Full and Annotated R Script – Introduction to Image Processing in R

Step 2: Exploring R and RStudio

```
#Try out assigning some text to a variable, then using the print function
greeting <- "Hello world"
print(greeting)
help(print)

#Working with multiple variables, lists, and indexing
x <- 3
y <- 5
z <- 10
max(x, y, z)
numlist <- c(3, 5, 10, 14, 21, 37, 42)
max(numlist)
numlist[5]

#Using simple graphical functions
numlist <- runif(100, min=0, max=1000)
hist(numlist)
plot(numlist)
plot(sort(numlist, decreasing = TRUE))
plot(sort(numlist, decreasing = TRUE), main = "My Random Numbers")
plot(sort(numlist, decreasing = TRUE), main = "My Random Numbers", col = "blue")
```

Step 3: Setting the workspace, loading data, and exploring data properties

```
#Install packages
install.packages("raster")

#Load libraries
library(raster)

#Set workspace
setwd("C:/Users/username/Desktop/IIP_Data_2020")
```

```
#Create a raster object from each band
band2 <- raster('LC08_L1TP_034036_20190918_20190926_01_T1_sr_band2.tif')
band3 <- raster('LC08_L1TP_034036_20190918_20190926_01_T1_sr_band3.tif')
band4 <- raster('LC08_L1TP_034036_20190918_20190926_01_T1_sr_band4.tif')
band5 <- raster('LC08_L1TP_034036_20190918_20190926_01_T1_sr_band5.tif')

#Visualize each band
plot(band2, main = "Band 2 - Blue", col = gray(0:100 / 100))
plot(band3, main = "Band 3 - Green", col = gray(0:100 / 100))
plot(band4, main = "Band 4 - Red", col = gray(0:100 / 100))
plot(band5, main = "Band 5 - NIR", col = gray(0:100 / 100))

#Explore properties of band 2
band2         #prints general information
crs(band2)    #prints coordinate system
ncell(band2)  #prints total number of cells
dim(band2)    #prints number of rows and columns
res(band2)    #prints spatial resolution (in meters)

#Compare properties of the bands
compareRaster(band2, band3, band4, band5)

#Create a true-color composite by combining band4, band3 and band2, and plotting them
in red, green, and blue
landsat432 <- stack(band4, band3, band2)
plotRGB(landsat432, axes=TRUE, stretch="lin", main= "True Color Composite")

#Create a false-color composite by combining band5, band4 and band3, and plotting them
in red, green, and blue
landsat543 <- stack(band5, band4, band3)
plotRGB(landsat543, axes=TRUE, stretch="lin", main= "False Color Composite")
```

Step 4: Image Processing (calculating NDVI, and unsupervised classification)

```
#Crop the scene to Albuquerque

landsat5432 <- stack(band5, band4, band3, band2)

e <- extent(342000, 357000,3884000,3899000)

landsat5432_crop <- crop(landsat5432, e)


#Calculate the NDVI

NIRband <- landsat5432_crop[[1]]

redband <- landsat5432_crop[[2]]

NDVI <- (NIRband - redband) / (NIRband + redband)

plot(NDVI, col = hcl.colors(20, "RdYlGn"), main="Albuquerque NDVI")


#Use histogram to visualize distribution of NDVI values

hist(NDVI, main = 'Distribution of NDVI values', xlab="NDVI value", xlim = c(-1, 1),
breaks = 40)


#Reclassify via thresholding (where NDVI values are greater than 0.4)

vegonly <- reclassify(NDVI, cbind(-Inf, 0.4, NA))

plot(vegonly, col = "green3" , main="Albuquerque Vegetation")


#Install and load color palette package

install.packages("RColorBrewer")

library(RColorBrewer)


#Retrieve the values of the NDVI raster object as a matrix

ndvi_matrix <- getValues(NDVI)

str(ndvi_matrix)


#Perform k-means clustering on the matrix

set.seed(99)

cluster_values <- kmeans(x = na.omit(ndvi_matrix), centers = 6, iter.max = 500, nstart
= 5, algorithm = "Lloyd")


#View details about the matrix produced by clustering

head(cluster_values)
```

```
#Produce raster object of cluster values

classified_image <- setValues(NDVI, cluster_values$cluster)


plot(classified_image, main = "Unsupervised Classification - k-means", col =
brewer.pal(6, "Set1"))


par(mfrow = c(1,2))

plot(vegonly, col = "green3" , main="Albuquerque Vegetation")

plot(classified_image, main = "Unsupervised Classification - k-means", col =
brewer.pal(6, "Set1"))


#Save the results

writeRaster(classified_image, filename="MyResults.tif", overwrite=TRUE)
```