RBighornVisitationAnalysis

This is my preliminary analysis of bighorn visitation to water, Summer 2019. First, I combined all of my observations into one dataframe. Due to some technical difficulties with the database at the CDFW office, I included observations from both a version of the database from 7/3 and 7/22 since I thought some had gotten deleted from the 7/22. I then removed fully duplicated rows in order to remove data that were in both the 7/3 and 7/22 database. The dataframe had 2108 observations before eliminating duplicated rows, and 2062 observations after the removal of duplicated rows. This makes sense as the approximate number of observations that I had input into the database by 7/3. Before coding further, I went through the SBrs and Marb data and visually checked that there were observations for every date that were supposedly covered. The only discrepancy that I found was that there were no observations in the Marbles on 6/25 after 14:49 pm. I went into the photo data and confirmed that this was indeed the case. Perhaps it was a really cool afternoon?

I then formatted the data where major things needed to be fixed. I changed the cells of KI\_DG\_FirstPhotoTime2, KI\_DG\_FirstPhotoTime3, KI\_DG\_FirstPhotoTime4, KI\_DG\_LastPhotoTime2, KI\_DG\_LastPhotoTime3, and KI\_DG\_LastPhotoTime4 that were ‘0’ or blank to NAs. I also changed the cells of KI\_DG\_SheepDrank2, KI\_DG\_SheepDrank3, and KI\_DG\_SheepDrank4 that were blank to NAs.

AshleyDataall <- read.csv("AshleyDatabasequeryall121419.csv", header = TRUE, sep = ",")  
twodbobs <- read.csv("ObservationsBeginningoftheSummer2databases121419.csv", header = TRUE, sep = ",")  
firstpartdgobs <- read.csv("NopaCMPRMarbfirstpartofsummerDanielleObservations.csv", header = TRUE, sep = ",")  
threedf <- rbind(AshleyDataall, twodbobs, firstpartdgobs)

## Warning in `[<-.factor`(`\*tmp\*`, ri, value = c(6120703L, 6120703L,  
## 6120703L, : invalid factor level, NA generated  
  
## Warning in `[<-.factor`(`\*tmp\*`, ri, value = c(6120703L, 6120703L,  
## 6120703L, : invalid factor level, NA generated

library(tidyverse)

## -- Attaching packages ----------------------------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.1 v purrr 0.3.2  
## v tibble 2.1.3 v dplyr 0.8.3  
## v tidyr 1.0.0 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.4.0

## -- Conflicts -------------------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

nodups <- distinct(threedf)  
# table(duplicated(nodups))  
# table(duplicated(nodups$DateObsTimeIDwaterspecific))  
# THERE ARE NO DUPLICATES OF THE ROWS OVERALL, BUT THERE ARE 204 DUPLICATES OF THE NODUPS$DateObsTimeODwaterspecific. WHY?!!!!!!! B/C some of the observations have the capture data and some don't. I SHOULD BE ABLE TO FIX THIS AFTER INPUTTING THE CORRECT WEIGHTS, BODY LENGTHS, AND CHEST GIRTHS THAT ARE MISSING.  
# View(duplicated(nodups$DateObsTimeIDwaterspecific))  
  
# as.character(nodups$KI\_DG\_FirstPhotoTime2, nodups$KI\_DG\_LastPhotoTime2, nodups$KI\_DG\_FirstPhotoTime3, nodups$KI\_DG\_LastPhotoTime3, nodups$KI\_DG\_FirstPhotoTime4, nodups$KI\_DG\_LastPhotoTime4)  
nodups$KI\_DG\_FirstPhotoTime2[nodups$KI\_DG\_FirstPhotoTime2 == "0"] <- NA  
nodups$KI\_DG\_FirstPhotoTime2[nodups$KI\_DG\_FirstPhotoTime2 == ""] <- NA  
nodups$KI\_DG\_LastPhotoTime2[nodups$KI\_DG\_LastPhotoTime2 == "0"] <- NA  
nodups$KI\_DG\_LastPhotoTime2[nodups$KI\_DG\_LastPhotoTime2 == ""] <- NA  
nodups$KI\_DG\_FirstPhotoTime3[nodups$KI\_DG\_FirstPhotoTime3 == "0"] <- NA  
nodups$KI\_DG\_FirstPhotoTime3[nodups$KI\_DG\_FirstPhotoTime3 == ""] <- NA  
nodups$KI\_DG\_LastPhotoTime3[nodups$KI\_DG\_LastPhotoTime3 == "0"] <- NA  
nodups$KI\_DG\_LastPhotoTime3[nodups$KI\_DG\_LastPhotoTime3 == ""] <- NA  
nodups$KI\_DG\_FirstPhotoTime4[nodups$KI\_DG\_FirstPhotoTime4 == "0"] <- NA  
nodups$KI\_DG\_FirstPhotoTime4[nodups$KI\_DG\_FirstPhotoTime4 == ""] <- NA  
nodups$KI\_DG\_LastPhotoTime4[nodups$KI\_DG\_LastPhotoTime4 == "0"] <- NA  
nodups$KI\_DG\_LastPhotoTime4[nodups$KI\_DG\_LastPhotoTime4 == ""] <- NA  
  
  
# as.character(nodups$KI\_DG\_SheepDrank2, nodups$KI\_DG\_SheepDrank3, nodups$KI\_DG\_SheepDrank4)  
nodups$KI\_DG\_SheepDrank2[nodups$KI\_DG\_SheepDrank2 == ""] <- NA  
nodups$KI\_DG\_SheepDrank3[nodups$KI\_DG\_SheepDrank3 == ""] <- NA  
nodups$KI\_DG\_SheepDrank4[nodups$KI\_DG\_SheepDrank4 == ""] <- NA  
  
# as.character(nodups$GO\_DG\_SpecificSource)  
nodups$GO\_DG\_SpecificSource[nodups$GO\_DG\_SpecificSource == ""] <- NA  
# as.factor(nodups$GO\_DG\_SpecificSource)

I noted that there were still some duplicate observations that had not been removed despite the previous code chunk. In an attempt to figure out why there were still duplicated rows, I made a column with everything needed to make each observation individually identifiable. I discovered that some rows did not have the capture data needed, and that this was what was making the duplicated rows not seen as duplicated. I then worked to input the missing information - weight, body length, and chest girth. I subset by the distinct rows and this equaled the previous number of individually identifiable observations. No more duplicates!

# as.character(nodups$GO\_Date, nodups$KI\_Known\_ID, nodups$GO\_Time\_HHMM, nodups$GO\_WaterSource, nodups$GO\_DG\_SpecificSource)  
nodups$DateObsTimeIDwaterspecific <- paste(nodups$GO\_Date, nodups$KI\_Known\_ID, nodups$GO\_Time\_HHMM, nodups$GO\_WaterSource, nodups$GO\_DG\_SpecificSource, sep = "")  
# as.character(nodups$KI\_Sex)  
# as.character(nodups$KI\_Known\_ID)  
# table(nodups$KI\_Sex)  
nodups$KI\_Sex[nodups$KI\_Known\_ID == "BHS\_1489" | nodups$KI\_Known\_ID == "BHS\_1596" | nodups$KI\_Known\_ID == "BHS\_1361" | nodups$KI\_Known\_ID == "BHS\_1357" | nodups$KI\_Known\_ID == "BHS\_1367" | nodups$KI\_Known\_ID == "BHS\_1362" | nodups$KI\_Known\_ID == "BHS\_1584" | nodups$KI\_Known\_ID == "BHS\_1585" | nodups$KI\_Known\_ID == "BHS\_1583" | nodups$KI\_Known\_ID == "BHS\_1582" | nodups$KI\_Known\_ID == "BHS\_1588" | nodups$KI\_Known\_ID == "BHS\_1365" | nodups$KI\_Known\_ID == "BHS\_1574" | nodups$KI\_Known\_ID == "BHS\_1366" | nodups$KI\_Known\_ID == "BHS\_1594" | nodups$KI\_Known\_ID == "BHS\_1577" | nodups$KI\_Known\_ID == "BHS\_1581" | nodups$KI\_Known\_ID == "BHS\_1576" | nodups$KI\_Known\_ID == "BHS\_1416" | nodups$KI\_Known\_ID == "BHS\_1587" | nodups$KI\_Known\_ID == "BHS\_1575" | nodups$KI\_Known\_ID == "BHS\_1592" | nodups$KI\_Known\_ID == "BHS\_1589" | nodups$KI\_Known\_ID == "BHS\_1421" | nodups$KI\_Known\_ID == "BHS\_1740" | nodups$KI\_Known\_ID == "BHS\_1738" | nodups$KI\_Known\_ID == "BHS\_1736" | nodups$KI\_Known\_ID == "BHS\_1734" | nodups$KI\_Known\_ID == "BHS\_1733" | nodups$KI\_Known\_ID == "BHS\_1741"] <- "F"  
  
nodups$KI\_Sex[nodups$KI\_Known\_ID == "BHS\_1494" | nodups$KI\_Known\_ID == "BHS\_1684" | nodups$KI\_Known\_ID == "BHS\_1683" | nodups$KI\_Known\_ID == "BHS\_1682" | nodups$KI\_Known\_ID == "BHS\_1493" | nodups$KI\_Known\_ID == "BHS\_1591" | nodups$KI\_Known\_ID == "BHS\_1590" | nodups$KI\_Known\_ID == "BHS\_1420" | nodups$KI\_Known\_ID == "1728" | nodups$KI\_Known\_ID == "BHS\_1739"] <- "M"  
  
# table(nodups$KI\_Known\_ID) - going through the individual sheep to check and make sure their weight, body length, and chest girth are filled out correctly.  
# table(nodups$KI\_Known\_ID[!is.na()])  
  
# LIST OF SHEEP THAT HAVE MEASUREMENTS (\* sheep that had different measurements?, ^ sheep that I am checking measurements with CDFW): BHS\_1420^, BHS\_1428\*, BHS\_1490\*, BHS\_1492^, 1493\*, 1494\*, 1563, 1565, 1574, 1575, 1576, 1577, 1581, 1582, 1583, 1587, 1592, 1594, 1682, 1683, 1684, 1689, 1713, 1730, 1731, 1733, 1734, 1736, 1738, 1739  
# nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == ""]  
#### NEED TO FIX ID THAT HAS "BHs\_1596" to "BHS\_1596"!!!!!!!!  
  
# as.character(nodups$KI\_Known\_ID)  
nodups$KI\_Known\_ID[nodups$KI\_Known\_ID == "BHs\_1596"] <- "BHS\_1596"  
# as.factor(nodups$KI\_Known\_ID)  
# now comes back at BHs\_1596 = 0, BHS\_1596=78 - did this truly fix the problem?  
  
# Four of the bighorn have two different numbers at their measurements. I am going to set them to the lower number b/c the higher number is probably the untarred number.  
#BHS\_1428 - body length comes back at both 151.3 and 149.0. I am going to set it to 149.0 b/c that is what in the 7/22 database and because the higher number is probably the untarred number.  
  
# There are measurements that are not in the 7/22 database that are in my datasheet. I need to add the measurements for the sheep that this is true so as to make there be no duplicated rows.  
  
# BODY LENGTHS:  
# as.character(nodups$CP\_BdyLgth\_cm)  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1420"] <- "155.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1428"] <- "149.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1490"] <- "158.2"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1492"] <- "152.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1493"] <- "162.5"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1494"] <- "145.1"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1563"] <- "165.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1565"] <- "136.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1574"] <- "146.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1575"] <- "143.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1576"] <- "141.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1577"] <- "141.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1581"] <- "139.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1582"] <- "138.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1583"] <- "142.5"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1587"] <- "139.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1592"] <- "137.5"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1594"] <- "125.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1682"] <- "140.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1683"] <- "144.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1684"] <- "149.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1689"] <- "155.5"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1713"] <- "140.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1730"] <- "152.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1731"] <- "161.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1733"] <- "153.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1734"] <- "149.5"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1736"] <- "118.0"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1738"] <- "130.5"  
nodups$CP\_BdyLgth\_cm[nodups$KI\_Known\_ID == "BHS\_1739"] <- "145.0"  
# as.numeric(nodups$CP\_BdyLgth\_cm)  
  
  
#CHEST GIRTHS  
# as.character(nodups$CP\_ChstGth\_cm)  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1420"] <- "111.5"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1428"] <- "100.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1490"] <- "101.8"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1492"] <- "105.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1493"] <- "111.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1494"] <- "104.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1563"] <- "107.2"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1565"] <- "99.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1574"] <- "95.5"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1575"] <- "99.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1576"] <- "97.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1577"] <- "99.1"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1581"] <- "101.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1582"] <- "92.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1583"] <- "93.2"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1587"] <- "97.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1592"] <- "103.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1594"] <- "98.9"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1682"] <- "98.5"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1683"] <- "95.5"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1684"] <- "102.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1689"] <- "108.5"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1713"] <- "97.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1730"] <- "101.5"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1731"] <- "115.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1733"] <- "109.5"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1734"] <- "96.0"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1736"] <- "82.5"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1738"] <- "84.5"  
nodups$CP\_ChstGth\_cm[nodups$KI\_Known\_ID == "BHS\_1739"] <- "96.0"  
# as.numeric(nodups$CP\_ChstGth\_cm)  
  
#WEIGHT - this is unadjusted, raw weight (no tar)b/c the vast majority of the entries do not have a tar weight or a net weight.  
# as.character(nodups$CP\_Weight..kg.)  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1420"] <- "96.5"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1428"] <- "63.0"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1490"] <- "82.0"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1492"] <- "88.4"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1493"] <- "87.4"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1494"] <- "64.6"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1563"] <- "89.8"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1565"] <- "61.3"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1574"] <- "63.6"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1575"] <- "67.1"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1576"] <- "58.5"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1577"] <- "61.3"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1581"] <- "68.0"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1582"] <- "59.2"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1583"] <- "63.0"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1587"] <- "65.5"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1592"] <- "60.5"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1594"] <- "56.5"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1682"] <- "61.0"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1683"] <- "64.6"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1684"] <- "76.3"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1689"] <- "85.2"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1713"] <- "59.6"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1730"] <- "69.0"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1731"] <- "100.04"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1733"] <- "77.3"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1734"] <- "58.6"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1736"] <- "47.1"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1738"] <- "44.2"  
nodups$CP\_Weight..kg.[nodups$KI\_Known\_ID == "BHS\_1739"] <- "71.8"  
# as.numeric(nodups$CP\_Weight..kg.)  
  
# AGE AT TIME OF CAPTURE:  
# as.character(nodups$CP\_AgeYears)  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1420"] <- "6"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1428"] <- "3"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1490"] <- "7"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1492"] <- "8"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1493"] <- "7"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1494"] <- "4"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1563"] <- "4"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1565"] <- "4"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1574"] <- "10"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1575"] <- "5"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1576"] <- "8"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1577"] <- "5"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1581"] <- "5"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1582"] <- "5"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1583"] <- "6"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1587"] <- "6"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1592"] <- "5"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1594"] <- "7"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1682"] <- "3"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1683"] <- "3"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1684"] <- "12"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1689"] <- "9"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1713"] <- "6"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1730"] <- "4"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1731"] <- "8"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1733"] <- "8"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1734"] <- "6"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1736"] <- "2"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1738"] <- "1"  
nodups$CP\_AgeYears[nodups$KI\_Known\_ID == "BHS\_1739"] <- "4"  
# as.integer(nodups$CP\_AgeYears)  
  
  
library(tidyverse)  
renodups <- distinct(nodups)  
# table(duplicated(renodups))  
# table(duplicated(renodups$DateObsTimeIDwaterspecific))  
# Before Age Years, 177 duplicates.  
# Down to 80 duplicates after Age Years fixed.  
# Now down to 54 duplicates.  
# No more duplicates! Yay!  
# View(duplicated(renodups$DateObsTimeIDwaterspecific))

Then, just to be sure, I cut the visitation data to the time periods when all cameras were running. This data should have already been cut, but I know that Ashley was not super meticulous in only going through the photos in the time period when all of the cameras in the range were running, so I feel like this is a good check.

At this time, I also checked that the GO\_WaterSource and GO\_RangeCode made sense for each range. I changed two entries that were actually Nopah but mistakenly CMPR.

# as.character(renodups$GO\_Date, renodups$GO\_Time\_HHMM)  
renodups$DateTime <- paste(renodups$GO\_Date, renodups$GO\_Time\_HHMM, sep = " ")  
library(lubridate)

##   
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':  
##   
## date

renodups$DateTime <- strptime(renodups$DateTime, "%Y%m%d %H:%M")  
renodups$DateTimect <- as.POSIXct(renodups$DateTime, tz = "America/Los\_Angeles", format = "%Y%m%d %H:%M")  
  
# Making the GO\_Date into a Date:  
renodups$GO\_Date <- strptime(renodups$GO\_Date, "%Y%m%d")  
renodups$GO\_Date <- as.POSIXct(renodups$GO\_Date, tz = "America/Los\_Angeles", format = "%Y%m%d")  
  
Marbobs <- renodups[renodups$GO\_RangeCode == "Marb", ]  
Marbcutobs <- Marbobs[(Marbobs$DateTimect >= as.POSIXct("2019-05-17 10:49:00") & Marbobs$DateTimect <= as.POSIXct("2019-05-24 23:59:59")) | (Marbobs$DateTimect >= as.POSIXct("2019-06-04 20:00:00") & Marbobs$DateTimect <= as.POSIXct("2019-06-12 13:32:00")) | (Marbobs$DateTimect >= as.POSIXct("2019-06-18 17:19:00") & Marbobs$DateTimect <= as.POSIXct("2019-06-22 23:59:59"))| (Marbobs$DateTimect >= as.POSIXct("2019-06-25 14:49:00") & Marbobs$DateTimect <= as.POSIXct("2019-06-30 10:46:00"))| (Marbobs$DateTimect >= as.POSIXct("2019-07-10 08:45:00") & Marbobs$DateTimect <= as.POSIXct("2019-07-13 23:00:18")) | (Marbobs$DateTimect >= as.POSIXct("2019-07-17 11:46:00") & Marbobs$DateTimect <= as.POSIXct("2019-07-25 10:30:00")) | (Marbobs$DateTimect >= as.POSIXct("2019-07-31 20:52:00") & Marbobs$DateTimect <= as.POSIXct("2019-08-04 13:13:03")) | (Marbobs$DateTimect >= as.POSIXct("2019-08-06 12:03:00") & Marbobs$DateTimect <= as.POSIXct("2019-08-10 12:03:23")), ]  
  
BrsSobs <- renodups[renodups$GO\_RangeCode == "BrsS", ]  
BrsScutobs <- BrsSobs[(BrsSobs$DateTimect >= as.POSIXct("2019-06-04 15:04:00") & BrsSobs$DateTimect <= as.POSIXct("2019-06-16 08:27:00")) | (BrsSobs$DateTimect >= as.POSIXct("2019-06-18 17:36:00") & BrsSobs$DateTimect <= as.POSIXct("2019-06-25 16:25:00")) | (BrsSobs$DateTimect >= as.POSIXct("2019-06-25 17:05:00") & BrsSobs$DateTimect <= as.POSIXct("2019-07-29 10:36:00")) | (BrsSobs$DateTimect >= as.POSIXct("2019-08-01 14:32:00") & BrsSobs$DateTimect <= as.POSIXct("2019-08-12 11:13:00")),]  
  
# I found a couple of entries that need to be fixed.  
# as.character(renodups$GO\_RangeCode)  
renodups$GO\_RangeCode[renodups$DateObsTimeIDwaterspecific == "20190718BHS\_174112:46NopaNA"] <- "Nopa"  
renodups$GO\_RangeCode[renodups$DateObsTimeIDwaterspecific == "20190722BHS\_17405:26NopaNA"] <- "Nopa"  
# as.factor(renodups$GO\_RangeCode)  
  
cmprobs <- renodups[renodups$GO\_RangeCode == "CMPR", ]  
cmprcutobs <- cmprobs[(cmprobs$DateTimect >= as.POSIXct("2019-05-23 15:48:00") & cmprobs$DateTimect <= as.POSIXct("2019-05-31 11:18:00")) | (cmprobs$DateTimect >= as.POSIXct("2019-06-20 17:47:00") & cmprobs$DateTimect <= as.POSIXct("2019-06-27 02:09:00")) | (cmprobs$DateTimect >= as.POSIXct("2019-07-11 18:25:00") & cmprobs$DateTimect <= as.POSIXct("2019-07-14 18:28:00")) | (cmprobs$DateTimect >= as.POSIXct("2019-07-30 14:30:00") & cmprobs$DateTimect <= as.POSIXct("2019-08-03 16:46:00")) | (cmprobs$DateTimect >= as.POSIXct("2019-08-05 19:10:00") & cmprobs$DateTimect <= as.POSIXct("2019-08-15 07:27:00")),]  
  
nopaobs <- renodups[renodups$GO\_RangeCode == "Nopa", ]  
nopacutobs <- nopaobs[(nopaobs$DateTimect >= as.POSIXct("2019-06-06 09:18:00") & nopaobs$DateTimect <= as.POSIXct("2019-06-13 11:04:00")) | (nopaobs$DateTimect >= as.POSIXct("2019-06-25 09:00:00") & nopaobs$DateTimect <= as.POSIXct("2019-07-08 12:35:00")) | (nopaobs$DateTimect >= as.POSIXct("2019-07-16 19:03:00") & nopaobs$DateTimect <= as.POSIXct("2019-07-29 15:06:00")) | (nopaobs$DateTimect >= as.POSIXct("2019-07-29 19:55:00") & nopaobs$DateTimect <= as.POSIXct("2019-08-07 14:49:00")),]  
  
allcut <- rbind(Marbcutobs, BrsScutobs, cmprcutobs, nopacutobs)  
# write.xlsx(allcut, file = "allcutVisitationData122719.xlsx")

I checked to make sure that every sheep that I thought was alive for the summer actually had documented proof of life for the entire summer. When counting the number of sheep in my sample, I made sure not to include sheep that are assumed dead (Marbles: BHS\_1413 (dead harvest), BHS\_1496, BHS\_1597, South Bristols: BHS\_1340, BHS\_1331, BHS\_1598). I also made sure to include sheep that were not seen at water all summer, but are documented to be alive (Marbles BHS\_1497, Nopah BHS\_1737). All of the South Bristols, and all of the CMPR sheep that I limited this analysis to (for reasons listed below), came into water at least once during the summer.

