Subsistence_Comparison

Dylan S. Davis

2023-08-26

R Markdown

This is an R Markdown document containing code used to conduct the analyses reported in the manuscript. This analysis was originally created in R v.4.0.2 (R Core Team 2020). PlanetScope imagery used for this analysis is proprietary (c) Planet Inc. 2022. Archaeological datapoints can be shared upon reasonable request, but are not provided to protect their geographic coordinates.

Load libraries and datasets

```
library(raster)
library(sf)
library(ggplot2)


#set working directory
setwd("C:/Users/dylan/Documents/School_Work/Dissertation/SPARC_Namonte")

#load archaeological point file
arch_points <- as(st_read("Mikea_Survey_2018.shp"), "Spatial")

#load ground-tested non-archaeological point locations


non_arch_points <- as(st_read("Non_Arch_Points.shp"), "Spatial")

##Foraging Site Analysis

#set working directory
setwd("C:/Users/dylan/Documents/School_Work/Dissertation/GEE_test/
Planet_test")

V_arch_points <- as(st_read("Arch_total_pts.shp"), "Spatial")

V_non_arch_points <- as(st_read("non_arch_pts.shp"), "Spatial")

##Mixed subsistence analysis
#set working directory
setwd("C:/Users/dylan/Documents/School_Work/Madagascar Projects/NCT_Project")
#replace with the full path to file that you are working with

F_arch_points <- as(st_read("Foraging_Sites_19_20.shp"), "Spatial")
```

```r
#primarily foraging

M_arch_points <- as(st_read("Mixed_Sites_v2.shp"), "Spatial") #mixed
subsistence

#Create Median Averaged Images for Multitemporal Analysis

#load each band of imagery from different years


J18_B1 <- raster("June18mrg.tif", band = 1)

J19_B1 <- raster("June19mrg.tif", band = 1)

J20_B1 <- raster("June20mrg.tif", band = 1)

J21_B1 <- raster("June21mrg.tif", band = 1)


J_B1 <- raster::stack(J18_B1, J19_B1, J20_B1, J21_B1)

J_B1_Med <- raster::calc(J_B1, median) #takes a long time depending on your
computer


J18_B2 <- raster("June18mrg.tif", band = 2)

J19_B2 <- raster("June19mrg.tif", band = 2)

J20_B2 <- raster("June20mrg.tif", band = 2)

J21_B2 <- raster("June21mrg.tif", band = 2)


J_B2 <- raster::stack(J18_B2, J19_B2, J20_B2, J21_B2)

J_B2_Med <- raster::calc(J_B2, median)


J18_B3 <- raster("June18mrg.tif", band = 3)

J19_B3 <- raster("June19mrg.tif", band = 3)

J20_B3 <- raster("June20mrg.tif", band = 3)

J21_B3 <- raster("June21mrg.tif", band = 3)


J_B3 <- raster::stack(J18_B3, J19_B3, J20_B3, J21_B3)
```

```r
J_B3_Med <- raster::calc(J_B3, median)


J18_B4 <- raster("June18mrg.tif", band = 4)

J19_B4 <- raster("June19mrg.tif", band = 4)

J20_B4 <- raster("June20mrg.tif", band = 4)

J21_B4 <- raster("June21mrg.tif", band = 4)


J_B4 <- raster::stack(J18_B4, J19_B4, J20_B4, J21_B4)

J_B4_Med <- raster::calc(J_B4, median)


J_med_comp <- raster::stack(J_B1_Med, J_B2_Med, J_B3_Med, J_B4_Med) #create a raster image stack of all the individual bands

writeRaster(J_med_comp, "June_Med.tiff")


F18_B1 <- raster("Feb18mrg.tif", band = 1)

F19_B1 <- raster("Feb19mrg.tif", band = 1)

F20_B1 <- raster("Feb20mrg.tif", band = 1)

F21_B1 <- raster("Feb21mrg.tif", band = 1)


F_B1 <- raster::stack(F18_B1, F19_B1, F20_B1, F21_B1)

F_B1_Med <- raster::calc(F_B1, median)




F18_B2 <- raster("Feb18mrg.tif", band = 2)

F19_B2 <- raster("Feb19mrg.tif", band = 2)

F20_B2 <- raster("Feb20mrg.tif", band = 2)

F21_B2 <- raster("Feb21mrg.tif", band = 2)


F_B2 <- raster::stack(F18_B2, F19_B2, F20_B2, F21_B2)
```

```r
F_B2_Med <- raster::calc(F_B2, median)


F18_B3 <- raster("Feb18mrg.tif", band = 3)

F19_B3 <- raster("Feb19mrg.tif", band = 3)

F20_B3 <- raster("Feb20mrg.tif", band = 3)

F21_B3 <- raster("Feb21mrg.tif", band = 3)



F_B3 <- raster::stack(F18_B3, F19_B3, F20_B3, F21_B3)

F_B3_Med <- raster::calc(F_B3, median)


F18_B4 <- raster("Feb18mrg.tif", band = 4)

F19_B4 <- raster("Feb19mrg.tif", band = 4)

F20_B4 <- raster("Feb20mrg.tif", band = 4)

F21_B4 <- raster("Feb21mrg.tif", band = 4)


F_B4 <- raster::stack(F18_B4, F19_B4, F20_B4, F21_B4)

F_B4_Med <- raster::calc(F_B4, median)


F_med_comp <- raster::stack(F_B1_Med, F_B2_Med, F_B3_Med, F_B4_Med)

writeRaster(F_med_comp, "Feb_Med.tiff")


#Load median composites that we just generated

J_B1 <- raster("June_Med.tif", band = 1)
J_B2 <- raster("June_Med.tif", band = 2)
J_B3 <- raster("June_Med.tif", band = 3)
J_B4 <- raster("June_Med.tif", band = 4)

DS_Comp <- raster::stack(J_B1, J_B2, J_B3, J_B4)

F_B1 <- raster("Feb_Med.tif", band = 1)
F_B2 <- raster("Feb_Med.tif", band = 2)
```

```
F_B3 <- raster("Feb_Med.tif", band = 3)
F_B4 <- raster("Feb_Med.tif", band = 4)

WS_Comp <- raster::stack(F_B1, F_B2, F_B3, F_B4)

