Transpose Columns to Rows

# The following code is designed to reformat winter colony count data where species counts were recorded in columns.

# The code provided will transpose species columns into rows that include the site metadata, along with the species code and count value

#INSTALL AND LOAD REQUIRED PACKAGES

install.packages(“tidyr”)

install.packages(“reshape2”)

library(tidyr)

library(reshape2)

# READ IN METADATA THAT NEEDS TO BE REFORMATTED BY PROVIDING THE FILEPATH AND FILENAME,

# AND RENAME THE OBJECT

# In this example, the file being read in is named “Winter\_hibernacula\_counts.csv” and it

# is renamed as “HIB\_COUNTS”

# Be sure that you include the file extension (.csv) in the filename

HIB\_COUNTS<-read.csv(“E:/Winter\_Counts/Winter\_hibernacula\_counts.csv”)

# SPECIFY WHICH COLUMNS FROM THE ORIGINAL DATA YOU WISH TO RETAIN AS COLUMNS AND

# WHICH COLUMNS FROM THE ORIGINAL DATA YOU WISH TO TRANSPOSE INTO ROWS

# “id.vars” will be retained as columns

# “measure.vars” will be transposed into rows

HIB\_MELT<-melt(HIB\_COUNTS,id.vars=c(“X..Site.Identifier”,“GRTS.Cell.Id”,“SURVEYDATE”,“Survey.Start.Time”, “Survey.End.Time”,“Observer”,“Site.Name”),measure.vars=c(“MYLU”,“MYSO”,“MYLE”, “MYSE”,“EPFU”,“PESU”),variable.name=“Species\_Code”,value.name=“Count”)

# VIEW THE NEW DATA STRUCTURE TO ENSURE IT IS CORRECT

HIB\_MELT

# EXPORT THE REFORMATTED DATA INTO CSV FORMAT BY PROVIDING THE NAME OF THE DATA OBJECT TO EXPORT,

# THE FILEPATH DESTINATION FOR THE CSV, AND WHAT YOU WISH TO NAME THE CSV

# In this example, the reformatted data to export is “HIB\_MELT”, the filepath is “C:/Users/danesmith/Desktop”,

# and the new filename is “HIB\_MELT.csv”

# Be sure to include the file extension in the filename (.csv)

write.csv(HIB\_MELT,“E:/Winter\_Counts/HIB\_MELT.csv”)