Some notes on the whereabouts of the sheep that were not see at water as per their GPS collars. Marb BHS\_1589 literally just chilled on Castle Peak. BHS\_1497’s collar is no longer working. Nopah BHS\_1737 was between the Nopah and El Si Guzzler - looks like stayed as high elevation? I specifically examined BHS\_1595, which only had 3 visitations prior to its death on 8/5. BHS\_1595 was in the southern Marbles, then moved up to just south of Castle Peak. There is a chance that there is an unknown water source in this location.

It is worth noting that a couple Marbles sheep died during the summer. BHS\_ 1367 was heard dead on flight on 8/19 and last seen 6/12 at Theresa. Given this information, I am making the conservative assumption in analyses that it died 6/12 and only including it through 6/11 in analyses. BHS\_1595 died on either 8/5 or 8/6, but stopped moving on the 5th. Therefore, I am only including it in analyses through 8/4.

After looking at the GPS data for the summer, I realized that sheep at Paiute Guzzler were making their way down to a water source in the south of the Castle Piutes that I did not put a camera on. As such, I limited my analysis to the sheep that were only located in the far north section of the range, by Viceroy Mine.

I fixed some observations that were incorrect, and added some observations that were supposed to be in the Marbles but were in the South Bristols to the Marbles.

## Individually Identifiable Bighorn in Visitation Preliminary Analyses  
  
# Marbles: BHS\_1367, BHS\_1365, BHS\_1361, BHS\_1362", BHS\_1357, BHS\_1416, BHS\_1421, BHS\_1581, BHS\_1577, BHS\_1574, BHS\_1576, BHS\_1583, BHS\_1584, BHS\_1585, BHS\_1587, BHS\_1582, BHS\_1575, BHS\_1592, BHS\_1594, BHS\_1420, BHS\_1493, BHS\_1494, BHS\_1590, BHS\_1591, BHS\_1682, BHS\_1683, BHS\_1684, BHS\_1366, BHS\_1497, BHS\_1588, BHS\_1595, BHS\_1589"  
# \*\*\* This currently includes 1497 (never visited, Lily said seen alive last November) and 1592 (gps track and visited twice), but not 1496 (assumed dead).  
  
# South Bristols: BHS\_1330, BHS\_1337, BHS\_1334, BHS\_1329, BHS\_1328, BHS\_1333, BHS\_1425, BHS\_1426, BHS\_1489, BHS\_1491, BHS\_1578, BHS\_1579, BHS\_1580, BHS\_1596, BHS\_1599, BHS\_1687, BHS\_1486, BHS\_1490, BHS\_1492, BHS\_1428, BHS\_1685, BHS\_1686, BHS\_1688, BHS\_1689  
  
# Nopah: BHS\_1737, BHS\_1740, BHS\_1741, BHS\_1738, BHS\_1736, BHS\_1739  
  
# Castle Piutes: BHS\_1730, BHS\_1733, BHS\_1734, BHS\_1728  
  
# \*\*\* DG note to self: 1728 doesn't have a working GPS collar  
  
## Fixing observations  
  
# One observation accidentally 1578 instead of 1576  
Marbcutobs$KI\_Known\_ID[Marbcutobs$DateObsTimeIDwaterspecific == "20190804BHS\_15787:24VernNA"] <- "BHS\_1576"  
# A different observation 1685 instead of 1590  
Marbcutobs$KI\_Known\_ID[Marbcutobs$DateObsTimeIDwaterspecific == "20190722BHS\_168511:23TherBoss Tank"] <- "BHS\_1590"  
# Another observation changing 1713 for 1582  
Marbcutobs$KI\_Known\_ID[Marbcutobs$DateObsTimeIDwaterspecific == "20190803BHS\_17139:37TherBoss Tank"] <- "BHS\_1582"

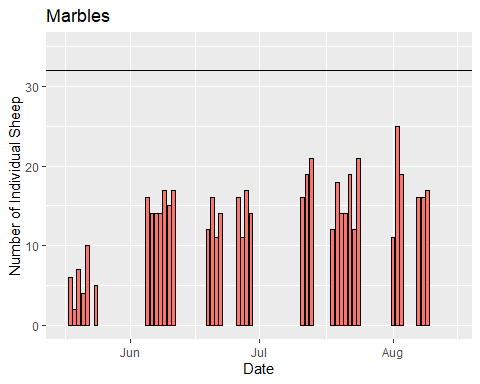
I then created figures of the number of individual sheep coming into water by day. I only included times when all cameras in the mountain range were running. This is given that sheep use more than one water source and move around throughout the mountain range. When examining the data, I noticed that the days at the beginning and end of each period when all of the cameras were running had a lower number of sheep coming in than the rest of the days. This is easily attributable to the cameras being up and running for only part of these days. As such, I cut out these days from my dataset (which probably decreased by dataset by about ~10%).

I did leave in a couple of days that had very near complete coverage. I decided that it was ok to keep these days given the short periods of time when 1 of the cameras was down for, and the relative bias of sheep for diurnality, which was confirmed when going through camera photos. For the Marbles, I did leave in one day when the cameras were only not running between 11 pm and 12 am. In the South Bristols, I included 6/25/19 which was not fully covered for 45 minutes. In the Castle Piutes, I included 7/14 and 8/3 because one of the cameras went out at night. I will need to limit other analyses to these time periods given the incomplete data on the start/end days.

Afterwards, I realized that the plots that I made here are not nearly as useful as percent visitation by day, as I did in the next chunk.

# Number of Individual Sheep Coming into Water by Day  
  
# Marbles  
  
marballsheepvisitsday <- as.data.frame(table(Marbcutobs$KI\_Known\_ID, Marbcutobs$GO\_Date))  
# View(marballsheepvisitsday)  
# Took out 1413 b/c dead, 1589 and 1597 because assumed dead  
marballsheepvisitsday1 <- marballsheepvisitsday[(marballsheepvisitsday$Var1 == "BHS\_1367") | (marballsheepvisitsday$Var1 == "BHS\_1365") | (marballsheepvisitsday$Var1 == "BHS\_1361") |(marballsheepvisitsday$Var1 == "BHS\_1362") | (marballsheepvisitsday$Var1 == "BHS\_1357") | (marballsheepvisitsday$Var1 == "BHS\_1416") | (marballsheepvisitsday$Var1 == "BHS\_1421") | (marballsheepvisitsday$Var1 == "BHS\_1581") | (marballsheepvisitsday$Var1 == "BHS\_1577") | (marballsheepvisitsday$Var1 == "BHS\_1574") | (marballsheepvisitsday$Var1 == "BHS\_1576") | (marballsheepvisitsday$Var1 == "BHS\_1583") | (marballsheepvisitsday$Var1 == "BHS\_1584") | (marballsheepvisitsday$Var1 == "BHS\_1585") | (marballsheepvisitsday$Var1 == "BHS\_1587") | (marballsheepvisitsday$Var1 == "BHS\_1582") | (marballsheepvisitsday$Var1 == "BHS\_1575") | (marballsheepvisitsday$Var1 == "BHS\_1592") | (marballsheepvisitsday$Var1 == "BHS\_1594") | (marballsheepvisitsday$Var1 == "BHS\_1420") | (marballsheepvisitsday$Var1 == "BHS\_1493") | (marballsheepvisitsday$Var1 == "BHS\_1494") | (marballsheepvisitsday$Var1 == "BHS\_1590") | (marballsheepvisitsday$Var1 == "BHS\_1591") | (marballsheepvisitsday$Var1 == "BHS\_1682") | (marballsheepvisitsday$Var1 == "BHS\_1683") | (marballsheepvisitsday$Var1 == "BHS\_1684") | (marballsheepvisitsday$Var1 == "BHS\_1366") | (marballsheepvisitsday$Var1 == "BHS\_1497") | (marballsheepvisitsday$Var1 == "BHS\_1588") | (marballsheepvisitsday$Var1 == "BHS\_1595") | (marballsheepvisitsday$Var1 == "BHS\_1589"), ]  
  
marballsheepvisitsday1$Freq[marballsheepvisitsday1$Freq == 2] <- 1  
marballsheepvisitsday1$Freq[marballsheepvisitsday1$Freq == 3] <- 1  
marballsheepvisitsday1$Freq[marballsheepvisitsday1$Freq == 4] <- 1  
marballsheepvisitsday1$Freq[marballsheepvisitsday1$Freq == 5] <- 1  
marbindsheepvisitperday <- as.data.frame(tapply(marballsheepvisitsday1$Freq, list(marballsheepvisitsday1$Var2), sum))  
# View(marbindsheepvisitperday)  
marbindsheepvisitperday <- cbind(rownames(marbindsheepvisitperday), data.frame(marbindsheepvisitperday, row.names = NULL))  
marbindsheepvisitperday$datenum <- as.character(marbindsheepvisitperday$`rownames(marbindsheepvisitperday)`)  
marbindsheepvisitperday$datenum <- as.Date(marbindsheepvisitperday$`rownames(marbindsheepvisitperday)`, format = "%Y-%m-%d")  
marbindsheepvisitperday$datenum <- as.numeric(marbindsheepvisitperday$datenum)  
  
#Cutting the rows where the day wasn't complete. Inputting NA rows for uncovered days.  
marbindsheepvisitperday <- marbindsheepvisitperday[(marbindsheepvisitperday$`rownames(marbindsheepvisitperday)` != "2019-06-12") & (marbindsheepvisitperday$`rownames(marbindsheepvisitperday)` != "2019-06-18") & (marbindsheepvisitperday$`rownames(marbindsheepvisitperday)` != "2019-06-30") & (marbindsheepvisitperday$`rownames(marbindsheepvisitperday)` != "2019-07-10") & (marbindsheepvisitperday$`rownames(marbindsheepvisitperday)` != "2019-07-17") & (marbindsheepvisitperday$`rownames(marbindsheepvisitperday)` != "2019-07-25") & (marbindsheepvisitperday$`rownames(marbindsheepvisitperday)` != "2019-08-04") & (marbindsheepvisitperday$`rownames(marbindsheepvisitperday)` != "2019-08-06") & (marbindsheepvisitperday$`rownames(marbindsheepvisitperday)` != "2019-08-10"), ]  
  
marbdatena <- read.csv("TotalSheepVisitationPerDayMarbNAs122919.csv", header = TRUE, sep = ",")  
marbdatena$tapply.marballsheepvisitsday.Freq..list.marballsheepvisitsday.Var2...[marbdatena$tapply.marballsheepvisitsday.Freq..list.marballsheepvisitsday.Var2... == "#N/A"] <- NA  
names(marbdatena)[names(marbdatena) == "rownames.marbindsheepvisitperday."] <- "rownames(marbindsheepvisitperday)"  
  
names(marbindsheepvisitperday)[names(marbindsheepvisitperday) == "tapply.marballsheepvisitsday1.Freq..list.marballsheepvisitsday1.Var2..."] <- "visits"  
names(marbdatena)[names(marbdatena) == "tapply.marballsheepvisitsday.Freq..list.marballsheepvisitsday.Var2..."] <- "visits"  
  
  
marbdatena$datenum <- as.Date(marbdatena$datenum, format = "%Y-%m-%d")  
marbdatena$datenum <- as.numeric(marbdatena$datenum)  
marbindsheepvisitperday1 <- rbind(marbindsheepvisitperday, marbdatena)  
marbindsheepvisitperday1$visits <- as.integer(marbindsheepvisitperday1$visits)  
marbindsheepvisitperday1$`rownames(marbindsheepvisitperday)` <- as.Date(marbindsheepvisitperday1$`rownames(marbindsheepvisitperday)`, "%Y-%m-%d")  
  
#Barplot can't get NA spaces  
# barplot(tapply.marballsheepvisitsday.Freq..list.marballsheepvisitsday.Var2... ~ `rownames(marbindsheepvisitperday)`, data = marbindsheepvisitperday1, main = "Marbles Number of Individual Sheep Coming to Water by Day", xlab = "Date", ylab = "Number of Individual Sheep")  
# xlim = c(0, 85), width = 1  
  
# GGPLOT WORKS!  
library(ggplot2)  
plotmarbnumsheep <- ggplot(data = marbindsheepvisitperday1, aes(x = `rownames(marbindsheepvisitperday)`, y = visits, fill = "supp")) +  
 geom\_bar(stat="identity", color = "black") + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + ggtitle("Marbles") + xlab("Date") + ylab("Number of Individual Sheep") + theme(legend.position = "none") + ylim(0, 35) + geom\_hline(yintercept = 32)  
print(plotmarbnumsheep)

## Warning: Removed 48 rows containing missing values (position\_stack).



## DG note to self: Marbles when sheep died not added as line in this figure, but added in the better figure below.  
  
  
  
# SOUTH BRISTOLS  
  
#### NEED TO CHECK AND POTENTIALLY FIX ALso BHS\_1591 Teresa Seep 7/19 21:59 and 7/20 2:43 - probably need to change to Marb. -- I did change to Marbles but I'm not sure if still showing up in South Bristols data?  
  
BrsScutobs$KI\_Known\_ID <- as.character(BrsScutobs$KI\_Known\_ID)  
BrsScutobs$KI\_Known\_ID[BrsScutobs$DateObsTimeIDwaterspecific == "20190623BHS\_149910:04TalcNA"] <- "BHS\_1599"  
BrsScutobs$KI\_Known\_ID <- as.factor(BrsScutobs$KI\_Known\_ID)  
  
BrsScutobs$GO\_RangeCode <- as.character(BrsScutobs$GO\_RangeCode)  
BrsScutobs$GO\_RangeCode[BrsScutobs$DateObsTimeIDwaterspecific == "20190719BHS\_159121:59TherSeep"] <- "Marb"  
BrsScutobs$GO\_RangeCode[BrsScutobs$DateObsTimeIDwaterspecific == "20190720BHS\_15912:43TherSeep"] <- "Marb"  
BrsScutobs$GO\_RangeCode <- as.factor(BrsScutobs$GO\_RangeCode)  
# Still shows up when type in View(BrsScutobs[BrsScutobs$KI\_Known\_ID == "BHS\_1591", ]) though now says Marb -- DID I FIX IT?  
  
# This might be good. Need to check that it is in the right place in the code.  
# brstomarb <- BrsScutobs[BrsScutobs$KI\_Known\_ID == "BHS\_1591",]  
# Marbcutobs <- rbind(Marbcutobs, brstomarb)  
  
sbrssheepvisitsday <- as.data.frame(table(BrsScutobs$KI\_Known\_ID, BrsScutobs$GO\_Date))  
sbrssheepvisitsday1 <- sbrssheepvisitsday[(sbrssheepvisitsday$Var1 == "BHS\_1330") | (sbrssheepvisitsday$Var1 == "BHS\_1337") | (sbrssheepvisitsday$Var1 == "BHS\_1334") | (sbrssheepvisitsday$Var1 == "BHS\_1329") | (sbrssheepvisitsday$Var1 == "BHS\_1328") | (sbrssheepvisitsday$Var1 == "BHS\_1333") | (sbrssheepvisitsday$Var1 == "BHS\_1425") | (sbrssheepvisitsday$Var1 == "BHS\_1426") | (sbrssheepvisitsday$Var1 == "BHS\_1489") | (sbrssheepvisitsday$Var1 == "BHS\_1491") | (sbrssheepvisitsday$Var1 == "BHS\_1578") | (sbrssheepvisitsday$Var1 == "BHS\_1579") | (sbrssheepvisitsday$Var1 == "BHS\_1580") | (sbrssheepvisitsday$Var1 == "BHS\_1596") | (sbrssheepvisitsday$Var1 == "BHS\_1599") | (sbrssheepvisitsday$Var1 == "BHS\_1687") | (sbrssheepvisitsday$Var1 == "BHS\_1486") | (sbrssheepvisitsday$Var1 == "BHS\_1490") | (sbrssheepvisitsday$Var1 == "BHS\_1492") | (sbrssheepvisitsday$Var1 == "BHS\_1428") | (sbrssheepvisitsday$Var1 == "BHS\_1685") | (sbrssheepvisitsday$Var1 == "BHS\_1686") | (sbrssheepvisitsday$Var1 == "BHS\_1688") | (sbrssheepvisitsday$Var1 == "BHS\_1689"),]  
# table(sbrssheepvisitsday1$Freq) - through 5  
sbrssheepvisitsday1$Freq[sbrssheepvisitsday1$Freq == 2] <- 1  
sbrssheepvisitsday1$Freq[sbrssheepvisitsday1$Freq == 3] <- 1  
sbrssheepvisitsday1$Freq[sbrssheepvisitsday1$Freq == 4] <- 1  
sbrssheepvisitsday1$Freq[sbrssheepvisitsday1$Freq == 5] <- 1  
sbrsindsheepvisitperday <- as.data.frame(tapply(sbrssheepvisitsday1$Freq, list(sbrssheepvisitsday1$Var2), sum))  
# View(sbrsindsheepvisitperday)  
sbrsindsheepvisitperday <- cbind(rownames(sbrsindsheepvisitperday), data.frame(sbrsindsheepvisitperday, row.names = NULL))  
names(sbrsindsheepvisitperday)[names(sbrsindsheepvisitperday) == "rownames(sbrsindsheepvisitperday)"] <- "Date"  
names(sbrsindsheepvisitperday)[names(sbrsindsheepvisitperday) == "tapply.sbrssheepvisitsday1.Freq..list.sbrssheepvisitsday1.Var2..."] <- "NumIndSheep"  
sbrsindsheepvisitperday$Date <- as.Date(sbrsindsheepvisitperday$Date, format = "%Y-%m-%d")  
sbrsindsheepvisitperday1 <- sbrsindsheepvisitperday[(sbrsindsheepvisitperday$Date != "2019-06-04") & (sbrsindsheepvisitperday$Date != "2019-06-16") & (sbrsindsheepvisitperday$Date != "2019-06-18") & (sbrsindsheepvisitperday$Date != "2019-07-29") & (sbrsindsheepvisitperday$Date != "2019-08-01") & (sbrsindsheepvisitperday$Date != "2019-08-12"), ]  
sbrsdatena <- read.csv("SBrsNAIndSheepVisits1320.csv", header = TRUE, sep = ",")  
sbrsdatena$Date <- as.Date(sbrsdatena$Date, format = "%Y-%m-%d")  
sbrsdatena$NumIndSheep <- as.numeric(sbrsdatena$NumIndSheep)  
sbrsindsheepvisitperday2 <- rbind(sbrsindsheepvisitperday1, sbrsdatena)  
library(ggplot2)  
plotsbrsnumsheep <- ggplot(data = sbrsindsheepvisitperday2, aes(x = Date, y = NumIndSheep, fill = "supp")) + geom\_bar(stat = "identity", color = "black") + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + ggtitle("South Bristols") + xlab("Date") + ylab("Number of Individual Sheep") + theme(legend.position = "none") + ylim(0, 35) + geom\_hline(yintercept = 24)  
  