#Create NDVI
NDVI_Feb = (WS_Comp[[4]] - WS_Comp[[3]])/(WS_Comp[[4]] + WS_Comp[[3]])

NDVI_Jun = (DS_Comp[[4]] - DS_Comp[[3]])/(DS_Comp[[4]] + DS_Comp[[3]])
```

Spectral Analysis for Wet and Dry Seasons

```
#Extract raster values at each archaeological point location
#Depending on the size of the study region and processing power
#of your computer, this can take a while to run
N_arch_DS=extract(DS_Comp, arch_points, buffer=20, fun=mean)

N_Narc_DS=extract(DS_Comp, non_arch_points, buffer=20, fun=mean)

V_arch_DS=extract(DS_Comp, V_arch_points, buffer=20, fun=mean)
V_Narc_DS=extract(DS_Comp, V_non_arch_points, buffer=20, fun=mean)


#Subset analysis of known foraging, pastoralist, and mixed subsistence sites
F_arch_DS=extract(DS_Comp, F_arch_points, buffer=20, fun=mean)

F_arch_DS <- na.omit(F_arch_DS)
M_arch_DS=extract(DS_Comp, M_arch_points, buffer=20, fun=mean)

M_arch_DS <- na.omit(M_arch_DS)

#Wet Season
N_arch_WS=extract(WS_Comp, arch_points, buffer=20, fun=mean)

N_Narc_WS=extract(WS_Comp, non_arch_points, buffer=20, fun=mean)

V_arch_WS=extract(WS_Comp, V_arch_points, buffer=20, fun=mean)
V_Narc_WS=extract(WS_Comp, V_non_arch_points, buffer=20, fun=mean)

F_arch_WS=extract(WS_Comp, F_arch_points, buffer=20, fun=mean)

M_arch_WS=extract(WS_Comp, M_arch_points, buffer=20, fun=mean)

#Standardize data values for comparison of data from different regions

standard <- function(x, y) {                              # Create user-
defined function
  min(y) + ((x - min(x))*(max(y)-min(y))) / (max(x) - min(x))
}
```

```r
N_arch_DS_s <- standard(N_arch_DS, N_Narc_DS)
N_Narc_DS_s <- standard(N_Narc_DS, N_Narc_DS)
V_arch_DS_s <- standard(V_arch_DS, N_Narc_DS) #already standardized against
itself
V_Narc_DS_s <- standard(V_Narc_DS, V_Narc_DS) #don't standardize

F_arch_DS_s <- standard(F_arch_DS, N_Narc_DS)
M_arch_DS_s <- standard(M_arch_DS, N_Narc_DS)

#make data frames from each raster point table for Dry Season
DS_arc_df <- data.frame(N_arch_DS_s)
DS_Narc_df <- data.frame(N_Narc_DS_s)

DS_v_arc_df <- data.frame(V_arch_DS_s)
DS_v_Narc_df <- data.frame(V_Narc_DS_s)
#assess possibility of mixed subsistence on results
DS_m_arc_df <- data.frame(M_arch_DS_s) #mixed sites (contains domesticate
animal bones and fishing/foraging evidence)
DS_f_arc_df <- data.frame(F_arch_DS_s) #foraging/fishing only (no animal
bones)

#add new column named "cat" (for category)
DS_arc_df$cat <- "Inland Pastoralist" #archaeological points
DS_Narc_df$cat <- "Non-Arc-Mikea" #non-archaeological points
DS_v_arc_df$cat <- "Coastal Communities" #archaeological points
DS_v_Narc_df$cat <- "Non-Arc-Velo" #non-archaeological points

DS_f_arc_df$cat <- "Fishing/Foraging (only)" #archaeological points
DS_m_arc_df$cat <- "Mixed Subsistence"

#combine both dataframes into a single dataframe
DS_df <- rbind(DS_arc_df, DS_v_arc_df, DS_f_arc_df, DS_m_arc_df)

#standardize values for wet season
N_arch_WS_s <- standard(N_arch_WS, N_Narc_WS)
N_Narc_WS_s <- standard(N_Narc_WS, N_Narc_WS)
V_arch_WS_s <- standard(V_arch_WS, N_Narc_WS) #already standardized against
itself
V_Narc_WS_s <- standard(V_Narc_WS, V_Narc_WS) #don't standardize

F_arch_WS_s <- standard(F_arch_WS, N_Narc_WS)
M_arch_WS_s <- standard(M_arch_WS, N_Narc_WS)

#make data frames from each raster point table for Wet Season
WS_arc_df <- data.frame(N_arch_WS)
WS_Narc_df <- data.frame(N_Narc_WS)

WS_v_arc_df <- data.frame(V_arch_WS)
WS_v_Narc_df <- data.frame(V_Narc_WS)
```

```r
#assess possibility of mixed subsistence on results
WS_m_arc_df <- data.frame(M_arch_WS)
WS_f_arc_df <- data.frame(F_arch_WS)

#add new column named "cat" (for category)
WS_arc_df$cat <- "Inland Pastoralist" #archaeological points
WS_Narc_df$cat <- "Non-Arc-Mikea" #non-archaeological points
WS_v_arc_df$cat <- "Coastal Communities" #archaeological points
WS_v_Narc_df$cat <- "Non-Arc-Velo" #non-archaeological points

WS_f_arc_df$cat <- "Fishing/Foraging (only)" #archaeological points
WS_m_arc_df$cat <- "Mixed Subsistence"

#combine both dataframes into a single dataframe
WS_df <- rbind(WS_arc_df, WS_v_arc_df, WS_f_arc_df, WS_m_arc_df)


#Calculate Separability Metric (M)

M_b1 <- abs((mean(WS_arc_df$Feb_Med.1)-
mean(WS_v_arc_df$Feb_Med.1))/(sd(WS_arc_df$Feb_Med.1)+sd(WS_v_arc_df$Feb_Med.
1)))
M_b2 <- abs((mean(WS_arc_df$Feb_Med.2)-
mean(WS_v_arc_df$Feb_Med.2))/(sd(WS_arc_df$Feb_Med.2)+sd(WS_v_arc_df$Feb_Med.
2)))
M_b3 <- abs((mean(WS_arc_df$Feb_Med.3)-
mean(WS_v_arc_df$Feb_Med.3))/(sd(WS_arc_df$Feb_Med.3)+sd(WS_v_arc_df$Feb_Med.
3)))
M_b4 <- abs((mean(WS_arc_df$Feb_Med.4)-
mean(WS_v_arc_df$Feb_Med.4))/(sd(WS_arc_df$Feb_Med.4)+sd(WS_v_arc_df$Feb_Med.
