probabilities of those predictions.

#QDA, cross-validated, even priors, on **squeakytoes**:

# Does not predict new specimens.
qCVfit <- qda(Species.simplified ~ DD + Bd + EAdGS10 + EAdGS12 + EAdGS13 +
EAdGS14,
              data=squeakytoes,
              prior = seq(from=1, to=1,
length.out=length(levels(squeakytoes$Species.simplified)))/
length(levels(squeakytoes$Species.simplified)),
              CV = TRUE)
qCVtab <- table(squeakytoes$Species.simplified, qCVfit$class)
kable(qCVtab) # Matrix of actual vs. predicted IDs
print(diag(round(prop.table(qCVtab, 1), digits=3))) # Proportion correct for
each species
print(sum(round((diag(prop.table(qCVtab))), digits=3))) # Total proportion
correct

#QDA, not cross-validated, even priors, to predict identity of Paisley Caves
phalanges:

# Now we use non-jackknifed LDA to predict the identity of the PC horses.
qfit <- qda(Species.simplified ~ DD + Bd + EAdGS10 + EAdGS12 + EAdGS13 +
EAdGS14,
            data=squeakytoes,
            prior = seq(from=1, to=1,
length.out=length(levels(squeakytoes$Species.simplified)))/
length(levels(squeakytoes$Species.simplified)),
            CV = FALSE)
qpnew <- predict(qfit, PChorses) # Predict the Paisley Caves horses...
qpostprob <- qpnew$posterior # Assign the posterior probabilities to an object
kable(round(qpostprob, digits = 4)) # Round for easier viewing.


#Finally, a plot to visualize the discriminant space. We'll be using SAVE
variates, which offer visualization for QDA analogous to plotting the first
two canonical axes of LDA.


qdaplot <- rbind(squeakytoes, PChorses) # Add the PC horses back in, so we can
see them on the plot.

qda.save <- dr(Species.simplified ~ DD + Bd + EAdGS10 + EAdGS12 + EAdGS13 +
EAdGS14,
               data=qdaplot,
               na.action=na.omit,
               method = "save")

# Save the first and second SAVE variates and labels. Label the PC horses as
Unknown.
```

```
variates <- dr.direction(qda.save, which = 1:2)
qy <- dr.y(qda.save)
qy.labels <- as.character(qy)
qy.labels[qy.labels == ""] <- "Unknown"

plotdata <- as.data.frame(cbind(qy, variates))

# Note that for the paper, colors but no positions have been changed.
plotpalette <- c("gray63", "darkgoldenrod2", "steelblue4", "darkmagenta",
"darkgreen", "red", "blue","purple", "green", "pink")

ggplot(plotdata, aes(Dir1,Dir2)) +
  geom_point(aes(color=qy.labels, shape=qy.labels), size=3) +
  scale_shape_manual(name = "Species", values = c(17, 18, 1, 15, 3, 8, 2, 4,
5, 6)) +
  scale_color_manual(name = "Species", values=plotpalette) +
  theme(legend.position = "right",
        legend.background = element_rect(fill="white", size=0.5,
linetype="solid", colour ="black")) +
  labs(title = "SAVE variates for QDA", x = "SAVE Variate 1", y = "SAVE
Variate 2", color = "Species")


# Section 4: Stout vs. stilt setup, cleaning, and assumptions
#We will create a **SStoes** dataset, which has non-identifications #removed.
We'll make a second object, **SStoesPC**, that includes the #Paisley Caves
specimens; finally, we'll create separate **stout** and **stilt** subsets.


# Set up data
SStoes <- toescleaned[toescleaned$Stout.Stilt != "",] # Remove specimens not
identified as stout- or stilt-legged
SStoesPC <- rbind(SStoes, PChorses) # Add the Paisley Caves specimens

stout <- SStoes[SStoes$Stout.Stilt == "Stout",]
stilt <- SStoes[SStoes$Stout.Stilt == "Stilt",]


# Within each group, is each measurement normally distributed? Use the Shapiro
test, then print the p-value, species, and column.
# p < 0.05 suggests non-normality.

SS.shap <- ddply(SStoes, "Stout.Stilt", summarize,
 glp = shapiro.test(GL)$p.value,
 bpp = shapiro.test(Bp)$p.value,
  bfpp = shapiro.test(BFp)$p.value,
  sdp = shapiro.test(SD)$p.value,
   bdp = shapiro.test(Bd)$p.value,
   mlbp = shapiro.test(MinLB)$p.value)

kable(SS.shap)
```

#The stilt-legged horses are normally distributed, but the stout-legged horses are not. We can look at the distribution of measurements to get an idea why. This section formats the data and then plots the distribution of each measurement.