#Nopah  
nopaallsheepvisitsday <- as.data.frame(table(nopacutobs$KI\_Known\_ID, nopacutobs$GO\_Date))  
# View(nopaallsheepvisitsday)  
nopaallsheepvisitsday1 <- nopaallsheepvisitsday[(nopaallsheepvisitsday$Var1 == "BHS\_1737") | (nopaallsheepvisitsday$Var1 == "BHS\_1740") | (nopaallsheepvisitsday$Var1 == "BHS\_1741") | (nopaallsheepvisitsday$Var1 == "BHS\_1738") | (nopaallsheepvisitsday$Var1 == "BHS\_1736") | (nopaallsheepvisitsday$Var1 == "BHS\_1739"), ]  
# Nopah only has up to 2 visits in a day, so only need to change 2 to 1.  
nopaallsheepvisitsday1$Freq[nopaallsheepvisitsday1$Freq == 2] <- 1  
nopaindsheepvisitperday <- as.data.frame(tapply(nopaallsheepvisitsday1$Freq, list(nopaallsheepvisitsday1$Var2), sum))  
View(nopaindsheepvisitperday)  
nopaindsheepvisitperday <- cbind(rownames(nopaindsheepvisitperday), data.frame(nopaindsheepvisitperday, row.names = NULL))  
names(nopaindsheepvisitperday)[names(nopaindsheepvisitperday) == "rownames(nopaindsheepvisitperday)"] <- "Date"  
names(nopaindsheepvisitperday)[names(nopaindsheepvisitperday) == "tapply.nopaallsheepvisitsday1.Freq..list.nopaallsheepvisitsday1.Var2..."] <- "NumIndSheep"  
nopaindsheepvisitperday$Date <- as.Date(nopaindsheepvisitperday$Date, format = "%Y-%m-%d")  
# Need to input days that have zero observations, as well as the NA days.  
nopaindsheepvisitperday1 <- nopaindsheepvisitperday[(nopaindsheepvisitperday$Date != "2019-06-25") & (nopaindsheepvisitperday$Date != "2019-08-07"), ]  
nopadatena <- read.csv("NopaNAIndSheepVisits1620.csv", header= TRUE, sep = ",")  
nopadatena$Date <- as.Date(nopadatena$Date, format = "%Y-%m-%d")  
nopadatena$NumIndSheep <- as.numeric(nopadatena$NumIndSheep)  
nopaindsheepvisitperday2 <- rbind(nopaindsheepvisitperday1, nopadatena)  
library(ggplot2)  
plotnopasheepnum <- ggplot(data = nopaindsheepvisitperday2, aes(x = Date, y = NumIndSheep, fill = "supp")) + geom\_bar(stat = "identity", color = "black") + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + ggtitle("Nopah") + xlab("Date") + ylab("Number of Individual Sheep") + theme(legend.position = "none") + ylim(0, 35) + geom\_hline(yintercept = 6)  
  
# CASTLE-PIUTE  
  
cmprallsheepvisitsday <- as.data.frame(table(cmprcutobs$KI\_Known\_ID, cmprcutobs$GO\_Date))  
# Also due to the lack of coverage in the Castle Piutes along the southern stream, I limited the analysis to bighorn that located in the mine area (1730, 1733, 1734, 1728 - which doesn't have a working GPS collar).  
cmprallsheepvisitsday1 <- cmprallsheepvisitsday[(cmprallsheepvisitsday$Var1 == "BHS\_1730") | (cmprallsheepvisitsday$Var1 == "BHS\_1733") | (cmprallsheepvisitsday$Var1 == "BHS\_1734") | (cmprallsheepvisitsday$Var1 == "BHS\_1728"), ]  
# CMPR only has up to 2 visits in a day, so only need to change 2 to 1.  
cmprallsheepvisitsday1$Freq[cmprallsheepvisitsday1$Freq == 2] <- 1  
cmprindsheepvisitperday <- as.data.frame(tapply(cmprallsheepvisitsday1$Freq, list(cmprallsheepvisitsday1$Var2), sum))  
cmprindsheepvisitperday <- cbind(rownames(cmprindsheepvisitperday), data.frame(cmprindsheepvisitperday, row.names = NULL))  
names(cmprindsheepvisitperday)[names(cmprindsheepvisitperday) == "rownames(cmprindsheepvisitperday)"] <- "Date"  
names(cmprindsheepvisitperday)[names(cmprindsheepvisitperday) == "tapply.cmprallsheepvisitsday1.Freq..list.cmprallsheepvisitsday1.Var2..."] <- "NumIndSheep"  
cmprindsheepvisitperday$Date <- as.Date(cmprindsheepvisitperday$Date, format = "%Y-%m-%d")  
# There were no dates that at the end of camera periods that needed to be deleted for the Castle Piutes.  
cmprdatena <- read.csv("CmprNAIndSheepVisits1820.csv", header = TRUE, sep = ",")  
cmprdatena$Date <- as.Date(cmprdatena$Date, format = "%Y-%m-%d")  
cmprdatena$NumIndSheep <- as.numeric(cmprdatena$NumIndSheep)  
cmprindsheepvisitperday2 <- rbind(cmprindsheepvisitperday, cmprdatena)  
library(ggplot2)  
plotcmprnumsheep <- ggplot(data = cmprindsheepvisitperday2, aes(x = Date, y = NumIndSheep, fill = "supp")) + geom\_bar(stat = "identity", color = "black") + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + ggtitle("Castle Piute") + xlab("Date") + ylab("Number of Individual Sheep") + theme(legend.position = "none") + ylim(0, 35) + geom\_hline(yintercept = 4)  
  
# Getting all plots in same file  
# library(gridExtra)  
# grid.arrange(plotmarbnumsheep, plotsbrsnumsheep, plotnopasheepnum, plotcmprnumsheep, nrow = 2, top = "Number of Individual Sheep Coming to Water by Day")

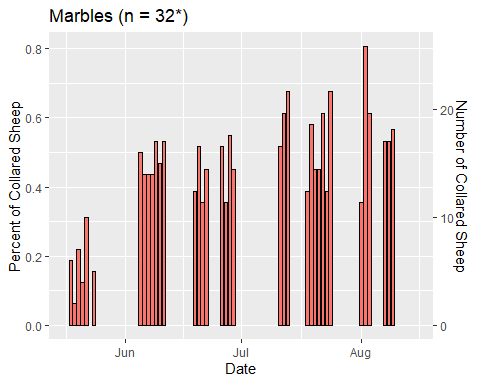
I then made figures of the percent of individual sheep coming into water by day. Given the extreme difference in the number of sheep between different mountain ranges, I also included the absolute number of sheep coming in as a secondary y axis. For the Marbles, the absolute number of collared sheep decreases towards the end of the summer due to multiple mortalities.

# MARBLES  
# Had to change the total number because sheep died.  
  
marbindsheepvisitperday1$TotalNumSheep <- NA  
marbindsheepvisitperday1$TotalNumSheep <- as.character(marbindsheepvisitperday1$TotalNumSheep)  
marbindsheepvisitperday1$TotalNumSheep[(marbindsheepvisitperday1$`rownames(marbindsheepvisitperday)` <= as.Date("2019-06-12"))] <- 32  
marbindsheepvisitperday1$TotalNumSheep[(marbindsheepvisitperday1$`rownames(marbindsheepvisitperday)` >= as.Date("2019-06-13")) & (marbindsheepvisitperday1$`rownames(marbindsheepvisitperday)` <= as.Date("2019-08-04"))] <- 31  
marbindsheepvisitperday1$TotalNumSheep[(marbindsheepvisitperday1$`rownames(marbindsheepvisitperday)` >= as.Date("2019-08-05"))] <- 30  
marbindsheepvisitperday1$TotalNumSheep <- as.numeric(marbindsheepvisitperday1$TotalNumSheep)  
  
marbindsheepvisitperday1$Percentvisited <- (marbindsheepvisitperday1$visits / marbindsheepvisitperday1$TotalNumSheep)  
library(ggplot2)  
plotmarbpercentsheep <- ggplot(data = marbindsheepvisitperday1, aes(x = `rownames(marbindsheepvisitperday)`, y = Percentvisited, fill = "supp")) + geom\_bar(stat = "identity", color = "black") + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + ggtitle("Marbles (n = 32\*)") + xlab("Date") + ylab("Percent of Sheep") + theme(legend.position = "none") + ylim(0, 1)  
plotmarbpercentsheep1 <- plotmarbpercentsheep %+% scale\_y\_continuous((name = "Percent of Collared Sheep"), sec.axis = sec\_axis(~. \* 32, name = "Number of Collared Sheep"))

## Scale for 'y' is already present. Adding another scale for 'y', which  
## will replace the existing scale.

print(plotmarbpercentsheep1)

## Warning: Removed 48 rows containing missing values (position\_stack).

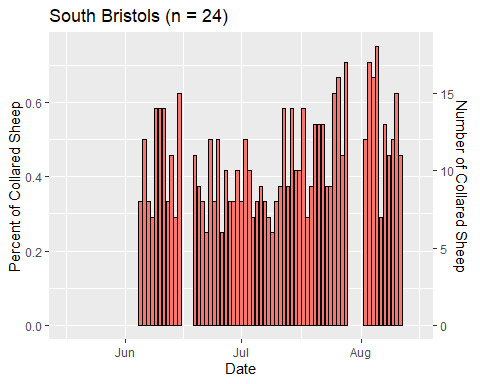


# SOUTH BRISTOLS  
sbrsindsheepvisitperday2$TotalNumSheep <- 24  
sbrsindsheepvisitperday2$Percentvisited <- (sbrsindsheepvisitperday2$NumIndSheep / sbrsindsheepvisitperday2$TotalNumSheep)  
library(ggplot2)  
plotsbrspercentsheep <- ggplot(data = sbrsindsheepvisitperday2, aes(x = Date, y = Percentvisited, fill = "supp")) + geom\_bar(stat = "identity", color = "black") + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + ggtitle("South Bristols (n = 24)") + xlab("Date") + ylab("Percent of Sheep") + theme(legend.position = "none") + ylim(0, 1)  
plotsbrspercentsheep1 <- plotsbrspercentsheep %+% scale\_y\_continuous((name = "Percent of Collared Sheep"), sec.axis = sec\_axis(~. \* 24, name = "Number of Collared Sheep"))

## Scale for 'y' is already present. Adding another scale for 'y', which  
## will replace the existing scale.

print(plotsbrspercentsheep1)

## Warning: Removed 9 rows containing missing values (position\_stack).

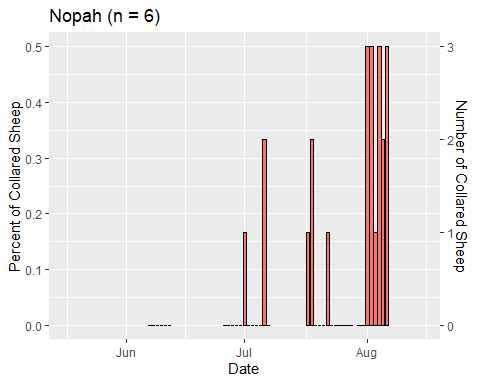


# NOPAH  
  
nopaindsheepvisitperday2$Percentvisited <- (nopaindsheepvisitperday2$NumIndSheep / 6)  
plotnopapercentsheep <- ggplot(data = nopaindsheepvisitperday2, aes(x = Date, y = Percentvisited, fill = "supp")) + geom\_bar(stat = "identity", color = "black") + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + ggtitle("Nopah (n = 6)") + xlab("Date") + ylab("Percent of Sheep") + theme(legend.position = "none") + ylim(0, 1)  
plotnopapercentsheep1 <- plotnopapercentsheep %+% scale\_y\_continuous((name = "Percent of Collared Sheep"), sec.axis = sec\_axis(~. \* 6, name = "Number of Collared Sheep"))

## Scale for 'y' is already present. Adding another scale for 'y', which  
## will replace the existing scale.

print(plotnopapercentsheep1)

## Warning: Removed 25 rows containing missing values (position\_stack).

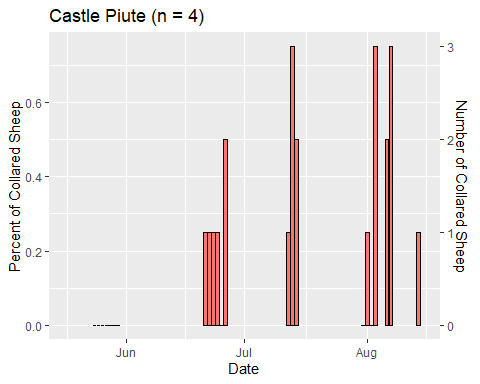


# CASTLE PIUTE  
cmprindsheepvisitperday2$Percentvisited <- (cmprindsheepvisitperday2$NumIndSheep / 4)  
plotcmprpercentsheep <- ggplot(data = cmprindsheepvisitperday2, aes(x = Date, y = Percentvisited, fill = "supp")) + geom\_bar(stat = "identity", color = "black") + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + ggtitle("Castle Piute (n = 4)") + xlab("Date") + ylab("Percent of Sheep") + theme(legend.position = "none") + ylim(0, 1)  
plotcmprpercentsheep1 <- plotcmprpercentsheep %+% scale\_y\_continuous((name = "Percent of Collared Sheep"), sec.axis = sec\_axis(~. \* 4, name = "Number of Collared Sheep"))

## Scale for 'y' is already present. Adding another scale for 'y', which  
## will replace the existing scale.

print(plotcmprpercentsheep1)

## Warning: Removed 56 rows containing missing values (position\_stack).



library(gridExtra)

## Warning: package 'gridExtra' was built under R version 3.6.2

##   
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':  
##   
## combine

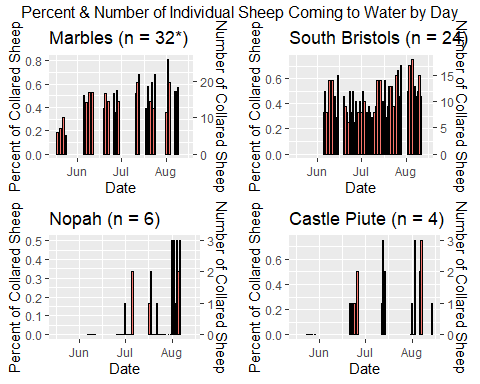
grid.arrange(plotmarbpercentsheep1, plotsbrspercentsheep1, plotnopapercentsheep1, plotcmprpercentsheep1, nrow = 2, top = "Percent & Number of Individual Sheep Coming to Water by Day")

## Warning: Removed 48 rows containing missing values (position\_stack).

## Warning: Removed 9 rows containing missing values (position\_stack).

## Warning: Removed 25 rows containing missing values (position\_stack).

## Warning: Removed 56 rows containing missing values (position\_stack).



# grid.arrange(plotmarbnumsheep, plotsbrsnumsheep, plotnopasheepnum, plotcmprnumsheep, nrow = 2, top = "Number of Individual Sheep Coming to Water by Day")

I then made figures of the mean number of visits per day by collared sheep. The error bars are the standard deviations around the mean for a given day.

# Mean Number of Visits Per Day for all sheep (and Standard Deviations)  
   
# MARBLES  
  
# I deleted the start and end days that weren't fully covered.  
Marbcutobsrd <- Marbcutobs[(Marbcutobs$GO\_Date != "2019-06-12") & (Marbcutobs$GO\_Date != "2019-06-18") & (Marbcutobs$GO\_Date != "2019-06-30") & (Marbcutobs$GO\_Date != "2019-07-10") & (Marbcutobs$GO\_Date != "2019-07-17") & (Marbcutobs$GO\_Date != "2019-07-25") & (Marbcutobs$GO\_Date != "2019-08-04") & (Marbcutobs$GO\_Date != "2019-08-06") & (Marbcutobs$GO\_Date != "2019-08-10"), ]  
  
# In order to calculate the number of visits that each sheep had to water on a given day, I duplicated the rows with multiple visits in a single observation. If an observation had three visits in it, I duplicated the row 3x.  
  
# The Marbles had no observations with multiple visits once cut to the appropriate dates.  
  
marballrd <- as.data.frame(table(Marbcutobsrd$KI\_Known\_ID, Marbcutobsrd$GO\_Date))  
  
marballrd1 <- marballrd[(marballrd$Var1 == "BHS\_1367") | (marballrd$Var1 == "BHS\_1365") | (marballrd$Var1 == "BHS\_1361") | (marballrd$Var1 == "BHS\_1362") | (marballrd$Var1 == "BHS\_1357") | (marballrd$Var1 == "BHS\_1416") | (marballrd$Var1 == "BHS\_1421") | (marballrd$Var1 == "BHS\_1581") | (marballrd$Var1 == "BHS\_1577") | (marballrd$Var1 == "BHS\_1574") | (marballrd$Var1 == "BHS\_1576") | (marballrd$Var1 == "BHS\_1583") | (marballrd$Var1 == "BHS\_1584") | (marballrd$Var1 == "BHS\_1585") | (marballrd$Var1 == "BHS\_1587") | (marballrd$Var1 == "BHS\_1582") | (marballrd$Var1 == "BHS\_1575") | (marballrd$Var1 == "BHS\_1592") | (marballrd$Var1 == "BHS\_1594") | (marballrd$Var1 == "BHS\_1420") | (marballrd$Var1 == "BHS\_1493") | (marballrd$Var1 == "BHS\_1494") | (marballrd$Var1 == "BHS\_1590") | (marballrd$Var1 == "BHS\_1591") | (marballrd$Var1 == "BHS\_1682") | (marballrd$Var1 == "BHS\_1683") | (marballrd$Var1 == "BHS\_1684") | (marballrd$Var1 == "BHS\_1366") | (marballrd$Var1 == "BHS\_1497") | (marballrd$Var1 == "BHS\_1588") | (marballrd$Var1 == "BHS\_1595") | (marballrd$Var1 == "BHS\_1589"), ]  
# Adding in sheep that never visited water all summer  
Marbinsert1497 <- read.csv("Marbinsert1497.csv", header = TRUE, sep = ",")  
marballrd1 <- rbind(marballrd1, Marbinsert1497)  
  
marbindrd <- as.data.frame(tapply(marballrd1$Freq, list(marballrd1$Var2), mean))  
marbindrd$sd <- tapply(marballrd1$Freq, list(marballrd1$Var2), sd)  
marbindrd <- cbind(rownames(marbindrd), data.frame(marbindrd, row.names = NULL))  
  
marbna2 <- read.csv("MarbNAv21120.csv", header = TRUE, sep = ",")  
marbna2$tapply.marballrd.Freq..list.marballrd.Var2.mean[marbna2$tapply.marballrd.Freq..list.marballrd.Var2.mean == "#N/A"] <- NA  
marbna2$sd.tapply.marballrd.Freq..list.marballrd.Var2...sd.[marbna2$sd.tapply.marballrd.Freq..list.marballrd.Var2...sd. == "#N/A"] <- NA  
names(marbna2)[names(marbna2) == "rownames.marbindrd."] <- "rownames(marbindrd)"  
names(marbna2)[names(marbna2) == "tapply.marballrd.Freq..list.marballrd.Var2.mean"] <- "tapply.marballrd.Freq..list.marballrd.Var2...mean."  
names(marbna2)[names(marbna2) == "sd.tapply.marballrd.Freq..list.marballrd.Var2...sd."] <- "sd"  
  
names(marbindrd)[names(marbindrd) == "tapply.marballrd1.Freq..list.marballrd1.Var2...mean."] <- "visits"  
names(marbna2)[names(marbna2) == "tapply.marballrd.Freq..list.marballrd.Var2...mean."] <- "visits"  
  
marbindrd11 <- rbind(marbindrd, marbna2)  
marbindrd11$visits <- as.numeric(marbindrd11$visits)  
marbindrd11$sd <- as.numeric(marbindrd11$sd)  
marbindrd11$`rownames(marbindrd)` <- as.Date(marbindrd11$`rownames(marbindrd)` , "%Y-%m-%d")  
library(ggplot2)  
p <- ggplot(data = marbindrd11, aes(x = `rownames(marbindrd)`, y = visits)) + geom\_point() + geom\_errorbar(aes(ymin = visits - sd, ymax = visits + sd), width = .2) + labs(title = "Marbles (n = 32\*)", x = "Date", y = "Mean Number of Visits by Day") + ylim(0, 2.5) + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + theme\_grey(base\_size = 10)  
# print(p)  
  
# An Attempt to make it boxplots ggplot(data = marbindrd11, aes(x = `rownames(marbindrd)`, y = visits)) + geom\_boxplot((aes(group = cut\_width(`rownames(marbindrd)`, 1))  
  
# SOUTH BRISTOLS  
  
BrsScutobsrd <- BrsScutobs[(BrsScutobs$GO\_Date != "2019-06-04") & (BrsScutobs$GO\_Date != "2019-06-16") & (BrsScutobs$GO\_Date != "2019-06-18") & (BrsScutobs$GO\_Date != "2019-07-29") & (BrsScutobs$GO\_Date != "2019-08-01") & (BrsScutobs$GO\_Date != "2019-08-12"), ]  
  
# In order to calculate the number of visits that each sheep had to water on a given day, I duplicated the rows with multiple visits in a single observation. If an observation had three visits in it, I duplicated the row 3x.  
  