4)))


M_b1

## [1] 0.4903444

M_b2

## [1] 0.2621236

M_b3

## [1] 0.08895461

M_b4

## [1] 0.2978269


wilcox.test(WS_arc_df$Feb_Med.1, WS_v_arc_df$Feb_Med.1)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_arc_df$Feb_Med.1 and WS_v_arc_df$Feb_Med.1
## W = 328, p-value = 2.77e-09
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(WS_arc_df$Feb_Med.2, WS_v_arc_df$Feb_Med.2)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_arc_df$Feb_Med.2 and WS_v_arc_df$Feb_Med.2
## W = 498, p-value = 3.349e-08
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(WS_arc_df$Feb_Med.3, WS_v_arc_df$Feb_Med.3)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_arc_df$Feb_Med.3 and WS_v_arc_df$Feb_Med.3
## W = 887, p-value = 5.235e-06
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(WS_arc_df$Feb_Med.4, WS_v_arc_df$Feb_Med.4)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_arc_df$Feb_Med.4 and WS_v_arc_df$Feb_Med.4
## W = 1080, p-value = 4.593e-05
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(WS_arc_df$Feb_Med.1, WS_f_arc_df$Feb_Med.1)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_arc_df$Feb_Med.1 and WS_f_arc_df$Feb_Med.1
## W = 228, p-value = 1.123e-07
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(WS_arc_df$Feb_Med.2, WS_f_arc_df$Feb_Med.2)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_arc_df$Feb_Med.2 and WS_f_arc_df$Feb_Med.2
## W = 464, p-value = 6.174e-05
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(WS_arc_df$Feb_Med.3, WS_f_arc_df$Feb_Med.3)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_arc_df$Feb_Med.3 and WS_f_arc_df$Feb_Med.3
## W = 876, p-value = 0.08233
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(WS_arc_df$Feb_Med.4, WS_f_arc_df$Feb_Med.4)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_arc_df$Feb_Med.4 and WS_f_arc_df$Feb_Med.4
## W = 1904, p-value = 8.931e-05
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(WS_arc_df$Feb_Med.1, WS_m_arc_df$Feb_Med.1)

##
##  Wilcoxon rank sum exact test
##
## data:  WS_arc_df$Feb_Med.1 and WS_m_arc_df$Feb_Med.1
## W = 7, p-value = 9.077e-11
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(WS_arc_df$Feb_Med.2, WS_m_arc_df$Feb_Med.2)

##
##  Wilcoxon rank sum exact test
##
## data:  WS_arc_df$Feb_Med.2 and WS_m_arc_df$Feb_Med.2
## W = 14, p-value = 1.025e-09
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(WS_arc_df$Feb_Med.3, WS_m_arc_df$Feb_Med.3)

##
##  Wilcoxon rank sum exact test
##
## data:  WS_arc_df$Feb_Med.3 and WS_m_arc_df$Feb_Med.3
## W = 251, p-value = 0.8107
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(WS_arc_df$Feb_Med.4, WS_m_arc_df$Feb_Med.4)

##
##  Wilcoxon rank sum exact test
##
## data:  WS_arc_df$Feb_Med.4 and WS_m_arc_df$Feb_Med.4
```

```
## W = 455, p-value = 1.837e-08
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(WS_f_arc_df$Feb_Med.1, WS_m_arc_df$Feb_Med.1)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_f_arc_df$Feb_Med.1 and WS_m_arc_df$Feb_Med.1
## W = 1117, p-value = 1.591e-05
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(WS_f_arc_df$Feb_Med.2, WS_m_arc_df$Feb_Med.2)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_f_arc_df$Feb_Med.2 and WS_m_arc_df$Feb_Med.2
## W = 552, p-value = 8.159e-11
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(WS_f_arc_df$Feb_Med.3, WS_m_arc_df$Feb_Med.3)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_f_arc_df$Feb_Med.3 and WS_m_arc_df$Feb_Med.3
## W = 2860, p-value = 0.01588
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(WS_f_arc_df$Feb_Med.4, WS_m_arc_df$Feb_Med.4)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  WS_f_arc_df$Feb_Med.4 and WS_m_arc_df$Feb_Med.4
## W = 1856, p-value = 0.1438
## alternative hypothesis: true location shift is not equal to 0
```

```
M_b1 <- abs((mean(DS_arc_df$June_Med.1)-
mean(DS_v_arc_df$June_Med.1))/(sd(DS_arc_df$June_Med.1)+sd(DS_v_arc_df$June_M
ed.1)))
M_b2 <- abs((mean(DS_arc_df$June_Med.2)-
mean(DS_v_arc_df$June_Med.2))/(sd(DS_arc_df$June_Med.2)+sd(DS_v_arc_df$June_M
ed.2)))
M_b3 <- abs((mean(DS_arc_df$June_Med.3)-
mean(DS_v_arc_df$June_Med.3))/(sd(DS_arc_df$June_Med.3)+sd(DS_v_arc_df$June_M
ed.3)))
M_b4 <- abs((mean(DS_arc_df$June_Med.4)-
mean(DS_v_arc_df$June_Med.4))/(sd(DS_arc_df$June_Med.4)+sd(DS_v_arc_df$June_M
ed.4)))
```

```
wilcox.test(DS_arc_df$June_Med.1, DS_v_arc_df$June_Med.1)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  DS_arc_df$June_Med.1 and DS_v_arc_df$June_Med.1
## W = 1338.5, p-value = 0.0005973
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_arc_df$June_Med.2, DS_v_arc_df$June_Med.2)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  DS_arc_df$June_Med.2 and DS_v_arc_df$June_Med.2
## W = 1957, p-value = 0.05804
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_arc_df$June_Med.3, DS_v_arc_df$June_Med.3)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  DS_arc_df$June_Med.3 and DS_v_arc_df$June_Med.3
## W = 2551, p-value = 0.6753
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_arc_df$June_Med.4, DS_v_arc_df$June_Med.4)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  DS_arc_df$June_Med.4 and DS_v_arc_df$June_Med.4
## W = 3648, p-value = 0.02114
## alternative hypothesis: true location shift is not equal to 0

M_b1 <- abs((mean(DS_arc_df$June_Med.1)-
mean(DS_f_arc_df$June_Med.1))/(sd(DS_arc_df$June_Med.1)+sd(DS_f_arc_df$June_M
ed.1)))
M_b2 <- abs((mean(DS_arc_df$June_Med.2)-
mean(DS_f_arc_df$June_Med.2))/(sd(DS_arc_df$June_Med.2)+sd(DS_f_arc_df$June_M
ed.2)))
M_b3 <- abs((mean(DS_arc_df$June_Med.3)-
mean(DS_f_arc_df$June_Med.3))/(sd(DS_arc_df$June_Med.3)+sd(DS_f_arc_df$June_M