```
# Make a new frame of stout measurements only
stoutmeasures <- stout[,10:15]
stoutplot <- cbind(stout$Species.simplified, stoutmeasures)
stoutplot <- melt(stoutplot)

# Plot the stouts
ggplot(stoutplot, aes(x=value, fill=variable)) + geom_density(alpha=.3) +
 facet_grid(variable ~ .) +
 theme(panel.background = element_blank(), panel.grid.minor = element_blank())
```

#It appears that most of the variables are bimodally distributed for the stout-legged horses, and this is why the normality assumption fails. While DFA is generally suggested to be robust to violations of normality, we will use a logistic regression instead. Logistic regression performs a similar function but makes fewer assumptions.

```
# Section 5: Stout vs. stilt
#We will now perform a logistic regression using **toescleaned**, using
#measurements to determine stout- vs. stilt-leggedness. First, though, #look
at the correlations of variables:


kable(cor(SStoes[,10:15])) # Look at pairwise correlations among variables
```

#The variables are all quite tightly correlated, violating assumptions of logistic regression. To solve this problem, we will use principal components to collapse the variation into orthogonal axes and then perform logistic regression on the first two PC axes; in this way, we combine the information from all 8 dimensions without violating assumptions.

```
SSpca <- princomp(SStoesPC[,10:15], cor=TRUE)
summary(SSpca) # How much variance included in each principal component?
PCs <- SSpca$scores # Save scores

SStoesPC <- cbind(SStoesPC, PCs) # Add the principal component scores to the
data frame

# Visualize the first two principal components
ggpairs(SStoesPC[SStoesPC$Stout.Stilt != "",], # Remove the PC horses so they
don't throw off the colors
     columns=c("Comp.1", "Comp.2"),
   colour='Stout.Stilt',
   title="Stout and Stilt Principal Components",
   lower=list(continuous='points'),
   axisLabels='none',
```

```
   upper=list(continuous='blank'),
   legends=T)

# Visualize the first and third principal components
ggpairs(SStoesPC[SStoesPC$Stout.Stilt != "",], # Remove the PC horses so they
don't throw off the colors
      columns=c("Comp.1", "Comp.3"),
     colour='Stout.Stilt',
      title="Stout and Stilt Principal Components",
    lower=list(continuous='points'),
   axisLabels='none',
  upper=list(continuous='blank'),
 legends=T)

# Visualize the second and third principal components
ggpairs(SStoesPC[SStoesPC$Stout.Stilt != "",], # Remove the PC horses so they
don't throw off the colors
 columns=c("Comp.2", "Comp.3"),
  colour='Stout.Stilt',
   title="Stout and Stilt Principal Components",
    lower=list(continuous='points'),
     axisLabels='none',
      upper=list(continuous='blank'),
       legends=T)


#We now perform the logistic regression.


# First we split apart the Tarapa unknowns and the training set again.
SSnoPC <- SStoesPC[SStoesPC$Repository != "ETSU",]
PCpc <- SStoesPC[SStoesPC$Repository == "ETSU",]

# Now we fit the logistic, which is a specific form of the generalized linear
model command.
SSfit <- glm(Stout.Stilt ~ Comp.1 + Comp.2 + Comp.3 + Comp.4 + Comp.5 + Comp.
6, data = SSnoPC, family = binomial())

summary(SSfit) # Summarize the logistic model

SSnoPC$IsStout <- SSnoPC$Stout.Stilt == "Stout"
SSpredict <- predict(SSfit, SSnoPC, type = "response") # Use the logistic
model to predict on the training set.
SSresults <- table(actual.stout = SSnoPC$IsStout, predicted.stout = #SSpredict
> 0.5)
#SSresults # Confusion matrix. "False" vs. "True" indicates whether or not the
specimen is (or is predicted to be) stout-legged.
sum(diag(prop.table(SSresults))) # Shows total percent correct.

# Next, we predict the identity of the Paisley Caves phalanges.
PCpredict <- predict(SSfit, PCpc, type = "response")
PCpredicts <- PCpredict > 0.5 # Where TRUE predicts stout-legged and FALSE
suggests stilt-legged.
PCpredicts
```

```
PCpredict <- round(PCpredict, digits = 3) # Round strength of predictions
PCpredict # Gives probability of "TRUE" for each specimen, i.e., probability
of being stout-legged.