# Divided the observations up into how many visits/observation.  
BrsSonevisitperobs <- BrsScutobsrd[(is.na(BrsScutobsrd$KI\_DG\_FirstPhotoTime) == FALSE) & (is.na(BrsScutobsrd$KI\_DG\_FirstPhotoTime2) == TRUE) & (is.na(BrsScutobsrd$KI\_DG\_FirstPhotoTime3) == TRUE) & (is.na(BrsScutobsrd$KI\_DG\_FirstPhotoTime4) == TRUE), ]  
BrsStwovisitperobs <- BrsScutobsrd[(is.na(BrsScutobsrd$KI\_DG\_FirstPhotoTime2) == FALSE) & (is.na(BrsScutobsrd$KI\_DG\_FirstPhotoTime3) == TRUE) & (is.na(BrsScutobsrd$KI\_DG\_FirstPhotoTime4) == TRUE), ]  
BrsSthreevisitperobs <- BrsScutobsrd[(is.na(BrsScutobsrd$KI\_DG\_FirstPhotoTime3) == FALSE) & (is.na(BrsScutobsrd$KI\_DG\_FirstPhotoTime4) == TRUE),]  
BrsSfourvisitperobs <- BrsScutobsrd[(is.na(BrsScutobsrd$KI\_DG\_FirstPhotoTime4) == FALSE) ,]  
#Duplicated the rows  
BrsStwovisitdup <- do.call("rbind", replicate(2, BrsStwovisitperobs, simplify = FALSE))  
BrsSthreevisitdup <- do.call("rbind", replicate(3, BrsSthreevisitperobs, simplify = FALSE))  
BrsSfourvisitdup <- do.call("rbind", replicate(4, BrsSfourvisitperobs, simplify = FALSE))  
# Combining these duplicated dataframes  
BrsScutobsrdwithdups <- rbind(BrsSonevisitperobs, BrsStwovisitdup, BrsSthreevisitdup, BrsSfourvisitdup)  
sbrsallrd <- as.data.frame(table(BrsScutobsrdwithdups$KI\_Known\_ID, BrsScutobsrdwithdups$GO\_Date))  
# I double-checked that the table function did not remove the duplicates before calculation. It didn't, so this worked.  
# tocompare <- as.data.frame(table(BrsScutobsrd$KI\_Known\_ID, BrsScutobsrd$GO\_Date))  
sbrsallrd1 <- sbrsallrd[(sbrsallrd$Var1 == "BHS\_1330") | (sbrsallrd$Var1 == "BHS\_1337") | (sbrsallrd$Var1 == "BHS\_1334") | (sbrsallrd$Var1 == "BHS\_1329") | (sbrsallrd$Var1 == "BHS\_1328") | (sbrsallrd$Var1 == "BHS\_1333") | (sbrsallrd$Var1 == "BHS\_1425") | (sbrsallrd$Var1 == "BHS\_1426") | (sbrsallrd$Var1 == "BHS\_1489") | (sbrsallrd$Var1 == "BHS\_1491") | (sbrsallrd$Var1 == "BHS\_1578") | (sbrsallrd$Var1 == "BHS\_1579") | (sbrsallrd$Var1 == "BHS\_1580") | (sbrsallrd$Var1 == "BHS\_1596") | (sbrsallrd$Var1 == "BHS\_1599") | (sbrsallrd$Var1 == "BHS\_1687") | (sbrsallrd$Var1 == "BHS\_1486") | (sbrsallrd$Var1 == "BHS\_1490") | (sbrsallrd$Var1 == "BHS\_1492") | (sbrsallrd$Var1 == "BHS\_1428") | (sbrsallrd$Var1 == "BHS\_1685") | (sbrsallrd$Var1 == "BHS\_1686") | (sbrsallrd$Var1 == "BHS\_1688") | (sbrsallrd$Var1 == "BHS\_1689"), ]  
sbrsindrd <- as.data.frame(tapply(sbrsallrd1$Freq, list(sbrsallrd1$Var2), mean))  
sbrsindrd$sd <- tapply(sbrsallrd1$Freq, list(sbrsallrd1$Var2), sd)  
sbrsindrd <- cbind(rownames(sbrsindrd), data.frame(sbrsindrd, row.names = NULL))  
names(sbrsindrd)[names(sbrsindrd) == "rownames(sbrsindrd)"] <- "Date"  
names(sbrsindrd)[names(sbrsindrd) == "tapply.sbrsallrd1.Freq..list.sbrsallrd1.Var2...mean."] <- "Meannumvisits"  
sbrsna2 <- read.csv("SBrsNAv21520.csv", header = TRUE, sep = ",")  
sbrsrd11 <- rbind(sbrsindrd, sbrsna2)  
sbrsrd11$Meannumvisits <- as.numeric(sbrsrd11$Meannumvisits)  
sbrsrd11$sd <- as.numeric(sbrsrd11$sd)  
sbrsrd11$Date <- as.Date(sbrsrd11$Date, "%Y-%m-%d")  
library(ggplot2)  
sbrsrd11plot <- ggplot(data = sbrsrd11, aes(x = Date, y = Meannumvisits)) + geom\_point() + geom\_errorbar(aes(ymin = Meannumvisits - sd, ymax = Meannumvisits + sd), width = .2) + labs(title = "South Bristols (n = 24)", x = "Date", y = "Mean Number of Visits by Day") + ylim(0, 2.5) + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + theme\_grey(base\_size = 10)  
# print(sbrsrd11plot)  
  
#NOPAH  
  
nopacutobsrd <- nopacutobs[(nopacutobs$GO\_Date != "2019-06-06") & (nopacutobs$GO\_Date != "2019-06-13") & (nopacutobs$GO\_Date != "2019-06-25") & (nopacutobs$GO\_Date != "2019-07-08") & (nopacutobs$GO\_Date != "2019-07-16") & (nopacutobs$GO\_Date != "2019-07-29") & (nopacutobs$GO\_Date != "2019-08-07"), ]  
nopaallrd <- as.data.frame(table(nopacutobsrd$KI\_Known\_ID, nopacutobsrd$GO\_Date))  
nopaallrd1 <- nopaallrd[(nopaallrd$Var1 == "BHS\_1737") | (nopaallrd$Var1 == "BHS\_1740") | (nopaallrd$Var1 == "BHS\_1741") | (nopaallrd$Var1 == "BHS\_1738") | (nopaallrd$Var1 == "BHS\_1736") | (nopaallrd$Var1 == "BHS\_1739"), ]  
nopaindrd <- as.data.frame(tapply(nopaallrd1$Freq, list(nopaallrd1$Var2), mean))  
nopaindrd$sd <- tapply(nopaallrd1$Freq, list(nopaallrd1$Var2), sd)  
nopaindrd <- cbind(rownames(nopaindrd), data.frame(nopaindrd, row.names = NULL))  
names(nopaindrd)[names(nopaindrd) == "rownames(nopaindrd)"] <- "Date"  
names(nopaindrd)[names(nopaindrd) == "tapply.nopaallrd1.Freq..list.nopaallrd1.Var2...mean."] <- "Meannumvisits"  
# I then needed to add in both the NA time periods and the mean = zero time periods.  
nopana2 <- read.csv("NopaNAZerov21620.csv", header = TRUE, sep = ",")  
nopard11 <- rbind(nopaindrd, nopana2)  
nopard11$Meannumvisits <- as.numeric(nopard11$Meannumvisits)  
nopard11$sd <- as.numeric(nopard11$sd)  
nopard11$Date <- as.Date(nopard11$Date, "%Y-%m-%d")  
library(ggplot2)  
nopard11plot <- ggplot(data = nopard11, aes(x = Date, y = Meannumvisits)) + geom\_point() + geom\_errorbar(aes(ymin = Meannumvisits - sd, ymax = Meannumvisits + sd), width = 0.2) + labs(title = "Nopah (n = 6)", x = "Date", y = "Mean Number of Visits by Day") + ylim(0, 2.5) + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + theme\_grey(base\_size = 10)  
# print(nopard11plot)  
  
# CASTLE PIUTE  
  
# Cmpr data already good for start and end dates based on decisions made while making the previous figures.  
cmprallrd <- as.data.frame(table(cmprcutobs$KI\_Known\_ID, cmprcutobs$GO\_Date))  
cmprallrd1 <- cmprallrd[(cmprallrd$Var1 == "BHS\_1730") | (cmprallrd$Var1 == "BHS\_1733") | (cmprallrd$Var1 == "BHS\_1734") | (cmprallrd$Var1 == "BHS\_1728"), ]  
cmprindrd <- as.data.frame(tapply(cmprallrd1$Freq, list(cmprallrd1$Var2), mean))  
cmprindrd$sd <- tapply(cmprallrd1$Freq, list(cmprallrd1$Var2), sd)  
cmprindrd <- cbind(rownames(cmprindrd), data.frame(cmprindrd, row.names = NULL))  
names(cmprindrd)[names(cmprindrd) == "rownames(cmprindrd)"] <- "Date"  
names(cmprindrd)[names(cmprindrd) == "tapply.cmprallrd1.Freq..list.cmprallrd1.Var2...mean."] <- "Meannumvisits"  
cmprna2 <- read.csv("CmprNAZerov21820.csv", header = TRUE, sep = ",")  
cmprrd11 <- rbind(cmprindrd, cmprna2)  
cmprrd11$Meannumvisits <- as.numeric(cmprrd11$Meannumvisits)  
cmprrd11$sd <- as.numeric(cmprrd11$sd)  
cmprrd11$Date <- as.Date(cmprrd11$Date, "%Y-%m-%d")  
library(ggplot2)  
cmprrd11plot <- ggplot(data = cmprrd11, aes(x = Date, y = Meannumvisits)) + geom\_point() + geom\_errorbar(aes(ymin = Meannumvisits - sd, ymax = Meannumvisits + sd), width = 0.2) + labs(title = "Castle Piute (n = 4)", x = "Date", y = "Mean Number of Visits by Day") + ylim(0, 2.5) + scale\_x\_date(limits = as.Date(c("2019-05-17", "2019-08-15"))) + theme\_grey(base\_size = 10)  
# print(cmprrd11plot)  
  
library(gridExtra)  
grid.arrange(p, sbrsrd11plot, nopard11plot, cmprrd11plot, nrow = 2, top = "Mean Number of Visits Per Day Given All Collared Sheep")

## Warning: Removed 48 rows containing missing values (geom\_point).

## Warning: Removed 90 rows containing missing values (geom\_errorbar).

## Warning: Removed 9 rows containing missing values (geom\_point).

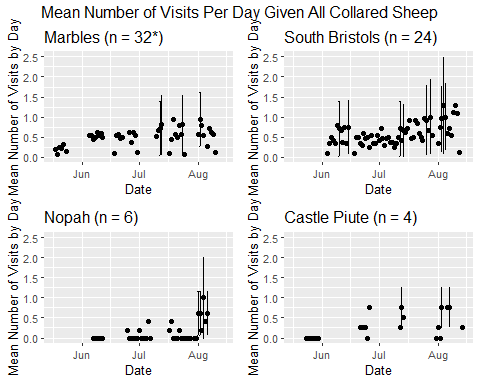
## Warning: Removed 66 rows containing missing values (geom\_errorbar).

## Warning: Removed 25 rows containing missing values (geom\_point).

## Warning: Removed 33 rows containing missing values (geom\_errorbar).

## Warning: Removed 56 rows containing missing values (geom\_point).

## Warning: Removed 66 rows containing missing values (geom\_errorbar).

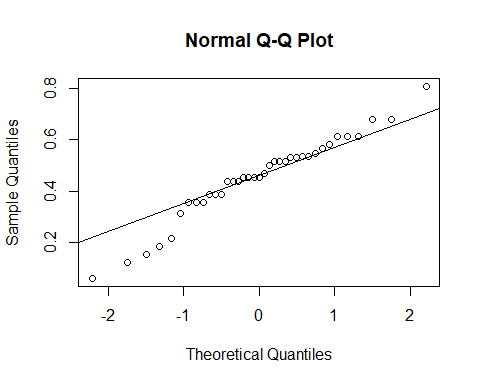


I then ran statistics associated with the previous figures. In the Marbles, there was a significant difference in the percent of collared sheep visiting water on a given day by month between May and June/July/August, though not among June, July, and August. In the South Bristols, there was a significant difference between July and August, and between June and August, thought not between June and July. In the Nopahs, June and July were not significant different, though they were both different from August. In the Castle Piutes, the only significant differences between months were between May and July, and May and August. Whether two months were significantly different on the mean number of visits to water on a given day was exactly the same as whether two months were significantly different in the percent of collared sheep visiting on a given day by month. To me, this suggests that the number of visits was not significantly different between different individual sheep.

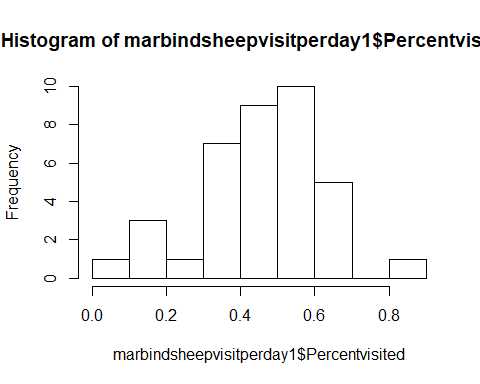
\*\*\*The figures of the ‘Mean Number of Visits to Water on a Given Day by Month’ and ‘Percent of Individual Sheep that Visited Water on a Given Day by Month’ are exceedingly similar. When I double-checked how I calculated these, I didn’t find a problem. Can someone double check me? If this is true, it suggests that there was basically no different in the number of visits to water between different sheep…

There was a significant difference in the percentage of collared sheep coming into water between mountain ranges. All populations were significantly different from each other except the Marbles from the South Bristols. There was also a significant difference in the number of visitations to water between mountain ranges. All populations were significantly different from each other except the Marbles and the South Bristols, and the Castle Piutes and Nopah.

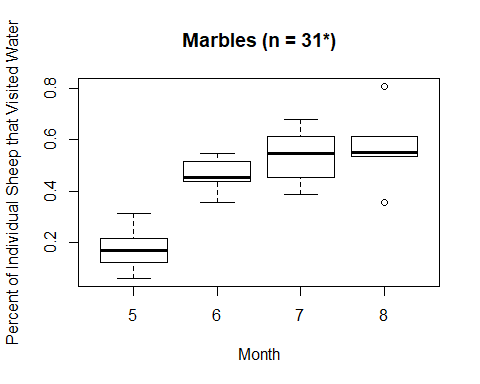
## whether there is a significant difference in the percentage of collared sheep visiting water among months within each mountain range ##  
  
# Marbles  
marbindsheepvisitperday1$Month <- month(marbindsheepvisitperday1$`rownames(marbindsheepvisitperday)`)  
qqnorm(marbindsheepvisitperday1$Percentvisited)  
qqline(marbindsheepvisitperday1$Percentvisited)



hist(marbindsheepvisitperday1$Percentvisited)



# slightly left skewed?  
boxplot(Percentvisited ~ Month, data = marbindsheepvisitperday1, main = "Marbles (n = 31\*)", ylab = "Percent of Individual Sheep that Visited Water")



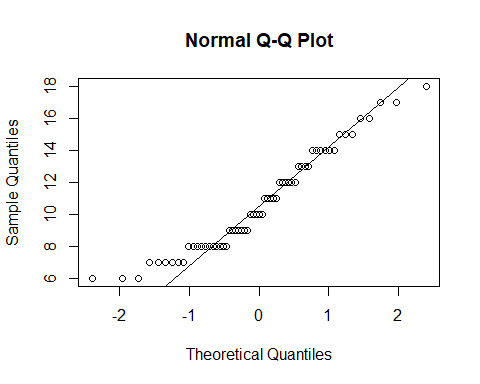
marbindsheepvisitperday1$Month <- as.factor(marbindsheepvisitperday1$Month)  
PercentMonthMarb <- aov(formula = Percentvisited ~ Month, data = marbindsheepvisitperday1)  
summary.lm(PercentMonthMarb)

##   
## Call:  
## aov(formula = Percentvisited ~ Month, data = marbindsheepvisitperday1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.21308 -0.05208 -0.01001 0.05450 0.23853   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.17708 0.03963 4.468 8.75e-05 \*\*\*  
## Month6 0.28454 0.04689 6.068 7.90e-07 \*\*\*  
## Month7 0.35840 0.05013 7.150 3.43e-08 \*\*\*  
## Month8 0.39084 0.05605 6.974 5.67e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.09707 on 33 degrees of freedom  
## (48 observations deleted due to missingness)  
## Multiple R-squared: 0.6606, Adjusted R-squared: 0.6297   
## F-statistic: 21.41 on 3 and 33 DF, p-value: 6.992e-08

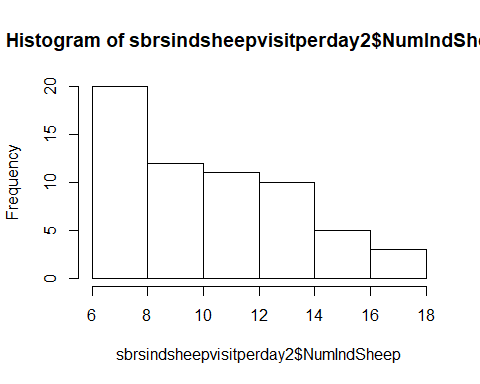
TukeyHSD(PercentMonthMarb)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Percentvisited ~ Month, data = marbindsheepvisitperday1)  
##   
## $Month  
## diff lwr upr p adj  
## 6-5 0.28454301 0.15770470 0.4113813 0.0000046  
## 7-5 0.35840054 0.22280468 0.4939964 0.0000002  
## 8-5 0.39083781 0.23923703 0.5424386 0.0000003  
## 7-6 0.07385753 -0.03334041 0.1810555 0.2630005  
## 8-6 0.10629480 -0.02054351 0.2331331 0.1267101  
## 8-7 0.03243728 -0.10315858 0.1680331 0.9158158

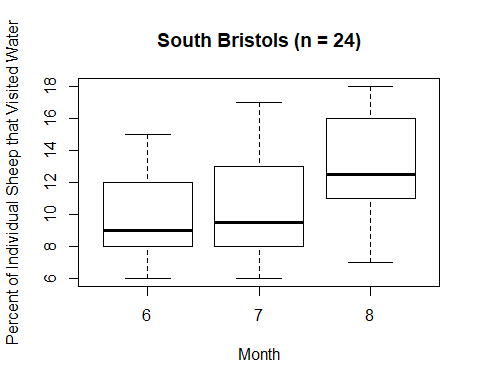
# $Month  
# diff lwr upr p adj  
# 6-5 0.29677419 0.16952135 0.4240270 0.0000023  
# 7-5 0.36344086 0.22740185 0.4994799 0.0000002  
# 8-5 0.38709677 0.23500054 0.5391930 0.0000004  
# 7-6 0.06666667 -0.04088161 0.1742149 0.3517407  
# 8-6 0.09032258 -0.03693026 0.2175754 0.2394620  
# 8-7 0.02365591 -0.11238310 0.1596949 0.9650476  
  
# South Bristols  
library(lubridate)  
sbrsindsheepvisitperday2$Month <- month(sbrsindsheepvisitperday2$Date)  
#Having trouble getting this to show up 'inline' in Markdown, but qq plot looks good.  
qqnorm(sbrsindsheepvisitperday2$NumIndSheep)  
qqline(sbrsindsheepvisitperday2$NumIndSheep)



hist(sbrsindsheepvisitperday2$NumIndSheep)



boxplot(NumIndSheep ~ Month, data = sbrsindsheepvisitperday2, main = "South Bristols (n = 24)", ylab = "Percent of Individual Sheep that Visited Water")



bartlett.test(NumIndSheep ~ Month, data = sbrsindsheepvisitperday2)

##   
## Bartlett test of homogeneity of variances  
##   
## data: NumIndSheep by Month  
## Bartlett's K-squared = 0.5211, df = 2, p-value = 0.7706

# p value = 0.7447  
sbrsindsheepvisitperday2$Month <- as.factor(sbrsindsheepvisitperday2$Month)  
NISMonth <- aov(NumIndSheep ~ Month, data = sbrsindsheepvisitperday2)  
summary.lm(NISMonth)