ed.3)))
M_b4 <- abs((mean(DS_arc_df$June_Med.4)-
mean(DS_f_arc_df$June_Med.4))/(sd(DS_arc_df$June_Med.4)+sd(DS_f_arc_df$June_M
ed.4)))

M_b1

## [1] 0.6978894
```

```
M_b2

## [1] 0.4217028

M_b3

## [1] 0.09401167

M_b4

## [1] 1.064309

wilcox.test(DS_arc_df$June_Med.1, DS_f_arc_df$June_Med.1)

## 
##  Wilcoxon rank sum test with continuity correction
## 
## data:  DS_arc_df$June_Med.1 and DS_f_arc_df$June_Med.1
## W = 365.5, p-value = 8.415e-06
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_arc_df$June_Med.2, DS_f_arc_df$June_Med.2)

## 
##  Wilcoxon rank sum test with continuity correction
## 
## data:  DS_arc_df$June_Med.2 and DS_f_arc_df$June_Med.2
## W = 664, p-value = 0.006045
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_arc_df$June_Med.3, DS_f_arc_df$June_Med.3)

## 
##  Wilcoxon rank sum test with continuity correction
## 
## data:  DS_arc_df$June_Med.3 and DS_f_arc_df$June_Med.3
## W = 1066, p-value = 0.6572
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_arc_df$June_Med.4, DS_f_arc_df$June_Med.4)

## 
##  Wilcoxon rank sum test with continuity correction
## 
## data:  DS_arc_df$June_Med.4 and DS_f_arc_df$June_Med.4
## W = 2108, p-value = 3.46e-08
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_arc_df$June_Med.1, DS_m_arc_df$June_Med.1)

## 
##  Wilcoxon rank sum exact test
## 
```

```
## data:  DS_arc_df$June_Med.1 and DS_m_arc_df$June_Med.1
## W = 27, p-value = 4.429e-08
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_arc_df$June_Med.2, DS_m_arc_df$June_Med.2)

##
##  Wilcoxon rank sum exact test
##
## data:  DS_arc_df$June_Med.2 and DS_m_arc_df$June_Med.2
## W = 4, p-value = 3.712e-11
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_arc_df$June_Med.3, DS_m_arc_df$June_Med.3)

##
##  Wilcoxon rank sum exact test
##
## data:  DS_arc_df$June_Med.3 and DS_m_arc_df$June_Med.3
## W = 351, p-value = 0.004125
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_arc_df$June_Med.4, DS_m_arc_df$June_Med.4)

##
##  Wilcoxon rank sum exact test
##
## data:  DS_arc_df$June_Med.4 and DS_m_arc_df$June_Med.4
## W = 450, p-value = 1.571e-09
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_m_arc_df$June_Med.1, DS_f_arc_df$June_Med.1)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  DS_m_arc_df$June_Med.1 and DS_f_arc_df$June_Med.1
## W = 2864, p-value = 0.001234
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_m_arc_df$June_Med.2, DS_f_arc_df$June_Med.2)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  DS_m_arc_df$June_Med.2 and DS_f_arc_df$June_Med.2
## W = 3695.5, p-value = 3.321e-11
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(DS_m_arc_df$June_Med.3, DS_f_arc_df$June_Med.3)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  DS_m_arc_df$June_Med.3 and DS_f_arc_df$June_Med.3
## W = 1076, p-value = 4.552e-05
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(DS_m_arc_df$June_Med.4, DS_f_arc_df$June_Med.4)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  DS_m_arc_df$June_Med.4 and DS_f_arc_df$June_Med.4
## W = 2075, p-value = 0.9967
## alternative hypothesis: true location shift is not equal to 0
```

```
#calculate average non-archaeological values
NA_B_DS_V <- mean(N_Narc_DS[,1])
NA_B_DS_M <- mean(V_Narc_DS[,1])
NA_B_DS <- (NA_B_DS_V+NA_B_DS_M)/2

NA_G_DS_V <- mean(N_Narc_DS[,2])
NA_G_DS_M <- mean(V_Narc_DS[,2])
NA_G_DS <- (NA_B_DS_V+NA_B_DS_M)/2

NA_R_DS_V <- mean(N_Narc_DS[,3])
NA_R_DS_M <- mean(V_Narc_DS[,3])
NA_R_DS <- (NA_B_DS_V+NA_B_DS_M)/2

NA_NIR_DS_V <- mean(N_Narc_DS[,4])
NA_NIR_DS_M <- mean(V_Narc_DS[,4])
NA_NIR_DS <- (NA_B_DS_V+NA_B_DS_M)/2


#wet season
NA_B_WS_V <- mean(N_Narc_WS[,1])
NA_B_WS_M <- mean(V_Narc_WS[,1])
NA_B_WS <- (NA_B_WS_V+NA_B_WS_M)/2

NA_G_WS_V <- mean(N_Narc_WS[,2])
NA_G_WS_M <- mean(V_Narc_WS[,2])
NA_G_WS <- (NA_G_WS_V+NA_G_WS_M)/2

NA_R_WS_V <- mean(N_Narc_WS[,3])
NA_R_WS_M <- mean(V_Narc_WS[,3])
NA_R_WS <- (NA_R_WS_V+NA_R_WS_M)/2

NA_NIR_WS_V <- mean(N_Narc_WS[,4])
NA_NIR_WS_M <- mean(V_Narc_WS[,4])
NA_NIR_WS <- (NA_NIR_WS_V+NA_NIR_WS_M)/2
```

Plot Results

```
#Reorder categories for plotting
WS_df$cat <- factor(WS_df$cat, c("Coastal Communities", "Inland Pastoralist",
"Fishing/Foraging (only)", "Mixed Subsistence"))
DS_df$cat <- factor(DS_df$cat, c("Coastal Communities", "Inland Pastoralist",
"Fishing/Foraging (only)", "Mixed Subsistence"))

library(gridExtra)
B1 <- ggplot(WS_df, aes(WS_df, fill=cat))+
  geom_boxplot(aes(cat, Feb_Med_1.1))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("Blue")+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.position = "none")+
  geom_hline(yintercept=NA_B_WS, color="#4CBB1B", linetype="dotted",
size=1.5)

B2 <- ggplot(WS_df, aes(WS_df, fill=cat))+
  geom_boxplot(aes(cat, Feb_Med_1.2))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("Green")+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.position = "none")+
  geom_hline(yintercept=NA_G_WS, color="#4CBB1B", linetype="dotted",
size=1.5)

B3 <- ggplot(WS_df, aes(WS_df, fill=cat))+
  geom_boxplot(aes(cat, Feb_Med_1.3))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("Red")+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.position = "none")+
  geom_hline(yintercept=NA_R_WS, color="#4CBB1B", linetype="dotted",
size=1.5)

B4 <- ggplot(WS_df, aes(WS_df, fill=cat))+
  geom_boxplot(aes(cat, Feb_Med_1.4))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("NIR")+
```

```
  theme(legend.text = element_text(size = 20))+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.title = element_blank(),
        legend.position="bottom")+