#The stout vs. stilt logistic predicts both Paisley Caves specimens as stout-
legged. See paper for discussion; here we plot the principal components of
each species, as well as the unknown Paisley Caves horses (red dots), to
explore why the logistic makes these predictions.


# We'll add the PC horses back in and label their species and stout/stilt
status as Unknown.
SStoesPC$Species.simplified <- as.character(SStoesPC$Species.simplified)
SStoesPC[SStoesPC$Species.simplified == "",7] <- "Unknown"
SStoesPC$Species.simplified <- as.factor(SStoesPC$Species.simplified)

SStoesPC$Stout.Stilt <- as.character(SStoesPC$Stout.Stilt)
SStoesPC[SStoesPC$Stout.Stilt == "",4] <- "Unknown"
SStoesPC$Stout.Stilt <- as.factor(SStoesPC$Stout.Stilt)

# Visualize the first two principal components

SSplot <- cbind(SStoesPC[,16:23])
SSplot$Stout.Stilt <- SStoesPC$Stout.Stilt
SSplot$Species <- SStoesPC$Species.simplified

ggplot(SSplot, aes(Comp.1, Comp.2)) +
 geom_point(aes(color=Species, shape=Stout.Stilt), size=4) +
scale_shape_manual(values = c(0, 6, 16)) +
scale_color_manual(name = "Species", values=plotpalette) +
theme(legend.position = "right",
     legend.background = element_rect(fill="white", size=0.5,
linetype="solid", colour ="black")) +
labs(x = "Principal Component 1", y = "Principal Component 2")


# Run this section if you wish to save plots for the paper.
if(plotflag == "yes") {
  postscript("Fig2A.eps", width = 7, height = 4.6, horizontal = FALSE,
             onefile = FALSE, paper = "special", colormodel = "cmyk")
}
ggplot(plotdata, aes(Dir1,Dir2)) +
  geom_point(aes(color=qy.labels, shape=qy.labels), size=4) +
  scale_shape_manual(name = "Species", labels = unique(qy.labels), values =
c(17, 18, 1, 15, 3, 8)) +
  scale_color_manual(name = "Species", labels = unique(qy.labels),
values=plotpalette) +
  theme(legend.position = "right",
        legend.background = element_rect(fill="white", size=0.5,
linetype="solid", colour ="black")) +
  labs(x = "SAVE Variate 1", y = "SAVE Variate 2", color = "Species")
```

```
if(plotflag == "yes") {
  dev.off()
}


if(plotflag == "yes") {
  postscript("Fig2B.eps", width = 7, height = 4.6, horizontal = FALSE,
             onefile = FALSE, paper = "special", colormodel = "cmyk")
}
ggplot(SSplot, aes(Comp.1, Comp.2)) +
  geom_point(aes(color=Species, shape=Stout.Stilt), size=4) +
  scale_shape_manual(values = c(0, 6, 16)) +
  scale_color_manual(values=plotpalette) +
  theme(legend.position = "right",
        legend.background = element_rect(fill="white", size=0.5,
linetype="solid", colour ="black")) +
  labs(x = "Principal Component 1", y = "Principal Component 2")

if(plotflag == "yes") {
  dev.off()
}



# References
#R Core Team. 2014. R: A Language and Environment for Statistical Computing
(version $3.1.0$). Vienna: R Foundation for Statistical Computing.

#Schloerke, B., J. Crowley, D. Cook, H. Hofmann, H. Wickham, F. Briatte, and
M. Marbach. 2011. Ggally: Extension to ggplot2.

#Venables, W. N., and B. D. Ripley. 2002. Modern Applied Statistics with S.
Springer-Verlag.

#Weisberg, S. 2002. Dimension reduction regression in R. Journal of
Statistical Software 7:1-22.

#Wickham, H. 2007. Reshaping data with the reshape package. Journal of
Statistical Software 21:1-20.

#Wickham, H. 2009. ggplot2: Elegant Graphics for Data Analysis. Springer, 221
pp.

#Wickham, H. 2011. The split-apply-combine strategy for data analysis. Journal
of Statistical Software 40:1-29.

#Xie, M. Y. 2015. Knitr: A General-Purpose Package for Dynamic Report
Generation in R.
```