##   
## Call:  
## aov(formula = NumIndSheep ~ Month, data = sbrsindsheepvisitperday2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -6.2000 -1.8261 -0.8261 2.4643 6.4643   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 9.8261 0.6194 15.863 < 2e-16 \*\*\*  
## Month7 0.7096 0.8360 0.849 0.39946   
## Month8 3.3739 1.1253 2.998 0.00399 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.971 on 58 degrees of freedom  
## (9 observations deleted due to missingness)  
## Multiple R-squared: 0.1364, Adjusted R-squared: 0.1066   
## F-statistic: 4.579 on 2 and 58 DF, p-value: 0.01424

# p-value = 0.01271  
TukeyHSD(NISMonth)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = NumIndSheep ~ Month, data = sbrsindsheepvisitperday2)  
##   
## $Month  
## diff lwr upr p adj  
## 7-6 0.7096273 -1.30121987 2.720475 0.6743918  
## 8-6 3.3739130 0.66727633 6.080550 0.0109783  
## 8-7 2.6642857 0.03189895 5.296672 0.0466810

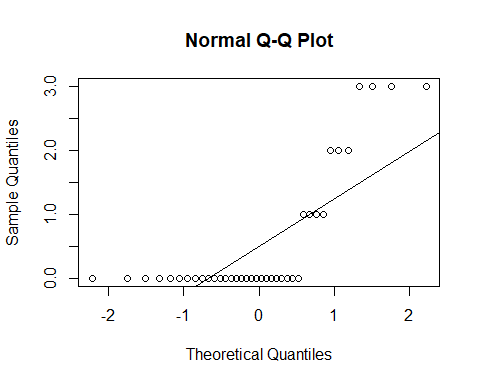
# diff lwr upr p adj  
# 7-6 0.7531056 -1.25106927 2.757280 0.6400796  
# 8-6 3.4173913 0.71973569 6.115047 0.0095915  
# 8-7 2.6642857 0.04063367 5.287938 0.0457927  
# South Bristols significant  
  
PercentMonthsbrs <- aov(formula = Percentvisited ~ Month, data = sbrsindsheepvisitperday2)  
summary.lm(PercentMonthsbrs)

##   
## Call:  
## aov(formula = Percentvisited ~ Month, data = sbrsindsheepvisitperday2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.25833 -0.07609 -0.03442 0.10268 0.26935   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.40942 0.02581 15.863 < 2e-16 \*\*\*  
## Month7 0.02957 0.03483 0.849 0.39946   
## Month8 0.14058 0.04689 2.998 0.00399 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1238 on 58 degrees of freedom  
## (9 observations deleted due to missingness)  
## Multiple R-squared: 0.1364, Adjusted R-squared: 0.1066   
## F-statistic: 4.579 on 2 and 58 DF, p-value: 0.01424

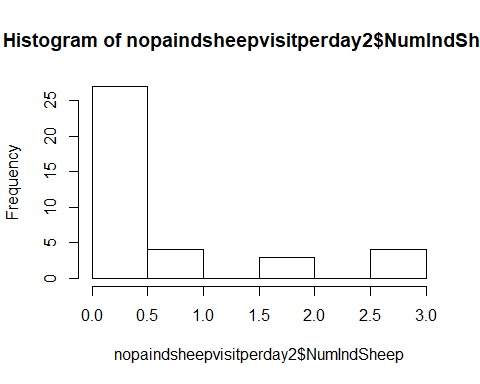
TukeyHSD(PercentMonthsbrs)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Percentvisited ~ Month, data = sbrsindsheepvisitperday2)  
##   
## $Month  
## diff lwr upr p adj  
## 7-6 0.02956781 -0.054217495 0.1133531 0.6743918  
## 8-6 0.14057971 0.027803180 0.2533562 0.0109783  
## 8-7 0.11101190 0.001329123 0.2206947 0.0466810

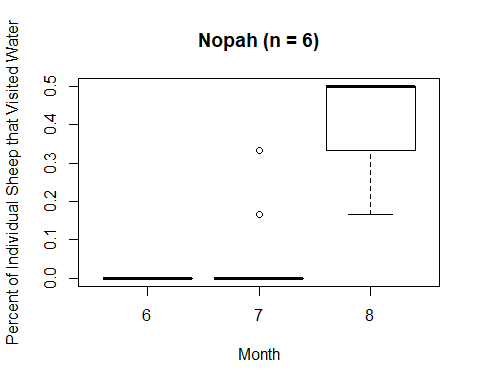
# diff lwr upr p adj  
# 7-6 0.02789280 -0.046335899 0.1021215 0.6400796  
# 8-6 0.12657005 0.026656877 0.2264832 0.0095915  
# 8-7 0.09867725 0.001504951 0.1958495 0.0457927  
  
# ANOVA on actual numbers and percentages yields the same exact result.  
  
# Nopah - very few sheep so results may not mean much?  
nopaindsheepvisitperday2$Month <- month(nopaindsheepvisitperday2$Date)  
qqnorm(nopaindsheepvisitperday2$NumIndSheep)  
qqline(nopaindsheepvisitperday2$NumIndSheep)



hist(nopaindsheepvisitperday2$NumIndSheep)



# right-skewed b/c very small sample size  
boxplot(Percentvisited ~ Month, data = nopaindsheepvisitperday2, main = "Nopah (n = 6)", ylab = "Percent of Individual Sheep that Visited Water")



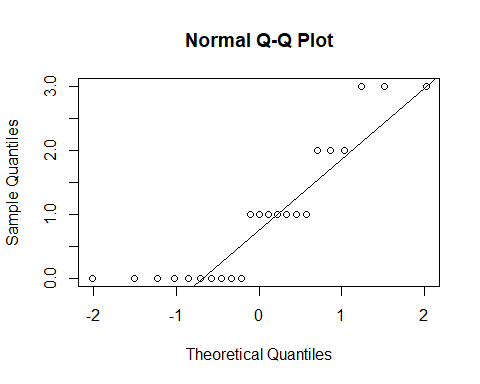
nopaindsheepvisitperday2$Month <- as.factor(nopaindsheepvisitperday2$Month)  
PercentMonthNopa <- aov(formula = Percentvisited ~ Month, data = nopaindsheepvisitperday2)  
summary.lm(PercentMonthNopa)

##   
## Call:  
## aov(formula = Percentvisited ~ Month, data = nopaindsheepvisitperday2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.25000 -0.05556 0.00000 0.00000 0.27778   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.981e-16 2.963e-02 0.000 1.000   
## Month7 5.556e-02 3.657e-02 1.519 0.138   
## Month8 4.167e-01 4.987e-02 8.355 7.48e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.09827 on 35 degrees of freedom  
## (25 observations deleted due to missingness)  
## Multiple R-squared: 0.6901, Adjusted R-squared: 0.6724   
## F-statistic: 38.97 on 2 and 35 DF, p-value: 1.247e-09

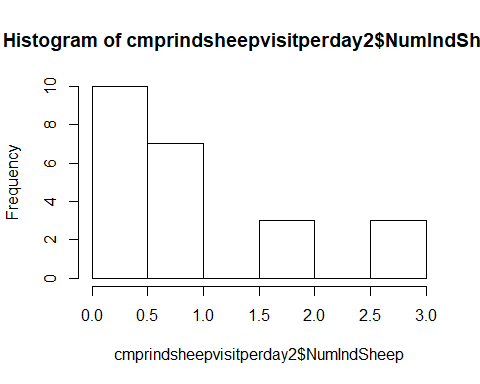
# p-value: 1.247e-09  
TukeyHSD(PercentMonthNopa)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Percentvisited ~ Month, data = nopaindsheepvisitperday2)  
##   
## $Month  
## diff lwr upr p adj  
## 7-6 0.05555556 -0.0339505 0.1450616 0.2945005  
## 8-6 0.41666667 0.2946172 0.5387161 0.0000000  
## 8-7 0.36111111 0.2497894 0.4724329 0.0000000

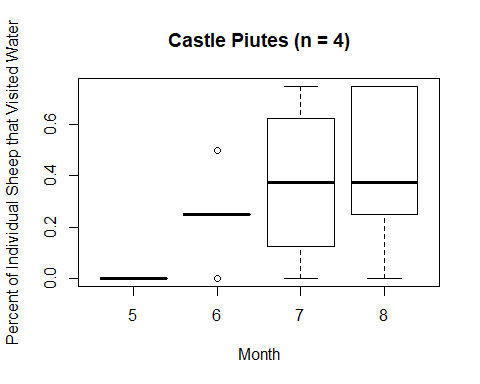
# diff lwr upr p adj  
# 7-6 0.05555556 -0.0339505 0.1450616 0.2945005  
# 8-6 0.41666667 0.2946172 0.5387161 0.0000000  
# 8-7 0.36111111 0.2497894 0.4724329 0.0000000  
  
# Castle Piutes  
cmprindsheepvisitperday2$Month <- month(cmprindsheepvisitperday2$Date)  
qqnorm(cmprindsheepvisitperday2$NumIndSheep)  
qqline(cmprindsheepvisitperday2$NumIndSheep)



hist(cmprindsheepvisitperday2$NumIndSheep)



# right-skewed b/c very small sample size  
boxplot(Percentvisited ~ Month, data = cmprindsheepvisitperday2, main = "Castle Piutes (n = 4)", ylab = "Percent of Individual Sheep that Visited Water")



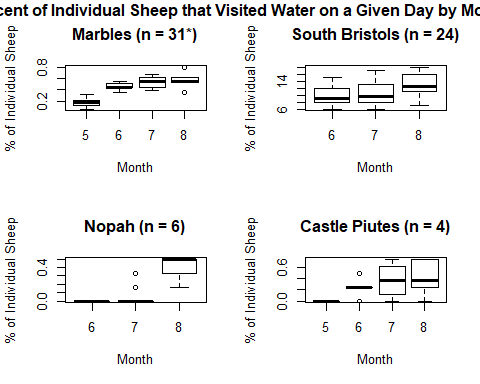
cmprindsheepvisitperday2$Month <- as.factor(cmprindsheepvisitperday2$Month)  
PercentMonthCMPR <- aov(formula = Percentvisited ~ Month, data = cmprindsheepvisitperday2)  
summary.lm(PercentMonthCMPR)

##   
## Call:  
## aov(formula = Percentvisited ~ Month, data = cmprindsheepvisitperday2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.41667 -0.06250 0.00000 0.04167 0.37500   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.00000 0.08207 0.000 1.00000   
## Month6 0.25000 0.12080 2.069 0.05238 .   
## Month7 0.37500 0.13610 2.755 0.01259 \*   
## Month8 0.41667 0.12080 3.449 0.00269 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2171 on 19 degrees of freedom  
## (56 observations deleted due to missingness)  
## Multiple R-squared: 0.4257, Adjusted R-squared: 0.335   
## F-statistic: 4.694 on 3 and 19 DF, p-value: 0.01289

# p-value: 0.01747  
TukeyHSD(PercentMonthCMPR)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Percentvisited ~ Month, data = cmprindsheepvisitperday2)  
##   
## $Month  
## diff lwr upr p adj  
## 6-5 0.25000000 -0.089684221 0.5896842 0.1987704  
## 7-5 0.37500000 -0.007688728 0.7576887 0.0560011  
## 8-5 0.41666667 0.076982445 0.7563509 0.0131113  
## 7-6 0.12500000 -0.269114780 0.5191148 0.8091270  
## 8-6 0.16666667 -0.185840309 0.5191736 0.5563724  
## 8-7 0.04166667 -0.352448113 0.4357814 0.9905641

# diff lwr upr p adj  
# 6-5 0.25000000 -0.147049891 0.6470499 0.3173379  
# 7-5 0.43750000 -0.009816973 0.8848170 0.0565871  
# 8-5 0.45833333 0.061283442 0.8553832 0.0202925  
# 7-6 0.18750000 -0.273172651 0.6481727 0.6675089  
# 8-6 0.20833333 -0.203704812 0.6203715 0.5018073  
# 8-7 0.02083333 -0.439839317 0.4815060 0.9992381  
  
  
par(mfrow = c(2,2))  
#title = "Percent of Individual Sheep that Visited Water by Month")  
boxplot(Percentvisited ~ Month, data = marbindsheepvisitperday1, main = "Marbles (n = 31\*)", ylab = "% of Individual Sheep")  
boxplot(NumIndSheep ~ Month, data = sbrsindsheepvisitperday2, main = "South Bristols (n = 24)", ylab = "% of Individual Sheep")  
boxplot(Percentvisited ~ Month, data = nopaindsheepvisitperday2, main = "Nopah (n = 6)", ylab = "% of Individual Sheep")  
boxplot(Percentvisited ~ Month, data = cmprindsheepvisitperday2, main = "Castle Piutes (n = 4)", ylab = "% of Individual Sheep")  
title("Percent of Individual Sheep that Visited Water on a Given Day by Month", line = -1, outer = TRUE)



dev.off()

## null device   
## 1

## whether there is a significant difference in the mean number of visitations of collared sheep visiting water among months within each mountain range ##  
  
  
library(lubridate)  
marbindrd11$Month <- month(marbindrd11$`rownames(marbindrd)`)  
qqnorm(marbindrd11$visits)  
qqline(marbindrd11$visits)  
hist(marbindrd11$visits)  
boxplot(visits ~ Month, data = marbindrd11, main = "Marbles (n = 32\*)", ylab = "Mean Number of Visits")  
marbindrd11$Month <- as.factor(marbindrd11$Month)  
MeanVisitsMonthMarb <- aov(formula = visits ~ Month, data = marbindrd11)  
summary.lm(MeanVisitsMonthMarb)

##   
## Call:  
## aov(formula = visits ~ Month, data = marbindrd11)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.50923 -0.04389 0.01935 0.11384 0.36828   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.19792 0.08321 2.378 0.022009 \*   
## Month6 0.28422 0.09609 2.958 0.005067 \*\*   
## Month7 0.37583 0.10060 3.736 0.000558 \*\*\*  
## Month8 0.37130 0.10743 3.456 0.001266 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2038 on 42 degrees of freedom  
## (48 observations deleted due to missingness)  
## Multiple R-squared: 0.2746, Adjusted R-squared: 0.2228   
## F-statistic: 5.299 on 3 and 42 DF, p-value: 0.003444

# p-value: 0.0002828  
TukeyHSD(MeanVisitsMonthMarb)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = visits ~ Month, data = marbindrd11)  
##   
## $Month  
## diff lwr upr p adj  
## 6-5 0.284218190 0.02719465 0.5412417 0.0251172  
## 7-5 0.375827130 0.10673025 0.6449240 0.0030207  
## 8-5 0.371303763 0.08394271 0.6586648 0.0066836  
## 7-6 0.091608940 -0.10684150 0.2900594 0.6086377  
## 8-6 0.087085573 -0.13550334 0.3096745 0.7232874  
## 8-7 -0.004523366 -0.24095064 0.2319039 0.9999511

# diff lwr upr p adj  
# 6-5 0.30213934 0.07318684 0.5310918 0.0054388  
# 7-5 0.39552316 0.15581593 0.6352304 0.0003925  
# 8-5 0.41789875 0.16192207 0.6738754 0.0004537  
# 7-6 0.09338382 -0.08339269 0.2701603 0.4984561  
# 8-6 0.11575941 -0.08251927 0.3140381 0.4110441  
# 8-7 0.02237559 -0.18823008 0.2329813 0.9918760  
  
# South Bristols  
sbrsrd11$Month <- month(sbrsrd11$Date)  
qqnorm(sbrsrd11$Meannumvisits)  
qqline(sbrsrd11$Meannumvisits)  
hist(sbrsrd11$Meannumvisits)  
boxplot(Meannumvisits ~ Month, data = sbrsrd11, main = "South Bristols (n = 24)", ylab = "Mean Number of Visits")  
sbrsrd11$Month <- as.factor(sbrsrd11$Month)  
MeanVisitsMonthSbrs <- aov(formula = Meannumvisits ~ Month, data = sbrsrd11)  
summary.lm(MeanVisitsMonthSbrs)

##   
## Call:  
## aov(formula = Meannumvisits ~ Month, data = sbrsrd11)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.69097 -0.12787 -0.03977 0.13969 0.47569   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.45167 0.04837 9.339 1.69e-13 \*\*\*  
## Month7 0.13454 0.06600 2.039 0.0457 \*   
## Month8 0.36431 0.08493 4.290 6.27e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2418 on 63 degrees of freedom  
## (9 observations deleted due to missingness)  
## Multiple R-squared: 0.2268, Adjusted R-squared: 0.2022   
## F-statistic: 9.238 on 2 and 63 DF, p-value: 0.0003033

# p-value: 0.001847  
TukeyHSD(MeanVisitsMonthSbrs)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Meannumvisits ~ Month, data = sbrsrd11)  
##   
## $Month  
## diff lwr upr p adj  
## 7-6 0.1345402 -0.02387736 0.2929578 0.1114116  
## 8-6 0.3643056 0.16045318 0.5681579 0.0001834  
## 8-7 0.2297653 0.03052472 0.4290059 0.0199562

# 7-6 0.1035732 -0.05770139 0.2648478 0.2784047  
# 8-6 0.3200483 0.11436138 0.5257352 0.0011822  
# 8-7 0.2164751 0.01821764 0.4147325 0.0291954  
  
#Nopah  
nopard11$Month <- month(nopard11$Date)  
qqnorm(nopard11$Meannumvisits)  
qqline(nopard11$Meannumvisits)  
hist(nopard11$Meannumvisits)  
boxplot(Meannumvisits ~ Month, data = nopard11, main = "Nopah (n = 6)", ylab = "Mean Number of Visits")  
nopard11$Month <- as.factor(nopard11$Month)  
MeanVisitsMonthNopa <- aov(formula = Meannumvisits ~ Month, data = nopard11)  
summary.lm(MeanVisitsMonthNopa)

##   
## Call:  
## aov(formula = Meannumvisits ~ Month, data = nopard11)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.36667 -0.06667 -0.01667 0.00833 0.43333   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.01667 0.04129 0.404 0.689   
## Month7 0.05000 0.05177 0.966 0.341   
## Month8 0.55000 0.07152 7.690 4.18e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.143 on 36 degrees of freedom  
## (25 observations deleted due to missingness)  
## Multiple R-squared: 0.6523, Adjusted R-squared: 0.633   
## F-statistic: 33.78 on 2 and 36 DF, p-value: 5.503e-09

# p-value: 3.149e-09  
TukeyHSD(MeanVisitsMonthNopa)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Meannumvisits ~ Month, data = nopard11)  
##   
## $Month  
## diff lwr upr p adj  
## 7-6 0.05 -0.07653042 0.1765304 0.6028115  
## 8-6 0.55 0.37517310 0.7248269 0.0000000  
## 8-7 0.50 0.33814174 0.6618583 0.0000000

# 7-6 0.0500000 -0.07666386 0.1766639 0.6040199  
# 8-6 0.5261905 0.35972154 0.6926594 0.0000000  
# 8-7 0.4761905 0.32342811 0.6289528 0.0000000  
  
#Castle Piutes  
cmprrd11$Month <- month(cmprrd11$Date)  
qqnorm(cmprrd11$Meannumvisits)  
qqline(cmprrd11$Meannumvisits)  
hist(cmprrd11$Meannumvisits)  
boxplot(Meannumvisits ~ Month, data = cmprrd11, main = "Castle Piutes (n = 4)", ylab = "Mean Number of Visits")  
cmprrd11$Month <- as.factor(cmprrd11$Month)  
MeanVisitsMonthCmpr <- aov(formula = Meannumvisits ~ Month, data = cmprrd11)  
summary.lm(MeanVisitsMonthCmpr)

##   
## Call:  
## aov(formula = Meannumvisits ~ Month, data = cmprrd11)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.45833 -0.08333 0.00000 0.06250 0.45833   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.157e-17 9.366e-02 0.000 1.00000   
## Month6 2.917e-01 1.379e-01 2.116 0.04781 \*   
## Month7 3.750e-01 1.553e-01 2.414 0.02601 \*   
## Month8 4.583e-01 1.379e-01 3.325 0.00356 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2478 on 19 degrees of freedom  
## (56 observations deleted due to missingness)  
## Multiple R-squared: 0.397, Adjusted R-squared: 0.3018   
## F-statistic: 4.17 on 3 and 19 DF, p-value: 0.0199

# p-value: 0.02232  
TukeyHSD(MeanVisitsMonthCmpr)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Meannumvisits ~ Month, data = cmprrd11)  
##   
## $Month  
## diff lwr upr p adj  
## 6-5 0.29166667 -0.09597920 0.6793125 0.1839315  
## 7-5 0.37500000 -0.06172239 0.8117224 0.1081250  
## 8-5 0.45833333 0.07068746 0.8459792 0.0171484  
## 7-6 0.08333333 -0.36642841 0.5330951 0.9529962  
## 8-6 0.16666667 -0.23561246 0.5689458 0.6552706  
## 8-7 0.08333333 -0.36642841 0.5330951 0.9529962