  geom_hline(yintercept=NA_NIR_WS, color="#4CBB1B", linetype="dotted",
size=1.5)

#Creates Figure 3
grid.arrange(B1, B2, B3, B4, ncol=2, nrow=2)



B1 <- ggplot(DS_df, aes(DS_df, fill=cat))+
  geom_boxplot(aes(cat, June_Med_1.1))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("Blue")+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.position = "none")+
  geom_hline(yintercept=NA_B_DS, color="#4CBB1B", linetype="dotted",
size=1.5)

B2 <- ggplot(DS_df, aes(DS_df, fill=cat))+
  geom_boxplot(aes(cat, June_Med_1.2))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("Green")+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.position = "none")+
  geom_hline(yintercept=NA_G_DS, color="#4CBB1B", linetype="dotted",
size=1.5)

B3 <- ggplot(DS_df, aes(DS_df, fill=cat))+
  geom_boxplot(aes(cat, June_Med_1.3))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("Red")+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.position = "none")+
  geom_hline(yintercept=NA_R_DS, color="#4CBB1B", linetype="dotted",
size=1.5)
```

```
B4 <- ggplot(DS_df, aes(DS_df, fill=cat))+
  geom_boxplot(aes(cat, June_Med_1.4))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("NIR")+
  theme(legend.text = element_text(size = 20))+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.title = element_blank(),
        legend.position="bottom")+
  geom_hline(yintercept=NA_NIR_DS, color="#4CBB1B", linetype="dotted",
size=1.5)

#Creates Figure 2
grid.arrange(B1, B2, B3, B4, ncol=2, nrow=2)
```

Annual Spectral Analysis

```
S_Dif <- abs(DS_Comp - WS_Comp)

#Extract Values
N_arch=extract(S_Dif, arch_points, buffer=20, fun=mean)

N_Narc=extract(S_Dif, non_arch_points, buffer=20, fun=mean)

V_arch=extract(S_Dif, V_arch_points, buffer=20, fun=mean)
V_Narc=extract(S_Dif, V_non_arch_points, buffer=20, fun=mean)

M_arc=extract(S_Dif, M_arch_points, buffer=20, fun=mean)

M_arc <- na.omit(M_arc)
F_arc=extract(S_Dif, F_arch_points, buffer=20, fun=mean)

F_arc <- na.omit(F_arc)

#Standardize data
N_arch_s <- abs(standard(N_arch, V_Narc))
N_Narc_s <- abs(standard(N_Narc, V_Narc))
V_arch_s <- abs(standard(V_arch, V_Narc)) #already standardized against
itself
V_Narc_s <- abs(standard(V_Narc, V_Narc))
F_arch_s <- abs(standard(F_arc, V_Narc))
M_arch_s <- abs(standard(M_arc, V_Narc))


#make data frames from each raster point table
arc_df <- data.frame(N_arch_s)
```

```
Narc_df <- data.frame(N_Narc_s)

v_arc_df <- data.frame(V_arch_s)
v_Narc_df <- data.frame(V_Narc_s)

F_arc_df <- data.frame(F_arch_s)
M_Narc_df <- data.frame(M_arch_s)

#add new column named "cat" (for category)
arc_df$cat <- "Inland Pastoralist" #archaeological points
Narc_df$cat <- "Non-Arc-Mikea" #non-archaeological points
v_arc_df$cat <- "Coastal Communities" #archaeological points
v_Narc_df$cat <- "Non-Arc-Velo" #non-archaeological points

F_arc_df$cat <- "Fishing/Foraging (only)"
M_Narc_df$cat <- "Mixed Subsistence"
#combine both dataframes into a single dataframe
df <- rbind(arc_df, v_arc_df, F_arc_df, M_Narc_df)

#Separability Test
M_b1 <- abs((mean(arc_df$layer.1)-
mean(v_arc_df$layer.1))/(sd(arc_df$layer.1)+sd(v_arc_df$layer.1)))
M_b2 <- abs((mean(arc_df$layer.2)-
mean(v_arc_df$layer.2))/(sd(arc_df$layer.2)+sd(v_arc_df$layer.2)))
M_b3 <- abs((mean(arc_df$layer.3)-
mean(v_arc_df$layer.3))/(sd(arc_df$layer.3)+sd(v_arc_df$layer.3)))
M_b4 <- abs((mean(arc_df$layer.4)-
mean(v_arc_df$layer.4))/(sd(arc_df$layer.4)+sd(v_arc_df$layer.4)))

M_b1

## [1] 0.4169188

M_b2

## [1] 0.6161475

M_b3

## [1] 0.6932375

M_b4

## [1] 0.1369413

wilcox.test(arc_df$layer.1, v_arc_df$layer.1)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arc_df$layer.1 and v_arc_df$layer.1
```

```
## W = 1360.5, p-value = 0.0007298
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(arc_df$layer.2, v_arc_df$layer.2)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arc_df$layer.2 and v_arc_df$layer.2
## W = 935, p-value = 9.171e-06
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(arc_df$layer.3, v_arc_df$layer.3)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arc_df$layer.3 and v_arc_df$layer.3
## W = 879, p-value = 4.761e-06
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(arc_df$layer.4, v_arc_df$layer.4)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arc_df$layer.4 and v_arc_df$layer.4
## W = 3171, p-value = 0.2628
## alternative hypothesis: true location shift is not equal to 0

D_B1 <- median(arc_df$layer.1-Narc_df$layer.1)

D_B2 <- median(arc_df$layer.2-Narc_df$layer.2)

D_B3 <- median(arc_df$layer.3-Narc_df$layer.3)

D_B4 <- median(arc_df$layer.4-Narc_df$layer.4)

D_B1v <- median(v_arc_df$layer.1-v_Narc_df$layer.1)

D_B2v <- median(v_arc_df$layer.2-v_Narc_df$layer.2)

D_B3v <- median(v_arc_df$layer.3-v_Narc_df$layer.3)

D_B4v <- median(v_arc_df$layer.4-v_Narc_df$layer.4)

#Calculate average non-archaeological values
#wet season
NA_B_V <- mean(N_Narc[,1])
NA_B_M <- mean(V_Narc[,1])
NA_B <- (NA_B_V+NA_B_M)/2

NA_G_V <- mean(N_Narc[,2])
```

```
NA_G_M <- mean(V_Narc[,2])
NA_G <- (NA_G_V+NA_G_M)/2

NA_R_V <- mean(N_Narc_WS[,3])
NA_R_M <- mean(V_Narc_WS[,3])
NA_R <- (NA_R_V+NA_R_M)/2

NA_NIR_V <- mean(N_Narc[,4])
NA_NIR_M <- mean(V_Narc[,4])
NA_NIR <- (NA_NIR_V+NA_NIR_M)/2

#arrange order of boxplots
df$cat <- factor(df$cat, c("Coastal Communities", "Inland Pastoralist",
"Fishing/Foraging (only)", "Mixed Subsistence", "Non-Arc-Velo", "Non-Arc-
Mikea"))


B1 <- ggplot(df, aes(df, fill=cat))+
  geom_boxplot(aes(cat, layer.1))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("\u0394 Blue")+      #\u0394 is unicode symbol for ??
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.position = "none")+
  geom_hline(yintercept=NA_B, color="#4CBB1B", linetype="dotted", size=1.5)

B2 <- ggplot(df, aes(df, fill=cat))+
  geom_boxplot(aes(cat, layer.2))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("\u0394 Green")+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.position = "none")+
  geom_hline(yintercept=NA_G, color="#4CBB1B", linetype="dotted", size=1.5)

B3 <- ggplot(df, aes(df, fill=cat))+
  geom_boxplot(aes(cat, layer.3))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("\u0394 Red")+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.position = "none")+
  geom_hline(yintercept=NA_R, color="#4CBB1B", linetype="dotted", size=1.5)
```

```
B4 <- ggplot(df, aes(df, fill=cat))+
  geom_boxplot(aes(cat, layer.4))+
  scale_fill_manual(values = c("#E81361","#FFC107","#00DFFF", "#004D40"))+
  xlab("Land use Type")+
  ylab("\u0394 NIR")+
  theme(legend.text = element_text(size = 20))+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  theme(axis.text.x=element_blank(),
        legend.title = element_blank(),
        legend.position="bottom")+
  geom_hline(yintercept=NA_NIR, color="#4CBB1B", linetype="dotted", size=1.5)

#Create Figure 4
gridExtra::grid.arrange(B1, B2, B3, B4, ncol=2, nrow=2)
```

Vegetative (NDVI) Analysis for Wet and Dry Seasons

```
#Extract raster values at each archaeological point location for NDVI
Arc_NDVI_WS=extract(NDVI_Feb, arch_points, buffer=20, fun=mean)

NArc_NDVI_WS=extract(NDVI_Feb, non_arch_points, buffer=20, fun=mean)

V_arch_NDVI_WS=extract(NDVI_Feb, V_arch_points, buffer=20, fun=mean)
V_Narc_NDVI_WS=extract(NDVI_Feb, V_non_arch_points, buffer=20, fun=mean)


M_arc_VI_WS=extract(NDVI_Feb, M_arch_points, buffer=20, fun=mean)
M_arc_VI_WS <- na.omit(M_arc_VI_WS)
F_arc_VI_WS=extract(NDVI_Feb, F_arch_points, buffer=20, fun=mean)
F_arc_VI_WS <- na.omit(F_arc_VI_WS)

#standardize values for wet season
Arc_WS_s <- standard(Arc_NDVI_WS, NArc_NDVI_WS)
NArc_WS_s <- standard(NArc_NDVI_WS, NArc_NDVI_WS)
V_arch_WS_s <- standard(V_arch_NDVI_WS, NArc_NDVI_WS)
V_Narc_WS_s <- standard(V_Narc_NDVI_WS, V_Narc_NDVI_WS) #don't standardize

F_arch_WS_s <- standard(F_arc_VI_WS, NArc_NDVI_WS)
M_arch_WS_s <- standard(M_arc_VI_WS, NArc_NDVI_WS)

#make data frames from each raster point table
arcNDVI_df <- data.frame(Arc_WS_s)
NarcNDVI_df <- data.frame(NArc_WS_s)
V_arch_NDVI_df <- data.frame(V_arch_WS_s)
V_Narc_NDVI_df <- data.frame(V_Narc_WS_s)
M_arc_VI_df <- data.frame(M_arch_WS_s)
F_arc_VI_df <- data.frame(F_arch_WS_s)

#add new column named "cat" (for category)
```

```r
arcNDVI_df$cat <- "Inland Pastoralist" #archaeological points
NarcNDVI_df$cat <- "Non Arch"
V_arch_NDVI_df$cat <- "Coastal Communities"
V_Narc_NDVI_df$cat <- "Non Arch Velo."
M_arc_VI_df$cat <- "Mixed Subsistence"
F_arc_VI_df$cat <- "Fishing/Foraging Only"


#Statistical tests for normality and independence
shapiro.test(arcNDVI_df$Arc_WS_s)

##
##  Shapiro-Wilk normality test
##
## data:  arcNDVI_df$Arc_WS_s
## W = 0.87598, p-value = 0.03361

shapiro.test(NarcNDVI_df$NArc_WS_s)

##
##  Shapiro-Wilk normality test
##
## data:  NarcNDVI_df$NArc_WS_s
## W = 0.91797, p-value = 0.1793

shapiro.test(V_arch_NDVI_df$V_arch_WS_s)

##
##  Shapiro-Wilk normality test
##
## data:  V_arch_NDVI_df$V_arch_WS_s
## W = 0.99395, p-value = 0.1943

shapiro.test(V_Narc_NDVI_df$V_Narc_WS_s)

##
##  Shapiro-Wilk normality test
##
## data:  V_Narc_NDVI_df$V_Narc_WS_s
## W = 0.90646, p-value = 0.0002023

shapiro.test(M_arc_VI_df$M_arch_WS_s)

##
##  Shapiro-Wilk normality test
##
## data:  M_arc_VI_df$M_arch_WS_s
## W = 0.97782, p-value = 0.7805

shapiro.test(F_arc_VI_df$F_arch_WS_s)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  F_arc_VI_df$F_arch_WS_s
## W = 0.89933, p-value = 3.565e-07

wilcox.test(arcNDVI_df$Arc_WS_s, NarcNDVI_df$NArc_WS_s)

## Warning in wilcox.test.default(arcNDVI_df$Arc_WS_s,
NarcNDVI_df$NArc_WS_s):
## cannot compute exact p-value with ties

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arcNDVI_df$Arc_WS_s and NarcNDVI_df$NArc_WS_s
## W = 102, p-value = 0.489
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(arcNDVI_df$Arc_WS_s, V_arch_NDVI_df$V_arch_WS_s) #*

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arcNDVI_df$Arc_WS_s and V_arch_NDVI_df$V_arch_WS_s
## W = 1881, p-value = 0.03713
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(V_arch_NDVI_df$V_arch_WS_s, V_Narc_NDVI_df$V_Narc_WS_s)#***

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  V_arch_NDVI_df$V_arch_WS_s and V_Narc_NDVI_df$V_Narc_WS_s
## W = 17613, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(arcNDVI_df$Arc_WS_s, M_arc_VI_df$M_arch_WS_s)

## Warning in wilcox.test.default(arcNDVI_df$Arc_WS_s,
M_arc_VI_df$M_arch_WS_s):
## cannot compute exact p-value with ties

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arcNDVI_df$Arc_WS_s and M_arc_VI_df$M_arch_WS_s
## W = 166, p-value = 0.1204
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(arcNDVI_df$Arc_WS_s, F_arc_VI_df$F_arch_WS_s)#**
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arcNDVI_df$Arc_WS_s and F_arc_VI_df$F_arch_WS_s
## W = 489, p-value = 0.003059
## alternative hypothesis: true location shift is not equal to 0
```

```
#Calculate average non-archaeological values
#wet season
NDVI_WS_M <- mean(NArc_NDVI_WS)
NDVI_WS_V <- mean(V_Narc_NDVI_WS)
NDVI_WS <- (NDVI_WS_M+NDVI_WS_V)/2


#PLOT RESULTS
NDVI_A <- ggplot(arcNDVI_df, aes(arcNDVI_df))+
  geom_boxplot(aes(cat, Arc_WS_s))+
  xlab("")+
  ylab("NDVI")+