# 6-5 0.2916667 -0.14405020 0.7273835 0.2683879  
# 7-5 0.4375000 -0.05337924 0.9283792 0.0910373  
# 8-5 0.5000000 0.06428314 0.9357169 0.0211369  
# 7-6 0.1458333 -0.35970223 0.6513689 0.8484271  
# 8-6 0.2083333 -0.24383142 0.6604981 0.5767064  
# 8-7 0.0625000 -0.44303557 0.5680356 0.9851190  
  
# If not working, change visits to 'Meannumvisits' for Marbles  
par(mfrow = c(2,2))  
boxplot(visits ~ Month, data = marbindrd11, main = "Marbles (n = 32\*)", ylab = "Mean Number of Visits")  
boxplot(Meannumvisits ~ Month, data = sbrsrd11, main = "South Bristols (n = 24)", ylab = "Mean Number of Visits")  
boxplot(Meannumvisits ~ Month, data = nopard11, main = "Nopah (n = 6)", ylab = "Mean Number of Visits")  
boxplot(Meannumvisits ~ Month, data = cmprrd11, main = "Castle Piutes (n = 4)", ylab = "Mean Number of Visits")  
title("Mean Number of Visits to Water on a Given Day by Month", line = -1, outer = TRUE)  
  
## whether there is a significant difference in the percent of collared sheep visiting water between mountain ranges ##  
  
# Need to change the names of the rows for Marb  
names(marbindsheepvisitperday1)[names(marbindsheepvisitperday1) == "rownames(marbindsheepvisitperday)"] <- "Date"  
names(marbindsheepvisitperday1)[names(marbindsheepvisitperday1) == "visits"] <- "NumIndSheep"  
marbindsheepvisitperday1$datenum <- NULL  
library(lubridate)  
marbindsheepvisitperday1$Month <- month(marbindsheepvisitperday1$Date)  
# Need to add a row in all four data sets saying what mountain range it is  
marbindsheepvisitperday1$MtnRange <- "Marb"  
sbrsindsheepvisitperday2$MtnRange <- "BrsS"  
nopaindsheepvisitperday2$MtnRange <- "Nopa"  
cmprindsheepvisitperday2$MtnRange <- "CMPR"  
marbindsheepvisitperday1$MtnRange <- as.factor(marbindsheepvisitperday1$MtnRange)  
sbrsindsheepvisitperday2$MtnRange <- as.factor(sbrsindsheepvisitperday2$MtnRange)  
nopaindsheepvisitperday2$MtnRange <- as.factor(nopaindsheepvisitperday2$MtnRange)  
cmprindsheepvisitperday2$MtnRange <- as.factor(cmprindsheepvisitperday2$MtnRange)  
# then rbind the four mountain ranges  
nopaindsheepvisitperday2$TotalNumSheep <- 6  
cmprindsheepvisitperday2$TotalNumSheep <- 4  
allindsheepvisitperday <- rbind(marbindsheepvisitperday1, sbrsindsheepvisitperday2, nopaindsheepvisitperday2, cmprindsheepvisitperday2)  
qqnorm(allindsheepvisitperday$Percentvisited)  
qqline(allindsheepvisitperday$Percentvisited)  
# IS THIS HISTOGRAM OK?  
hist(allindsheepvisitperday$Percentvisited)  
boxplot(Percentvisited ~ MtnRange, data = allindsheepvisitperday, main = "Percent of Individual Sheep that Visited Water in a Given Day by Mountain Range", ylab = "Percent of Individual Sheep", xlab = "Mountain Range")  
allindsheepvisitperday$MtnRange <- as.factor(allindsheepvisitperday$MtnRange)  
PercentVisitMtnRange <- aov(Percentvisited ~ MtnRange, data = allindsheepvisitperday)  
summary.lm(PercentVisitMtnRange)

##   
## Call:  
## aov(formula = Percentvisited ~ MtnRange, data = allindsheepvisitperday)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.39018 -0.09649 -0.02937 0.08814 0.51087   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.452683 0.028308 15.992 < 2e-16 \*\*\*  
## MtnRangeBrsS -0.006644 0.035880 -0.185 0.853   
## MtnRangeNopa -0.356191 0.039769 -8.957 9.90e-16 \*\*\*  
## MtnRangeCMPR -0.213552 0.045721 -4.671 6.46e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1722 on 155 degrees of freedom  
## (138 observations deleted due to missingness)  
## Multiple R-squared: 0.4418, Adjusted R-squared: 0.431   
## F-statistic: 40.9 on 3 and 155 DF, p-value: < 2.2e-16

# p-value: < 2.2e-16  
TukeyHSD(PercentVisitMtnRange)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Percentvisited ~ MtnRange, data = allindsheepvisitperday)  
##   
## $MtnRange  
## diff lwr upr p adj  
## BrsS-Marb -0.006644472 -0.09982856 0.08653961 0.9977377  
## Nopa-Marb -0.356191495 -0.45947535 -0.25290764 0.0000000  
## CMPR-Marb -0.213552288 -0.33229455 -0.09481002 0.0000381  
## Nopa-BrsS -0.349547023 -0.44196477 -0.25712928 0.0000000  
## CMPR-BrsS -0.206907817 -0.31632998 -0.09748565 0.0000135  
## CMPR-Nopa 0.142639207 0.02449738 0.26078104 0.0109354

# diff lwr upr p adj  
# BrsS-Marb -0.006644472 -0.10476697 0.09147802 0.9980599  
# Nopa-Marb -0.356191495 -0.46494901 -0.24743398 0.0000000  
# CMPR-Marb -0.191813158 -0.31684832 -0.06677799 0.0005967  
# Nopa-BrsS -0.349547023 -0.44686256 -0.25223148 0.0000000  
# CMPR-BrsS -0.185168686 -0.30038982 -0.06994756 0.0002885  
# CMPR-Nopa 0.164378337 0.03997543 0.28878124 0.0042406  
  
  
## whether there is a significant difference in the number of visitations between mountain ranges ##  
names(marbindrd11)[names(marbindrd11) == "rownames(marbindrd)"] <- "Date"  
names(marbindrd11)[names(marbindrd11) == "visits"] <- "Meannumvisits"  
marbindrd11$MtnRange <- "Marb"  
sbrsrd11$MtnRange <- "BrsS"  
nopard11$MtnRange <- "Nopa"  
cmprrd11$MtnRange <- "CMPR"  
marbindrd11$MtnRange <- as.factor(marbindrd11$MtnRange)  
sbrsrd11$MtnRange <- as.factor(sbrsrd11$MtnRange)  
nopard11$MtnRange <- as.factor(nopard11$MtnRange)  
cmprrd11$MtnRange <- as.factor(cmprrd11$MtnRange)  
allindrd11 <- rbind(marbindrd11, sbrsrd11, nopard11, cmprrd11)  
qqnorm(allindrd11$Meannumvisits)  
qqline(allindrd11$Meannumvisits)  
hist(allindrd11$Meannumvisits)  
boxplot(Meannumvisits ~ MtnRange, data = allindrd11, main = "Mean Number of Visitations of Collared Bighorn to Water in a Given Day by Mountain Range", ylab = "Mean Number of Visitations", xlab = "Mountain Range")  
allindrd11$MtnRange <- as.factor(allindrd11$MtnRange)  
MeanVisitationMtnRange <- aov(Meannumvisits ~ MtnRange, data = allindrd11)  
summary.lm(MeanVisitationMtnRange)

##   
## Call:  
## aov(formula = Meannumvisits ~ MtnRange, data = allindrd11)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.49369 -0.15232 -0.03535 0.13131 0.87179   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.48799 0.03789 12.881 < 2e-16 \*\*\*  
## MtnRangeBrsS 0.08903 0.04935 1.804 0.07301 .   
## MtnRangeNopa -0.35979 0.05593 -6.433 1.22e-09 \*\*\*  
## MtnRangeCMPR -0.22712 0.06562 -3.461 0.00068 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2569 on 170 degrees of freedom  
## (138 observations deleted due to missingness)  
## Multiple R-squared: 0.3379, Adjusted R-squared: 0.3263   
## F-statistic: 28.93 on 3 and 170 DF, p-value: 3.668e-15

# p-value: 1.084e-15  
TukeyHSD(MeanVisitationMtnRange)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Meannumvisits ~ MtnRange, data = allindrd11)  
##   
## $MtnRange  
## diff lwr upr p adj  
## BrsS-Marb 0.08902932 -0.03902055 0.2170792 0.2750408  
## Nopa-Marb -0.35978576 -0.50490307 -0.2146684 0.0000000  
## CMPR-Marb -0.22712132 -0.39737743 -0.0568652 0.0037652  
## Nopa-BrsS -0.44881507 -0.58346663 -0.3141635 0.0000000  
## CMPR-BrsS -0.31615064 -0.47757910 -0.1547222 0.0000058  
## CMPR-Nopa 0.13266444 -0.04261083 0.3079397 0.2058705

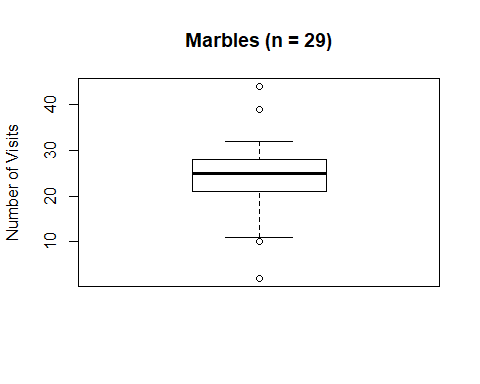
# diff lwr upr p adj  
# BrsS-Marb 0.08276173 -0.0457863 0.21130977 0.3424864  
# Nopa-Marb -0.37468619 -0.5184595 -0.23091286 0.0000000  
# CMPR-Marb -0.22707749 -0.3969096 -0.05724541 0.0036673  
# Nopa-BrsS -0.45744792 -0.5914880 -0.32340783 0.0000000  
# CMPR-BrsS -0.30983922 -0.4715145 -0.14816390 0.0000096  
# CMPR-Nopa 0.14760870 -0.0264174 0.32163479 0.1271620

Next, I did a more general examination of the spread in the number of individual visits between individual sheep. In doing this, I was not trying to examine the difference in number of visits by individual sheep between mountain ranges, but rather what was the general spread of the number of visits within each mountain range (these spreads are therefore not comparable because I did not account for the different study time periods within the different ranges). In the Marbles, the spread of the data was relatively small (mean of 25.41, 1st quartile at 22, 3rd quartile at 30). In the South Bristols, the spread was larger (mean of 36.48, 1st quartile of 23, 3rd quartile of 54).

I also looked at the amount of time between visits by individual sheep. The relatively small chunks of time when all of the cameras were running were not worth analyzing because they would be biased against long amounts of time between visits. Therefore, I only analyzed the time period in the South Bristols when the cameras ran for 33 complete days straight (6/26/19 - 7/28/19). I only included complete days to have this analysis be at the same relative time period as my other analyses.

The mean number of hours between visits by an individual sheep was 37.57. The first quartile was 11.87 hours, while the third quartile was 47.16 hours. There was a significant difference between sheep in the amount of time between visits to water (p-value: < 8.52e-16), though only 13% of combinations between individual sheep (36 out of 276 possible combinations) were significantly different.

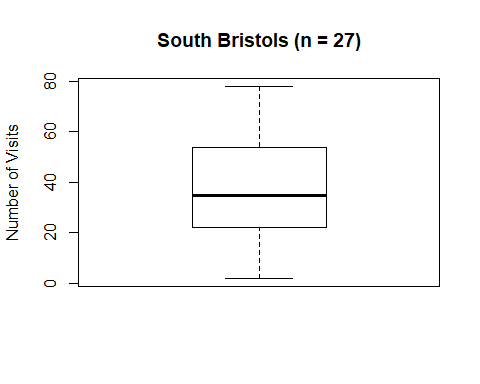
# Whether there is a large spread in the number of individual visits between individual sheep  
  
## It is worth noting that I have not deleted replicate observations from Theresa from this dataset. So if a sheep went to 'Theresa Seep' and 'Theresa Drinker Box' within the same visit, it still counts as two visits to water. SHOULD I FIX THIS AT SOME POINT, OR SHOULD THESE COUNT AS SEPARATE OBSERVATIONS?  
# Also, Marbles has including the total, completely covered time frame (only includes complete days)  
# Note that the exact numbers are not comparable between mountain ranges b/c of different amounts of time that cameras were functioning within each range. Didn't use a modifer b/c comparing between ranges isn't what I'm interested in right now.  
  
MarbNumVisitbyID <- as.data.frame(table(Marbcutobsrd$KI\_Known\_ID))  
## Removing the two sheep that didn't live through the whole time period: BHS\_1595 & BHS\_1367  
MarbNumVisitbyID1 <- MarbNumVisitbyID[(MarbNumVisitbyID$Var1 == "BHS\_1365") | (MarbNumVisitbyID$Var1 == "BHS\_1361") | (MarbNumVisitbyID$Var1 == "BHS\_1362") | (MarbNumVisitbyID$Var1 == "BHS\_1357") | (MarbNumVisitbyID$Var1 == "BHS\_1416") | (MarbNumVisitbyID$Var1 == "BHS\_1421") | (MarbNumVisitbyID$Var1 == "BHS\_1581") | (MarbNumVisitbyID$Var1 == "BHS\_1577") | (MarbNumVisitbyID$Var1 == "BHS\_1574") | (MarbNumVisitbyID$Var1 == "BHS\_1576") | (MarbNumVisitbyID$Var1 == "BHS\_1583") | (MarbNumVisitbyID$Var1 == "BHS\_1584") | (MarbNumVisitbyID$Var1 == "BHS\_1585") | (MarbNumVisitbyID$Var1 == "BHS\_1587") | (MarbNumVisitbyID$Var1 == "BHS\_1582") | (MarbNumVisitbyID$Var1 == "BHS\_1575") | (MarbNumVisitbyID$Var1 == "BHS\_1592") | (MarbNumVisitbyID$Var1 == "BHS\_1594") | (MarbNumVisitbyID$Var1 == "BHS\_1420") | (MarbNumVisitbyID$Var1 == "BHS\_1493") | (MarbNumVisitbyID$Var1 == "BHS\_1494") | (MarbNumVisitbyID$Var1 == "BHS\_1590") | (MarbNumVisitbyID$Var1 == "BHS\_1591") | (MarbNumVisitbyID$Var1 == "BHS\_1682") | (MarbNumVisitbyID$Var1 == "BHS\_1683") | (MarbNumVisitbyID$Var1 == "BHS\_1684") | (MarbNumVisitbyID$Var1 == "BHS\_1366") | (MarbNumVisitbyID$Var1 == "BHS\_1497") | (MarbNumVisitbyID$Var1 == "BHS\_1588") | (MarbNumVisitbyID$Var1 == "BHS\_1589"), ]  
boxplot(MarbNumVisitbyID1$Freq, ylab = "Number of Visits", main = "Marbles (n = 29)")



summary(MarbNumVisitbyID1$Freq)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.00 21.00 25.00 24.34 28.00 44.00

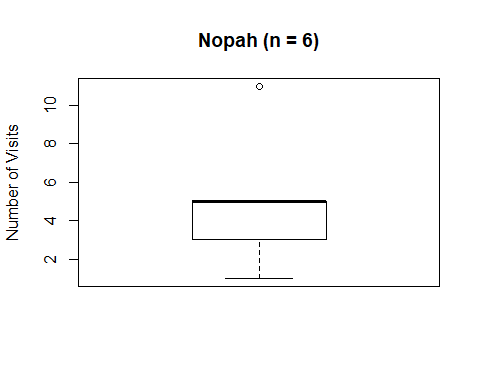
# Min. 1st Qu. Median Mean 3rd Qu. Max.  
# 2.00 22.00 25.00 25.41 30.00 46.00  
  
# South Bristols only covers complete days  
BrsSNumVisitbyID <- as.data.frame(table(BrsScutobsrdwithdups$KI\_Known\_ID))  
boxplot(BrsSNumVisitbyID$Freq, ylab = "Number of Visits", main = "South Bristols (n = 27)")



summary(BrsSNumVisitbyID$Freq)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.00 22.00 35.00 36.64 54.00 78.00

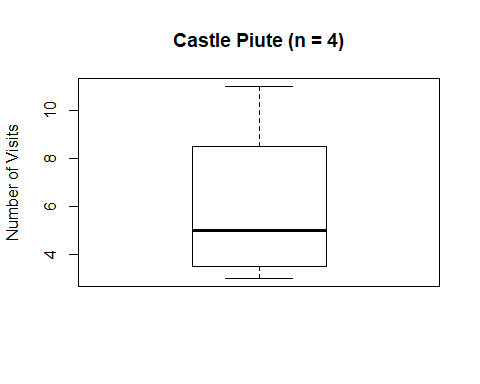
#Min. 1st Qu. Median Mean 3rd Qu. Max.  
# 2.00 23.00 35.00 36.48 54.00 78.00  
  
# Nopah  
NopaNumVisitbyID <- as.data.frame(table((nopacutobsrd$KI\_Known\_ID)))  
NopaNumVisitbyID1 <- NopaNumVisitbyID[(NopaNumVisitbyID$Var1 == "BHS\_1737") | (NopaNumVisitbyID == "BHS\_1740") | (NopaNumVisitbyID$Var1 == "BHS\_1741") | (NopaNumVisitbyID$Var1 == "BHS\_1738") | (NopaNumVisitbyID$Var1 == "BHS\_1736") | (NopaNumVisitbyID$Var1 == "BHS\_1739"),]  
boxplot(NopaNumVisitbyID1$Freq, ylab = "Number of Visits", main = "Nopah (n = 6)")



summary(NopaNumVisitbyID1$Freq)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 1 3 5 5 5 11 4

# Min. 1st Qu. Median Mean 3rd Qu. Max. NA's  
# 1.0 4.0 5.0 5.4 5.0 12.0  
  
CMPRNumVisitbyID <- as.data.frame(table((cmprcutobs$KI\_Known\_ID)))  
CMPRNumVisitbyID1 <- CMPRNumVisitbyID[(CMPRNumVisitbyID$Var1 == "BHS\_1730") | (CMPRNumVisitbyID$Var1 == "BHS\_1733") | (CMPRNumVisitbyID$Var1 == "BHS\_1734") | (CMPRNumVisitbyID$Var1 == "BHS\_1728"), ]  
boxplot(CMPRNumVisitbyID1$Freq, ylab = "Number of Visits", main = "Castle Piute (n = 4)")



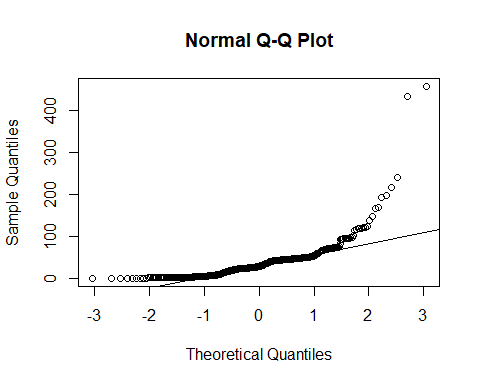
summary(CMPRNumVisitbyID$Freq)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0000 0.0000 0.0000 0.3803 0.0000 11.0000

# Min. 1st Qu. Median Mean 3rd Qu. Max.  
# 0.0000 0.0000 0.0000 0.4085 0.0000 11.0000  
  
# Note - different mountain ranges are not comparable b/c are over different amounts of time. Just looking at spread within each mountain range.  
  