  ylim(-.3,.6)+
  theme(axis.text = element_text(size = 12),
        axis.text.y = element_blank())+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_WS, color="#4CBB1B", linetype="dotted",
size=1.5)

NDVI_NA <- ggplot(NarcNDVI_df, aes(NarcNDVI_df))+
  geom_boxplot(aes(cat, NArc_WS_s))+
  xlab("Non \nArchaeological")+
  ylab("NDVI")+
  ylim(-.3,.8)+
  theme(axis.text = element_text(size = 12))+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_WS, color="#4CBB1B", linetype="dotted",
size=1.5)

NDVI_V <- ggplot(V_arch_NDVI_df, aes(V_arch_NDVI_df))+
  geom_boxplot(aes(cat, V_arch_WS_s))+
  xlab("")+
  ylab("")+
  ylim(-.3,.6)+
  theme(axis.text = element_text(size = 12),
        axis.text.y = element_blank())+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_WS, color="#4CBB1B", linetype="dotted",
size=1.5)

NDVI_V_NA <- ggplot(V_Narc_NDVI_df, aes(V_Narc_NDVI_df))+
  geom_boxplot(aes(cat, V_Narc_NDVI))+
  xlab("")+
```

```
  ylab("")+
  ylim(-.3,.6)+
  theme(axis.text = element_text(size = 12),
        axis.text.y = element_blank())+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_WS, color="#4CBB1B", linetype="dotted",
size=1.5)

NDVI_M <- ggplot(M_arc_VI_df, aes(M_arc_VI_df))+
  geom_boxplot(aes(cat, M_arch_WS_s))+
  xlab("")+
  ylab("")+
  ylim(-.3,.6)+
  theme(axis.text = element_text(size = 12),
        axis.text.y = element_blank())+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_WS, color="#006600", linetype="dotted",
size=1.5)

NDVI_F <- ggplot(F_arc_VI_df, aes(F_arc_VI_df))+
  geom_boxplot(aes(cat, F_arch_WS_s))+
  xlab("")+
  ylab("")+
  ylim(-.3,.6)+
  theme(axis.text = element_text(size = 12),
        axis.text.y = element_blank())+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_WS, color="#006600", linetype="dotted",
size=1.5)


#Create Figure 6
grid.arrange(NDVI_A,NDVI_V, NDVI_M, NDVI_F, ncol=4, nrow=1)



#Extract raster values at each archaeological point location for NDVI
Arc_NDVI_DS=extract(NDVI_Jun, arch_points, buffer=20, fun=mean)

NArc_NDVI_DS=extract(NDVI_Jun, non_arch_points, buffer=20, fun=mean)

V_arch_NDVI_DS=extract(NDVI_Jun, V_arch_points, buffer=20, fun=mean)
V_Narc_NDVI_DS=extract(NDVI_Jun, V_non_arch_points, buffer=20, fun=mean)

M_arc_VI_DS=extract(NDVI_Jun, M_arch_points, buffer=20, fun=mean)
M_arc_VI_DS <- na.omit(M_arc_VI_DS)
F_arc_VI_DS=extract(NDVI_Jun, F_arch_points, buffer=20, fun=mean)
F_arc_VI_DS <- na.omit(F_arc_VI_DS)

#standardize values for wet season
```

```
Arc_DS_s <- standard(Arc_NDVI_DS, NArc_NDVI_DS)
NArc_DS_s <- standard(NArc_NDVI_DS, NArc_NDVI_DS)
V_arch_DS_s <- standard(V_arch_NDVI_DS, NArc_NDVI_DS)
V_Narc_DS_s <- standard(V_Narc_NDVI_DS, V_Narc_NDVI_DS) #don't standardize

F_arch_DS_s <- standard(F_arc_VI_DS, NArc_NDVI_DS)
M_arch_DS_s <- standard(M_arc_VI_DS, NArc_NDVI_DS)


#make data frames from each raster point table
arcNDVI_df <- data.frame(Arc_DS_s)
NarcNDVI_df <- data.frame(NArc_DS_s)
V_arch_NDVI_df <- data.frame(V_arch_DS_s)
V_Narc_NDVI_df <- data.frame(V_Narc_DS_s)
M_arc_VI_df <- data.frame(M_arch_DS_s)
F_arc_VI_df <- data.frame(F_arch_DS_s)

#add new column named "cat" (for category)
arcNDVI_df$cat <- "Inland Pastoralist" #archaeological points
NarcNDVI_df$cat <- "Non Arch"
V_arch_NDVI_df$cat <- "Coastal Communities"
V_Narc_NDVI_df$cat <- "Non Arch Velo."
M_arc_VI_df$cat <- "Mixed Subsistence"
F_arc_VI_df$cat <- "Fishing/Foraging Only"

#Statistical tests for normality and independence
shapiro.test(arcNDVI_df$Arc_DS_s)

##
##  Shapiro-Wilk normality test
##
## data:  arcNDVI_df$Arc_DS_s
## W = 0.86983, p-value = 0.02703

shapiro.test(NarcNDVI_df$NArc_DS_s)

##
##  Shapiro-Wilk normality test
##
## data:  NarcNDVI_df$NArc_DS_s
## W = 0.85103, p-value = 0.01796

shapiro.test(V_arch_NDVI_df$V_arch_DS_s)

##
##  Shapiro-Wilk normality test
##
## data:  V_arch_NDVI_df$V_arch_DS_s
## W = 0.99089, p-value = 0.03378

shapiro.test(V_Narc_NDVI_df$V_Narc_DS_s)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  V_Narc_NDVI_df$V_Narc_DS_s
## W = 0.789, p-value = 6.039e-08
```

```
shapiro.test(M_arc_VI_df$M_arch_DS_s)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  M_arc_VI_df$M_arch_DS_s
## W = 0.92209, p-value = 0.03445
```

```
shapiro.test(F_arc_VI_df$F_arch_DS_s)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  F_arc_VI_df$F_arch_DS_s
## W = 0.98792, p-value = 0.4115
```

```
wilcox.test(arcNDVI_df$Arc_DS_s, NarcNDVI_df$NArc_DS_s)
```

```
## Warning in wilcox.test.default(arcNDVI_df$Arc_DS_s,
NarcNDVI_df$NArc_DS_s):
## cannot compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arcNDVI_df$Arc_DS_s and NarcNDVI_df$NArc_DS_s
## W = 76, p-value = 0.08546
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(arcNDVI_df$Arc_DS_s, V_arch_NDVI_df$V_arch_DS_s)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arcNDVI_df$Arc_DS_s and V_arch_NDVI_df$V_arch_DS_s
## W = 2389.5, p-value = 0.412
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(V_arch_NDVI_df$V_arch_DS_s, V_Narc_NDVI_df$V_Narc_DS_s)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  V_arch_NDVI_df$V_arch_DS_s and V_Narc_NDVI_df$V_Narc_DS_s
## W = 11090, p-value = 0.388
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(arcNDVI_df$Arc_DS_s, M_arc_VI_df$M_arch_DS_s)#*

## Warning in wilcox.test.default(arcNDVI_df$Arc_DS_s,
M_arc_VI_df$M_arch_DS_s):
## cannot compute exact p-value with ties

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arcNDVI_df$Arc_DS_s and M_arc_VI_df$M_arch_DS_s
## W = 145, p-value = 0.04025
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(arcNDVI_df$Arc_DS_s, F_arc_VI_df$F_arch_DS_s)#**

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arcNDVI_df$Arc_DS_s and F_arc_VI_df$F_arch_DS_s
## W = 561, p-value = 0.01439
## alternative hypothesis: true location shift is not equal to 0

#Calculate average non-archaeological values
#wet season
NDVI_DS_M <- mean(NArc_NDVI_DS)
NDVI_DS_V <- mean(V_Narc_NDVI_DS)
NDVI_DS <- (NDVI_DS_M+NDVI_DS_V)/2

#Plot Results

NDVI_A <- ggplot(arcNDVI_df, aes(arcNDVI_df))+
  geom_boxplot(aes(cat, Arc_DS_s))+
  xlab("")+
  ylab("NDVI")+
  ylim(-.1,.5)+
  theme(axis.text = element_text(size = 12))+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_DS, color="#4CBB1B", linetype="dotted",
size=1.5)


NDVI_V <- ggplot(V_arch_NDVI_df, aes(V_arch_NDVI_df))+
  geom_boxplot(aes(cat, V_arch_DS_s))+
  xlab("")+
  ylab("")+
  ylim(-.1,.5)+
  theme(axis.text = element_text(size = 12),
              axis.text.y = element_blank())+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_DS, color="#4CBB1B", linetype="dotted",
size=1.5)
```

```
NDVI_M <- ggplot(M_arc_VI_df, aes(M_arc_VI_df))+
  geom_boxplot(aes(cat, M_arch_DS_s))+
  xlab("")+
  ylab("")+
  ylim(-.1,.5)+
  theme(axis.text = element_text(size = 12),
            axis.text.y = element_blank())+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_DS, color="#4CBB1B", linetype="dotted",
size=1.5)

NDVI_F <- ggplot(F_arc_VI_df, aes(F_arc_VI_df))+
  geom_boxplot(aes(cat, F_arch_DS_s))+
  xlab("")+
  ylab("")+
  ylim(-.1,.5)+
  theme(axis.text = element_text(size = 12),
        axis.text.y = element_blank())+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_DS, color="#4CBB1B", linetype="dotted",
size=1.5)

#Create Figure 5
grid.arrange(NDVI_A,NDVI_V, NDVI_M, NDVI_F, ncol=4, nrow=1)
```

Vegetative (NDVI) Analysis of Annual Variation

```
##ASSESS VEGETATION DIVERSITY/HEALTH CHANGES ACROSS SEASONS

VI_Dif <- abs(NDVI_Feb - NDVI_Jun)

#Extract raster values at each archaeological point location for NDVI
Arc_NDVI_D=extract(VI_Dif, arch_points, buffer=20, fun=mean)

NArc_NDVI_D=extract(VI_Dif, non_arch_points, buffer=20, fun=mean)

V_arch_NDVI_D=extract(VI_Dif, V_arch_points, buffer=20, fun=mean)
V_Narc_NDVI_D=extract(VI_Dif, V_non_arch_points, buffer=20, fun=mean)

M_arc_VI_D=extract(VI_Dif, M_arch_points, buffer=20, fun=mean)
M_arc_VI_D <- na.omit(M_arc_VI_D)
F_arc_VI_D=extract(VI_Dif, F_arch_points, buffer=20, fun=mean)
F_arc_VI_D <- na.omit(F_arc_VI_D)

#standardize NDVI values against non-archaeological zones
Arc_s <- standard(Arc_NDVI_D, NArc_NDVI_D)
NArc_s <- standard(NArc_NDVI_D, NArc_NDVI_D)
```

```r
V_arch_s <- standard(V_arch_NDVI_D, NArc_NDVI_D)
V_Narc_s <- standard(V_Narc_NDVI_D, V_Narc_NDVI_D) #don't standardize

F_arc_s <- standard(F_arc_VI_D, NArc_NDVI_D)
M_arc_s <- standard(M_arc_VI_D, NArc_NDVI_D)

#make data frames from each raster point table
arcNDVI_df <- data.frame(Arc_s)
NarcNDVI_df <- data.frame(NArc_s)
V_arch_NDVI_df <- data.frame(V_arch_s)
V_Narc_NDVI_df <- data.frame(V_Narc_s)
M_arc_VI_df <- data.frame(M_arc_s)
F_arc_VI_df <- data.frame(F_arc_s)


wilcox.test(arcNDVI_df$Arc_s, NarcNDVI_df$NArc_s)

## Warning in wilcox.test.default(arcNDVI_df$Arc_s, NarcNDVI_df$NArc_s):
cannot
## compute exact p-value with ties

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  arcNDVI_df$Arc_s and NarcNDVI_df$NArc_s
## W = 148, p-value = 0.2769
## alternative hypothesis: true location shift is not equal to 0

wilcox.test(V_arch_NDVI_df$V_arch_s, V_Narc_NDVI_df$V_Narc_s)#**

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  V_arch_NDVI_df$V_arch_s and V_Narc_NDVI_df$V_Narc_s
## W = 8146, p-value = 0.007641
## alternative hypothesis: true location shift is not equal to 0

      #add new column named "cat" (for category)
arcNDVI_df$cat <- "Pastoralist" #archaeological points
NarcNDVI_df$cat <- "Non Arch" #archaeological points
V_arch_NDVI_df$cat <- "Velondriake" #non-archaeological points
V_Narc_NDVI_df$cat <- "Non Arch Velo."
M_arc_VI_df$cat <- "Mixed Subsistence"
F_arc_VI_df$cat <- "Foraging Only"

#Calculate average non-archaeological values
NDVI_M <- mean(NArc_NDVI_D)
NDVI_V <- mean(V_Narc_NDVI_D)
NDVI_A <- (NDVI_M+NDVI_V)/2
```

```r
NDVI_Arc <- ggplot(arcNDVI_df, aes(arcNDVI_df))+
  geom_boxplot(aes(cat, Arc_s))+
  xlab("")+
  ylab("\u0394 NDVI")+
  ylim(0,.4)+
  theme(axis.text = element_text(size = 20))+
  theme(axis.title = element_text(size = 20))+
  geom_hline(yintercept=NDVI_A, color="#4CBB1B", linetype="dotted", size=1.5)

NDVI_V <- ggplot(V_arch_NDVI_df, aes(V_arch_NDVI_df))+
  geom_boxplot(aes(cat, V_arch_s))+
  xlab("")+
  ylab("")+
  ylim(0,.4)+
  theme(axis.text = element_text(size = 12),
            axis.text.y = element_blank())+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_A, color="#4CBB1B", linetype="dotted", size=1.5)

NDVI_Mix <- ggplot(M_arc_VI_df, aes(M_arc_VI_df))+
  geom_boxplot(aes(cat, M_arc_s))+
  xlab("")+
  ylab("")+
  ylim(0,.4)+
  theme(axis.text = element_text(size = 12),
            axis.text.y = element_blank())+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_A, color="#4CBB1B", linetype="dotted", size=1.5)

NDVI_FF <- ggplot(F_arc_VI_df, aes(F_arc_VI_df))+
  geom_boxplot(aes(cat, F_arc_s))+
  xlab("")+
  ylab("")+
  ylim(0,.4)+
  theme(axis.text = element_text(size = 12),
      axis.text.y = element_blank())+
  theme(axis.title = element_text(size = 12))+
  geom_hline(yintercept=NDVI_A, color="#4CBB1B", linetype="dotted", size=1.5)


#Create Figure 7
grid.arrange(NDVI_Arc, NDVI_V,
             NDVI_Mix, NDVI_FF, ncol=4, nrow=1)
```