# Time between sheep coming into water  
  
BrsScutobslong <- BrsScutobs[(BrsScutobs$DateTimect >= as.POSIXct("2019-06-25 17:05:00") & BrsScutobs$DateTimect <= as.POSIXct("2019-07-29 10:36:00")), ]  
# 25 sheep  
# https://stackoverflow.com/questions/32999460/how-to-calculate-time-difference-with-previous-row-of-a-data-frame-by-group/32999651  
BrsScutobslong$DateTime <- as.POSIXct(BrsScutobslong$DateTime, format = )  
library(dplyr)  
BrsSlongtimedif <- BrsScutobslong %>%  
 arrange(KI\_Known\_ID, DateTimect) %>%  
 group\_by(KI\_Known\_ID) %>%  
 mutate(Timediff = DateTimect - lag(DateTimect), diff\_hours = as.numeric(Timediff, units = 'hours'))  
# mean(BrsSlongtimedif$diff\_hours, na.rm = TRUE)  
# [1] 37.57339  
# boxplot(BrsSlongtimedif$diff\_hours, ylab = "Time Between Visits to Water (hours)", main = "Time between Visits to Water for all Sheep")  
summary(BrsSlongtimedif$diff\_hours)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.10 11.87 28.53 37.57 47.16 456.88 25

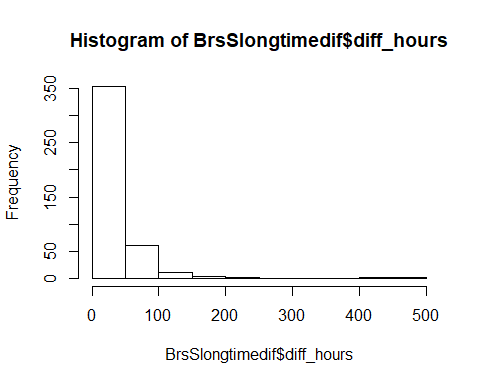
# Min. 1st Qu. Median Mean 3rd Qu. Max. NA's  
# 0.10 11.87 28.53 37.57 47.16 456.88 25  
qqnorm(BrsSlongtimedif$diff\_hours)  
qqline(BrsSlongtimedif$diff\_hours)



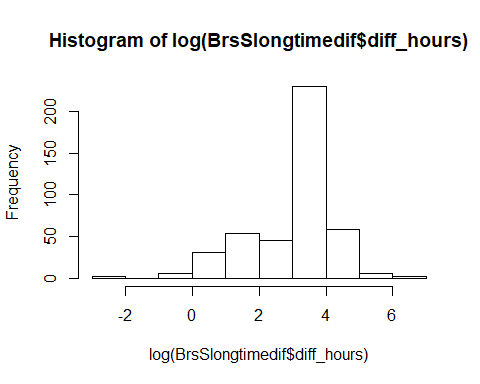
shapiro.test(BrsSlongtimedif$diff\_hours)

##   
## Shapiro-Wilk normality test  
##   
## data: BrsSlongtimedif$diff\_hours  
## W = 0.62787, p-value < 2.2e-16

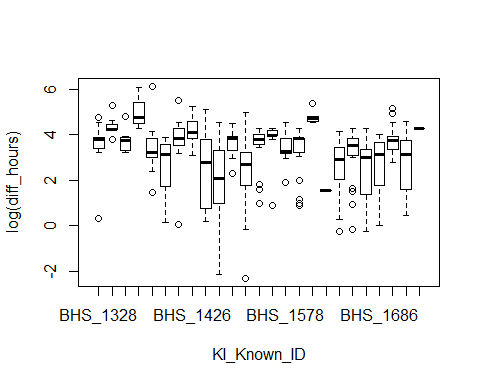
# p-value < 2.2e-16 NOT NORMALLY DISTRIBUTED? Shouldn't use anova?  
hist(BrsSlongtimedif$diff\_hours)



# very right-skewed - log transform  
hist(log(BrsSlongtimedif$diff\_hours))



boxplot(log(diff\_hours) ~ KI\_Known\_ID, data = BrsSlongtimedif)  
boxplot(log(diff\_hours) ~ KI\_Known\_ID, data = BrsSlongtimedif)



# Looking at the non-logged boxplot, there is one sheep that is super off compared to the rest of them???  
BrsSdiffhoursindsheep <- aov(log(diff\_hours) ~ KI\_Known\_ID, data = BrsSlongtimedif)  
summary.lm(BrsSdiffhoursindsheep)

##   
## Call:  
## aov(formula = log(diff\_hours) ~ KI\_Known\_ID, data = BrsSlongtimedif)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -4.6564 -0.4224 0.2094 0.6603 2.7361   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.59037 0.27901 12.868 < 2e-16 \*\*\*  
## KI\_Known\_IDBHS\_1329 0.77790 0.48326 1.610 0.108237   
## KI\_Known\_IDBHS\_1330 0.10219 0.38346 0.266 0.789990   
## KI\_Known\_IDBHS\_1333 1.38022 0.62389 2.212 0.027498 \*   
## KI\_Known\_IDBHS\_1334 -0.20205 0.42620 -0.474 0.635700   
## KI\_Known\_IDBHS\_1337 -1.02294 0.33533 -3.051 0.002433 \*\*   
## KI\_Known\_IDBHS\_1425 0.10215 0.42620 0.240 0.810702   
## KI\_Known\_IDBHS\_1426 0.57744 0.48326 1.195 0.232829   
## KI\_Known\_IDBHS\_1428 -1.06894 0.46502 -2.299 0.022025 \*   
## KI\_Known\_IDBHS\_1486 -1.50335 0.36669 -4.100 4.99e-05 \*\*\*  
## KI\_Known\_IDBHS\_1489 0.08061 0.38346 0.210 0.833601   
## KI\_Known\_IDBHS\_1490 -1.23654 0.33835 -3.655 0.000291 \*\*\*  
## KI\_Known\_IDBHS\_1491 -0.11208 0.38346 -0.292 0.770222   
## KI\_Known\_IDBHS\_1492 0.23527 0.41672 0.565 0.572678   
## KI\_Known\_IDBHS\_1578 -0.10778 0.37035 -0.291 0.771190   
## KI\_Known\_IDBHS\_1579 -0.31331 0.37035 -0.846 0.398051   
## KI\_Known\_IDBHS\_1580 1.21367 0.57180 2.123 0.034393 \*   
## KI\_Known\_IDBHS\_1591 -2.03574 1.15039 -1.770 0.077538 .   
## KI\_Known\_IDBHS\_1596 -0.97615 0.33393 -2.923 0.003657 \*\*   
## KI\_Known\_IDBHS\_1599 -0.41418 0.36020 -1.150 0.250879   
## KI\_Known\_IDBHS\_1685 -1.10866 0.33393 -3.320 0.000981 \*\*\*  
## KI\_Known\_IDBHS\_1686 -0.86874 0.38874 -2.235 0.025971 \*   
## KI\_Known\_IDBHS\_1687 0.23170 0.41672 0.556 0.578507   
## KI\_Known\_IDBHS\_1688 -0.82077 0.34756 -2.362 0.018667 \*   
## KI\_Known\_IDBHS\_1689 0.69551 1.15039 0.605 0.545789   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.116 on 409 degrees of freedom  
## (25 observations deleted due to missingness)  
## Multiple R-squared: 0.2592, Adjusted R-squared: 0.2157   
## F-statistic: 5.962 on 24 and 409 DF, p-value: 8.52e-16

# p-value: < 8.52e-16  
TukeyHSD(BrsSdiffhoursindsheep)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = log(diff\_hours) ~ KI\_Known\_ID, data = BrsSlongtimedif)  
##   
## $KI\_Known\_ID  
## diff lwr upr p adj  
## BHS\_1329-BHS\_1328 7.779021e-01 -1.002218132 2.558022266 0.9968279  
## BHS\_1330-BHS\_1328 1.021923e-01 -1.310318957 1.514703486 1.0000000  
## BHS\_1333-BHS\_1328 1.380222e+00 -0.917903436 3.678347153 0.8756911  
## BHS\_1334-BHS\_1328 -2.020487e-01 -1.771967142 1.367869757 1.0000000  
## BHS\_1337-BHS\_1328 -1.022938e+00 -2.258142989 0.212267473 0.2826535  
## BHS\_1425-BHS\_1328 1.021488e-01 -1.467769637 1.672067263 1.0000000  
## BHS\_1426-BHS\_1328 5.774367e-01 -1.202683541 2.357556856 0.9999757  
## BHS\_1428-BHS\_1328 -1.068940e+00 -2.781861062 0.643981858 0.8299849  
## BHS\_1486-BHS\_1328 -1.503349e+00 -2.854081198 -0.152617386 0.0114974  
## BHS\_1489-BHS\_1328 8.061168e-02 -1.331899545 1.493122898 1.0000000  
## BHS\_1490-BHS\_1328 -1.236540e+00 -2.482873001 0.009793860 0.0548887  
## BHS\_1491-BHS\_1328 -1.120782e-01 -1.524589426 1.300433017 1.0000000  
## BHS\_1492-BHS\_1328 2.352687e-01 -1.299758345 1.770295688 1.0000000  
## BHS\_1578-BHS\_1328 -1.077761e-01 -1.471981972 1.256429778 1.0000000  
## BHS\_1579-BHS\_1328 -3.133143e-01 -1.677520150 1.050891600 1.0000000  
## BHS\_1580-BHS\_1328 1.213675e+00 -0.892591864 3.319941384 0.9142283  
## BHS\_1591-BHS\_1328 -2.035740e+00 -6.273273819 2.201793511 0.9885776  
## BHS\_1596-BHS\_1328 -9.761460e-01 -2.206204546 0.253912494 0.3674132  
## BHS\_1599-BHS\_1328 -4.141769e-01 -1.741000202 0.912646313 0.9999880  
## BHS\_1685-BHS\_1328 -1.108662e+00 -2.338720402 0.121396638 0.1468609  
## BHS\_1686-BHS\_1328 -8.687404e-01 -2.300668503 0.563187685 0.8645947  
## BHS\_1687-BHS\_1328 2.317035e-01 -1.303323492 1.766730541 1.0000000  
## BHS\_1688-BHS\_1328 -8.207737e-01 -2.101027116 0.459479807 0.7917425  
## BHS\_1689-BHS\_1328 6.955129e-01 -3.542020720 4.933046609 1.0000000  
## BHS\_1330-BHS\_1329 -6.757098e-01 -2.422553792 1.071134188 0.9995032  
## BHS\_1333-BHS\_1329 6.023198e-01 -1.915150336 3.119789919 0.9999999  
## BHS\_1334-BHS\_1329 -9.799508e-01 -2.856362205 0.896460686 0.9688932  
## BHS\_1337-BHS\_1329 -1.800840e+00 -3.407702588 -0.193977063 0.0103121  
## BHS\_1425-BHS\_1329 -6.757533e-01 -2.552164700 1.200658192 0.9998476  
## BHS\_1426-BHS\_1329 -2.004654e-01 -2.255971162 1.855040342 1.0000000  
## BHS\_1428-BHS\_1329 -1.846842e+00 -3.844434196 0.150750857 0.1161626  
## BHS\_1486-BHS\_1329 -2.281251e+00 -3.978529373 -0.583973346 0.0002994  
## BHS\_1489-BHS\_1329 -6.972904e-01 -2.444134381 1.049553599 0.9991823  
## BHS\_1490-BHS\_1329 -2.014442e+00 -3.629874395 -0.399008880 0.0015239  
## BHS\_1491-BHS\_1329 -8.899803e-01 -2.636824262 0.856863718 0.9765812  
## BHS\_1492-BHS\_1329 -5.426334e-01 -2.389951429 1.304684638 0.9999962  
## BHS\_1578-BHS\_1329 -8.856782e-01 -2.593698551 0.822342223 0.9713002  
## BHS\_1579-BHS\_1329 -1.091216e+00 -2.799236729 0.616804045 0.7969536  
## BHS\_1580-BHS\_1329 4.357727e-01 -1.907864452 2.779409837 1.0000000  
## BHS\_1591-BHS\_1329 -2.813642e+00 -7.174028389 1.546743947 0.7818000  
## BHS\_1596-BHS\_1329 -1.754048e+00 -3.356957927 -0.151138259 0.0148972  
## BHS\_1599-BHS\_1329 -1.192079e+00 -2.870392431 0.486234406 0.6049378  
## BHS\_1685-BHS\_1329 -1.886564e+00 -3.489473784 -0.283654115 0.0045365  
## BHS\_1686-BHS\_1329 -1.646642e+00 -3.409224117 0.115939165 0.1049605  
## BHS\_1687-BHS\_1329 -5.461985e-01 -2.393516576 1.301119491 0.9999957  
## BHS\_1688-BHS\_1329 -1.598676e+00 -3.240420190 0.043068747 0.0679365  
## BHS\_1689-BHS\_1329 -8.238912e-02 -4.442775291 4.277997045 1.0000000  
## BHS\_1333-BHS\_1330 1.278030e+00 -0.994417517 3.550476705 0.9320625  
## BHS\_1334-BHS\_1330 -3.042410e-01 -1.836324487 1.227842572 1.0000000  
## BHS\_1337-BHS\_1330 -1.125130e+00 -2.311876822 0.061616776 0.0904643  
## BHS\_1425-BHS\_1330 -4.345152e-05 -1.532126981 1.532040078 1.0000000  
## BHS\_1426-BHS\_1330 4.752444e-01 -1.271599597 2.222088382 0.9999991  
## BHS\_1428-BHS\_1330 -1.171132e+00 -2.849445286 0.507181551 0.6411595  
## BHS\_1486-BHS\_1330 -1.605542e+00 -2.912106781 -0.298976333 0.0020367  
## BHS\_1489-BHS\_1330 -2.158059e-02 -1.391917756 1.348756580 1.0000000  
## BHS\_1490-BHS\_1330 -1.338732e+00 -2.537056926 -0.140406744 0.0108417  
## BHS\_1491-BHS\_1330 -2.142705e-01 -1.584607637 1.156066699 1.0000000  
## BHS\_1492-BHS\_1330 1.330764e-01 -1.363233703 1.629386517 1.0000000  
## BHS\_1578-BHS\_1330 -2.099684e-01 -1.530458298 1.110521574 1.0000000  
## BHS\_1579-BHS\_1330 -4.155065e-01 -1.735996476 0.904983396 0.9999861  
## BHS\_1580-BHS\_1330 1.111482e+00 -0.966736717 3.189701707 0.9595655  
## BHS\_1591-BHS\_1330 -2.137932e+00 -6.361595232 2.085730394 0.9783029  
## BHS\_1596-BHS\_1330 -1.078338e+00 -2.259727289 0.103050708 0.1310111  
## BHS\_1599-BHS\_1330 -5.163692e-01 -1.798202257 0.765463837 0.9990573  
## BHS\_1685-BHS\_1330 -1.210854e+00 -2.392243145 -0.029465149 0.0368886  
## BHS\_1686-BHS\_1330 -9.709327e-01 -2.361275822 0.419410474 0.6396127  
## BHS\_1687-BHS\_1330 1.295113e-01 -1.366798850 1.625821370 1.0000000  
## BHS\_1688-BHS\_1330 -9.229659e-01 -2.156531859 0.310600021 0.4936624  
## BHS\_1689-BHS\_1330 5.933207e-01 -3.630342134 4.816983493 1.0000000  
## BHS\_1334-BHS\_1333 -1.582271e+00 -3.955764150 0.791223047 0.7276477  
## BHS\_1337-BHS\_1333 -2.403160e+00 -4.569852923 -0.236466310 0.0121335  
## BHS\_1425-BHS\_1333 -1.278073e+00 -3.651566644 1.095420553 0.9565005  
## BHS\_1426-BHS\_1333 -8.027852e-01 -3.320255329 1.714684926 0.9999822  
## BHS\_1428-BHS\_1333 -2.449161e+00 -4.919571923 0.021249001 0.0553716  
## BHS\_1486-BHS\_1333 -2.883571e+00 -5.118141491 -0.649000811 0.0007475  
## BHS\_1489-BHS\_1333 -1.299610e+00 -3.572057293 0.972836929 0.9201127  
## BHS\_1490-BHS\_1333 -2.616761e+00 -4.789818019 -0.443704839 0.0029947  
## BHS\_1491-BHS\_1333 -1.492300e+00 -3.764747174 0.780147048 0.7534942  
## BHS\_1492-BHS\_1333 -1.144953e+00 -3.495513942 1.205607568 0.9864000  
## BHS\_1578-BHS\_1333 -1.487998e+00 -3.730738598 0.754742687 0.7359915  
## BHS\_1579-BHS\_1333 -1.693536e+00 -3.936276776 0.549204509 0.4736719  
## BHS\_1580-BHS\_1333 -1.665471e-01 -2.924297453 2.591203255 1.0000000  
## BHS\_1591-BHS\_1333 -3.415962e+00 -8.012212602 1.180288577 0.5082432  
## BHS\_1596-BHS\_1333 -2.356368e+00 -4.520131246 -0.192604523 0.0160096  
## BHS\_1599-BHS\_1333 -1.794399e+00 -4.014598767 0.425801160 0.3300678  
## BHS\_1685-BHS\_1333 -2.488884e+00 -4.652647102 -0.325120379 0.0067422  
## BHS\_1686-BHS\_1333 -2.248962e+00 -4.533529183 0.035604647 0.0600887  
## BHS\_1687-BHS\_1333 -1.148518e+00 -3.499079089 1.202042421 0.9858625  
## BHS\_1688-BHS\_1333 -2.200996e+00 -4.393682642 -0.008308384 0.0477798  
## BHS\_1689-BHS\_1333 -6.847089e-01 -5.280959504 3.911541675 1.0000000  
## BHS\_1337-BHS\_1334 -8.208891e-01 -2.191226234 0.549448102 0.8784062  
## BHS\_1425-BHS\_1334 3.041975e-01 -1.374115913 1.982510924 1.0000000  
## BHS\_1426-BHS\_1334 7.794853e-01 -1.096926096 2.655896795 0.9984919  
## BHS\_1428-BHS\_1334 -8.668909e-01 -2.679676589 0.945894770 0.9892274  
## BHS\_1486-BHS\_1334 -1.301301e+00 -2.776620454 0.174019255 0.1767067  
## BHS\_1489-BHS\_1334 2.826604e-01 -1.249423161 1.814743899 1.0000000  
## BHS\_1490-BHS\_1334 -1.034491e+00 -2.414867280 0.345885524 0.4901281  
## BHS\_1491-BHS\_1334 8.997049e-02 -1.442113042 1.622054018 1.0000000  
## BHS\_1492-BHS\_1334 4.373174e-01 -1.208404342 2.083039070 0.9999995  
## BHS\_1578-BHS\_1334 9.427260e-02 -1.393393247 1.581938438 1.0000000  
## BHS\_1579-BHS\_1334 -1.112656e-01 -1.598931425 1.376400260 1.0000000  
## BHS\_1580-BHS\_1334 1.415723e+00 -0.772529523 3.603976427 0.7777056  
## BHS\_1591-BHS\_1334 -1.833691e+00 -6.112567897 2.445184974 0.9976074  
## BHS\_1596-BHS\_1334 -7.740973e-01 -2.139797137 0.591602470 0.9266775  
## BHS\_1599-BHS\_1334 -2.121283e-01 -1.665590308 1.241333804 1.0000000  
## BHS\_1685-BHS\_1334 -9.066132e-01 -2.272312993 0.459086614 0.7350169  
## BHS\_1686-BHS\_1334 -6.666917e-01 -2.216694961 0.883311528 0.9974758  
## BHS\_1687-BHS\_1334 4.337522e-01 -1.211969489 2.079473923 0.9999995  
## BHS\_1688-BHS\_1334 -6.187250e-01 -2.029802889 0.792352965 0.9966708  
## BHS\_1689-BHS\_1334 8.975616e-01 -3.381314799 5.176438072 1.0000000  
## BHS\_1425-BHS\_1337 1.125087e+00 -0.245250596 2.495423740 0.2991116  
## BHS\_1426-BHS\_1337 1.600374e+00 -0.006488347 3.207237178 0.0524631  
## BHS\_1428-BHS\_1337 -4.600184e-02 -1.578085374 1.486081686 1.0000000  
## BHS\_1486-BHS\_1337 -4.804115e-01 -1.592911735 0.632088667 0.9973264  
## BHS\_1489-BHS\_1337 1.103549e+00 -0.083197364 2.290296234 0.1098522  
## BHS\_1490-BHS\_1337 -2.136018e-01 -1.196722880 0.769519256 1.0000000  
## BHS\_1491-BHS\_1337 9.108596e-01 -0.275887245 2.097606353 0.4383320  
## BHS\_1492-BHS\_1337 1.258206e+00 -0.072014588 2.588427448 0.0926668  
## BHS\_1578-BHS\_1337 9.151617e-01 -0.213659708 2.043983030 0.3238760  
## BHS\_1579-BHS\_1337 7.096235e-01 -0.419197886 1.838444852 0.8200769  
## BHS\_1580-BHS\_1337 2.236613e+00 0.274588281 4.198636755 0.0078323  
## BHS\_1591-BHS\_1337 -1.012802e+00 -5.180520185 3.154915394 0.9999999  
## BHS\_1596-BHS\_1337 4.679173e-02 -0.915613576 1.009197041 1.0000000  
## BHS\_1599-BHS\_1337 6.087608e-01 -0.474585840 1.692107467 0.9326268  
## BHS\_1685-BHS\_1337 -8.572412e-02 -1.048129432 0.876681184 1.0000000  
## BHS\_1686-BHS\_1337 1.541973e-01 -1.055595229 1.363989927 1.0000000  
## BHS\_1687-BHS\_1337 1.254641e+00 -0.075579735 2.584862301 0.0953880  
## BHS\_1688-BHS\_1337 2.021641e-01 -0.823618007 1.227946215 1.0000000  
## BHS\_1689-BHS\_1337 1.718451e+00 -2.449267087 5.886168492 0.9986527  
## BHS\_1426-BHS\_1425 4.752878e-01 -1.401123602 2.351699290 0.9999998  
## BHS\_1428-BHS\_1425 -1.171088e+00 -2.983874095 0.641697264 0.7800076  
## BHS\_1486-BHS\_1425 -1.605498e+00 -3.080817960 -0.130178251 0.0161794  
## BHS\_1489-BHS\_1425 -2.153714e-02 -1.553620667 1.510546393 1.0000000  
## BHS\_1490-BHS\_1425 -1.338688e+00 -2.719064786 0.041688018 0.0710728  
## BHS\_1491-BHS\_1425 -2.142270e-01 -1.746310548 1.317856512 1.0000000  
## BHS\_1492-BHS\_1425 1.331199e-01 -1.512601848 1.778841564 1.0000000  
## BHS\_1578-BHS\_1425 -2.099249e-01 -1.697590753 1.277740932 1.0000000  
## BHS\_1579-BHS\_1425 -4.154631e-01 -1.903128931 1.072202754 0.9999986  
## BHS\_1580-BHS\_1425 1.111526e+00 -1.076727029 3.299778921 0.9773912  
## BHS\_1591-BHS\_1425 -2.137889e+00 -6.416765403 2.140987468 0.9814217  
## BHS\_1596-BHS\_1425 -1.078295e+00 -2.443994643 0.287404965 0.3780158  
## BHS\_1599-BHS\_1425 -5.163258e-01 -1.969787814 0.937136298 0.9998793  
## BHS\_1685-BHS\_1425 -1.210811e+00 -2.576510499 0.154889108 0.1693006  
## BHS\_1686-BHS\_1425 -9.708892e-01 -2.520892467 0.579114022 0.8249911  
## BHS\_1687-BHS\_1425 1.295547e-01 -1.516166995 1.775276417 1.0000000  
## BHS\_1688-BHS\_1425 -9.229225e-01 -2.334000395 0.488155459 0.7601687  
## BHS\_1689-BHS\_1425 5.933641e-01 -3.685512305 4.872240566 1.0000000  
## BHS\_1428-BHS\_1426 -1.646376e+00 -3.643968786 0.351216267 0.2917629  
## BHS\_1486-BHS\_1426 -2.080786e+00 -3.778063963 -0.383507936 0.0021312  
## BHS\_1489-BHS\_1426 -4.968250e-01 -2.243668971 1.250019009 0.9999980  
## BHS\_1490-BHS\_1426 -1.813976e+00 -3.429408985 -0.198543470 0.0099977  
## BHS\_1491-BHS\_1426 -6.895149e-01 -2.436358852 1.057329128 0.9993144  
## BHS\_1492-BHS\_1426 -3.421680e-01 -2.189486019 1.505150048 1.0000000  
## BHS\_1578-BHS\_1426 -6.852128e-01 -2.393233141 1.022807633 0.9991159  
## BHS\_1579-BHS\_1426 -8.907509e-01 -2.598771319 0.817269455 0.9693834  
## BHS\_1580-BHS\_1426 6.362381e-01 -1.707399042 2.979875247 0.9999992  
## BHS\_1591-BHS\_1426 -2.613177e+00 -6.973562979 1.747209356 0.8779557  
## BHS\_1596-BHS\_1426 -1.553583e+00 -3.156492517 0.049327151 0.0715358  
## BHS\_1599-BHS\_1426 -9.916136e-01 -2.669927021 0.686699816 0.8922081  
## BHS\_1685-BHS\_1426 -1.686099e+00 -3.289008374 -0.083188705 0.0262022  
## BHS\_1686-BHS\_1426 -1.446177e+00 -3.208758707 0.316404575 0.3003840  
## BHS\_1687-BHS\_1426 -3.457331e-01 -2.193051166 1.501584901 1.0000000  
## BHS\_1688-BHS\_1426 -1.398210e+00 -3.039954780 0.243534157 0.2323425  
## BHS\_1689-BHS\_1426 1.180763e-01 -4.242309881 4.478462455 1.0000000  
## BHS\_1486-BHS\_1428 -4.344097e-01 -2.061070276 1.192250896 0.9999994  
## BHS\_1489-BHS\_1428 1.149551e+00 -0.528762140 2.827864697 0.6776946  
## BHS\_1490-BHS\_1428 -1.676000e-01 -1.708669402 1.373469466 1.0000000  
## BHS\_1491-BHS\_1428 9.568614e-01 -0.721452021 2.635174816 0.9224366  
## BHS\_1492-BHS\_1428 1.304208e+00 -0.478445905 3.086862453 0.5422603  
## BHS\_1578-BHS\_1428 9.611635e-01 -0.676702682 2.599029693 0.8985615  
## BHS\_1579-BHS\_1428 7.556253e-01 -0.882240860 2.393491515 0.9932733  
## BHS\_1580-BHS\_1428 2.282614e+00 -0.010398301 4.575627024 0.0527733  
## BHS\_1591-BHS\_1428 -9.668006e-01 -5.300187165 3.366586061 1.0000000  
## BHS\_1596-BHS\_1428 9.279358e-02 -1.435143576 1.620730729 1.0000000  
## BHS\_1599-BHS\_1428 6.547627e-01 -0.952100105 2.261625419 0.9988750  
## BHS\_1685-BHS\_1428 -3.972228e-02 -1.567659433 1.488214873 1.0000000  
## BHS\_1686-BHS\_1428 2.001992e-01 -1.494488403 1.894886789 1.0000000  
## BHS\_1687-BHS\_1428 1.300643e+00 -0.482011052 3.083297306 0.5481458  
## BHS\_1688-BHS\_1428 2.481659e-01 -1.320463041 1.816794937 1.0000000  
## BHS\_1689-BHS\_1428 1.764453e+00 -2.568934067 6.097839159 0.9988878  
## BHS\_1489-BHS\_1486 1.583961e+00 0.277395744 2.890526193 0.0026399  
## BHS\_1490-BHS\_1486 2.668097e-01 -0.858033269 1.391652713 0.9999999  
## BHS\_1491-BHS\_1486 1.391271e+00 0.084705864 2.697836312 0.0221311  
## BHS\_1492-BHS\_1486 1.738618e+00 0.300482825 3.176753102 0.0027820  
## BHS\_1578-BHS\_1486 1.395573e+00 0.141387243 2.649759148 0.0115388  
## BHS\_1579-BHS\_1486 1.190035e+00 -0.064150935 2.444220969 0.0897081  
## BHS\_1580-BHS\_1486 2.717024e+00 0.680290435 4.753757669 0.0003565  
## BHS\_1591-BHS\_1486 -5.323909e-01 -4.735796183 3.671014459 1.0000000  
## BHS\_1596-BHS\_1486 5.272033e-01 -0.579579773 1.633986306 0.9897565  
## BHS\_1599-BHS\_1486 1.089172e+00 -0.124246250 2.302590944 0.1522793  
## BHS\_1685-BHS\_1486 3.946874e-01 -0.712095630 1.501470450 0.9998711  
## BHS\_1686-BHS\_1486 6.346089e-01 -0.692923716 1.962141482 0.9892772  
## BHS\_1687-BHS\_1486 1.735053e+00 0.296917678 3.173187955 0.0028911  
## BHS\_1688-BHS\_1486 6.825756e-01 -0.479738287 1.844889562 0.8979108  
## BHS\_1689-BHS\_1486 2.198862e+00 -2.004543085 6.402267557 0.9683083  
## BHS\_1490-BHS\_1489 -1.317151e+00 -2.515476338 -0.118826156 0.0139332  
## BHS\_1491-BHS\_1489 -1.926899e-01 -1.563027049 1.177647287 1.0000000  
## BHS\_1492-BHS\_1489 1.546570e-01 -1.341653115 1.650967105 1.0000000  
## BHS\_1578-BHS\_1489 -1.883878e-01 -1.508877710 1.132102163 1.0000000  
## BHS\_1579-BHS\_1489 -3.939260e-01 -1.714415888 0.926563984 0.9999949  
## BHS\_1580-BHS\_1489 1.133063e+00 -0.945156129 3.211282295 0.9504077  
## BHS\_1591-BHS\_1489 -2.116352e+00 -6.340014644 2.107310983 0.9807713  
## BHS\_1596-BHS\_1489 -1.056758e+00 -2.238146701 0.124631296 0.1569349  
## BHS\_1599-BHS\_1489 -4.947886e-01 -1.776621668 0.787044425 0.9995198  
## BHS\_1685-BHS\_1489 -1.189274e+00 -2.370662557 -0.007884560 0.0461471  
## BHS\_1686-BHS\_1489 -9.493521e-01 -2.339695234 0.440991063 0.6836339  
## BHS\_1687-BHS\_1489 1.510918e-01 -1.345218262 1.647401958 1.0000000  
## BHS\_1688-BHS\_1489 -9.013853e-01 -2.134951271 0.332180609 0.5448944  
## BHS\_1689-BHS\_1489 6.149013e-01 -3.608761546 4.838564081 1.0000000  
## BHS\_1491-BHS\_1490 1.124461e+00 -0.073863725 2.322786457 0.1004470  
## BHS\_1492-BHS\_1490 1.471808e+00 0.131247533 2.812368951 0.0141761  
## BHS\_1578-BHS\_1490 1.128763e+00 -0.012224143 2.269751090 0.0567638  
## BHS\_1579-BHS\_1490 9.232253e-01 -0.217762321 2.064212912 0.3277627  
## BHS\_1580-BHS\_1490 2.450214e+00 0.481165280 4.419263381 0.0015898  
## BHS\_1591-BHS\_1490 -7.992006e-01 -4.970230029 3.371828861 1.0000000  
## BHS\_1596-BHS\_1490 2.603935e-01 -0.716253288 1.237040377 0.9999994  
## BHS\_1599-BHS\_1490 8.223626e-01 -0.273655180 1.918380431 0.4875410  
## BHS\_1685-BHS\_1490 1.278777e-01 -0.848769144 1.104524521 1.0000000  
## BHS\_1686-BHS\_1490 3.677992e-01 -0.853353221 1.588951543 0.9999939  
## BHS\_1687-BHS\_1490 1.468243e+00 0.127682386 2.808803804 0.0147034  
## BHS\_1688-BHS\_1490 4.157659e-01 -0.623389509 1.454921341 0.9991521  
## BHS\_1689-BHS\_1490 1.932053e+00 -2.238976930 6.103081960 0.9929011  
## BHS\_1492-BHS\_1491 3.473469e-01 -1.148963234 1.843656986 1.0000000  
## BHS\_1578-BHS\_1491 4.302107e-03 -1.316187829 1.324792044 1.0000000  
## BHS\_1579-BHS\_1491 -2.012361e-01 -1.521726007 1.119253865 1.0000000  
## BHS\_1580-BHS\_1491 1.325753e+00 -0.752466248 3.403972176 0.7991601  
## BHS\_1591-BHS\_1491 -1.923662e+00 -6.147324763 2.300000864 0.9943479  
## BHS\_1596-BHS\_1491 -8.640678e-01 -2.045456820 0.317321177 0.5428794  
## BHS\_1599-BHS\_1491 -3.020987e-01 -1.583931787 0.979734306 1.0000000  
## BHS\_1685-BHS\_1491 -9.965837e-01 -2.177972676 0.184805321 0.2489520  
## BHS\_1686-BHS\_1491 -7.566622e-01 -2.147005353 0.633680944 0.9513330  
## BHS\_1687-BHS\_1491 3.437817e-01 -1.152528381 1.840091839 1.0000000  
## BHS\_1688-BHS\_1491 -7.086954e-01 -1.942261390 0.524870490 0.9165921  
## BHS\_1689-BHS\_1491 8.075911e-01 -3.416071665 5.031253962 1.0000000  
## BHS\_1578-BHS\_1492 -3.430448e-01 -1.793842364 1.107752827 0.9999999  
## BHS\_1579-BHS\_1492 -5.485829e-01 -1.999380542 0.902214649 0.9996563  
## BHS\_1580-BHS\_1492 9.784061e-01 -1.184951276 3.141763452 0.9948657  
## BHS\_1591-BHS\_1492 -2.271009e+00 -6.537207081 1.995189430 0.9615669  
## BHS\_1596-BHS\_1492 -1.211415e+00 -2.536858003 0.114028608 0.1294345  
## BHS\_1599-BHS\_1492 -6.494456e-01 -2.065148958 0.766257725 0.9937669  
## BHS\_1685-BHS\_1492 -1.343931e+00 -2.669373859 -0.018487248 0.0422430  
## BHS\_1686-BHS\_1492 -1.104009e+00 -2.618662196 0.410644035 0.5502791  
## BHS\_1687-BHS\_1492 -3.565147e-03 -1.616036522 1.608906228 1.0000000  
## BHS\_1688-BHS\_1492 -1.056042e+00 -2.428195714 0.316111062 0.4324424  
## BHS\_1689-BHS\_1492 4.602443e-01 -3.805953982 4.726442528 1.0000000  
## BHS\_1579-BHS\_1578 -2.055382e-01 -1.474223872 1.063147515 1.0000000  
## BHS\_1580-BHS\_1578 1.321451e+00 -0.724243356 3.367145069 0.7801261  
## BHS\_1591-BHS\_1578 -1.927964e+00 -6.135718479 2.279790365 0.9938674  
## BHS\_1596-BHS\_1578 -8.683699e-01 -1.991557217 0.254817359 0.4226612  
## BHS\_1599-BHS\_1578 -3.064008e-01 -1.534800489 0.921998793 0.9999998  
## BHS\_1685-BHS\_1578 -1.000886e+00 -2.124073073 0.122301502 0.1621474  
## BHS\_1686-BHS\_1578 -7.609643e-01 -2.102203958 0.580275334 0.9259920  
## BHS\_1687-BHS\_1578 3.394796e-01 -1.111317974 1.790277217 1.0000000  
## BHS\_1688-BHS\_1578 -7.129976e-01 -1.890942652 0.464947537 0.8671957  
## BHS\_1689-BHS\_1578 8.032890e-01 -3.404465381 5.011043463 1.0000000  
## BHS\_1580-BHS\_1579 1.526989e+00 -0.518705178 3.572683247 0.4988091  
## BHS\_1591-BHS\_1579 -1.722426e+00 -5.930180301 2.485328543 0.9987935  
## BHS\_1596-BHS\_1579 -6.628318e-01 -1.786019038 0.460355537 0.8933453  
## BHS\_1599-BHS\_1579 -1.008627e-01 -1.329262310 1.127536971 1.0000000  
## BHS\_1685-BHS\_1579 -7.953476e-01 -1.918534895 0.327839681 0.6112666  
## BHS\_1686-BHS\_1579 -5.554261e-01 -1.896665780 0.785813512 0.9985614  
## BHS\_1687-BHS\_1579 5.450178e-01 -0.905779796 1.995815395 0.9996912  
## BHS\_1688-BHS\_1579 -5.074594e-01 -1.685404474 0.670485715 0.9974174  
## BHS\_1689-BHS\_1579 1.008827e+00 -3.198927203 5.216581642 0.9999999  
## BHS\_1591-BHS\_1580 -3.249415e+00 -7.752802383 1.253972556 0.5718783  
## BHS\_1596-BHS\_1580 -2.189821e+00 -4.148608958 -0.231032613 0.0107235  
## BHS\_1599-BHS\_1580 -1.627852e+00 -3.648808681 0.393105272 0.3368765  
## BHS\_1685-BHS\_1580 -2.322337e+00 -4.281124815 -0.363548469 0.0039814  
## BHS\_1686-BHS\_1580 -2.082415e+00 -4.173880017 0.009049679 0.0526434  
## BHS\_1687-BHS\_1580 -9.819712e-01 -3.145328599 1.181386129 0.9946031  
## BHS\_1688-BHS\_1580 -2.034448e+00 -4.025140773 -0.043756055 0.0382688  
## BHS\_1689-BHS\_1580 -5.181618e-01 -5.021549285 3.985225654 1.0000000  
## BHS\_1596-BHS\_1591 1.059594e+00 -3.106601207 5.225789463 0.9999998  
## BHS\_1599-BHS\_1591 1.621563e+00 -2.574220337 5.817346755 0.9995102  
## BHS\_1685-BHS\_1591 9.270783e-01 -3.239117064 5.093273607 1.0000000  
## BHS\_1686-BHS\_1591 1.167000e+00 -3.063196194 5.397195683 0.9999989  
## BHS\_1687-BHS\_1591 2.267444e+00 -1.998754576 6.533641934 0.9622195  
## BHS\_1688-BHS\_1591 1.214966e+00 -2.966323796 5.396256795 0.9999969  
## BHS\_1689-BHS\_1591 2.731253e+00 -3.082595125 8.545101322 0.9914449  
## BHS\_1599-BHS\_1596 5.619691e-01 -0.515505731 1.639443893 0.9693499  
## BHS\_1685-BHS\_1596 -1.325159e-01 -1.088306617 0.823274904 1.0000000  
## BHS\_1686-BHS\_1596 1.074056e-01 -1.097131673 1.311942906 1.0000000  
## BHS\_1687-BHS\_1596 1.207850e+00 -0.117593755 2.533292856 0.1329956  
## BHS\_1688-BHS\_1596 1.553724e-01 -0.864206434 1.174951177 1.0000000  
## BHS\_1689-BHS\_1596 1.671659e+00 -2.494536365 5.837854306 0.9991135  
## BHS\_1685-BHS\_1599 -6.944849e-01 -1.771959750 0.382989875 0.7835439  
## BHS\_1686-BHS\_1599 -4.545635e-01 -1.757761867 0.848634938 0.9999119  
## BHS\_1687-BHS\_1599 6.458805e-01 -0.769822872 2.061583811 0.9942152  
## BHS\_1688-BHS\_1599 -4.065967e-01 -1.541037952 0.727844534 0.9998595  
## BHS\_1689-BHS\_1599 1.109690e+00 -3.086093657 5.305473435 0.9999995  
## BHS\_1686-BHS\_1685 2.399215e-01 -0.964615817 1.444458763 1.0000000  
## BHS\_1687-BHS\_1685 1.340365e+00 0.014922101 2.665808713 0.0436518  
## BHS\_1688-BHS\_1685 2.878882e-01 -0.731690577 1.307467033 0.9999982  
## BHS\_1689-BHS\_1685 1.804175e+00 -2.362020509 5.970370162 0.9972151  
## BHS\_1687-BHS\_1686 1.100444e+00 -0.414209182 2.615097050 0.5572092  
## BHS\_1688-BHS\_1686 4.796675e-02 -1.207786025 1.303719535 1.0000000  
## BHS\_1689-BHS\_1686 1.564253e+00 -2.665942585 5.794449292 0.9997622  
## BHS\_1688-BHS\_1687 -1.052477e+00 -2.424630567 0.319676209 0.4397413  
## BHS\_1689-BHS\_1687 4.638094e-01 -3.802388836 4.730007675 1.0000000  
## BHS\_1689-BHS\_1688 1.516287e+00 -2.665003697 5.697576894 0.9998286

# Just a few combos are significantly different?  
# DID I DO THIS RIGHT???  
  
# 276 combos - 36 of which are significant- 13% are significant  
# Combos that are significantly different:  
# BHS\_1486 - BHS\_1328  
# BHS\_1337 - BHS\_1329  
# 1486 - 1329  
# 1490 - 1329  
# 1596 - 1329  
# 1685 - 1329  
# 1486 - 1330  
# 1490 - 1330  
# 1683 - 1330  
# 1337 - 1333  
# 1486 - 1333  
# 1490 - 1333  
# 1596 - 1333  
# 1685 - 1333  
# 1688 - 1333  
# 1580 - 1337  
# 1486 - 1425  
# 1486 - 1426  
# 1490 - 1426  
# 1685 - 1426  
# 1489 - 1486  
# 1491 - 1486  
# 1492 - 1486  
# 1578 - 1486  
# 1580 - 1486  
# 1687 - 1486  
# 1490 - 1489  
# 1685 - 1489  
# 1492 - 1490  
# 1580 - 1490  
# 1687 - 1490  
# 1685 - 1492  
# 1596 - 1580  
# 1685 - 1580  
# 1688 - 1580  
# 1687 - 1685

I examined whether there is a statistical difference in visitation between ewes and rams within each mountain range. There was no significant difference in visitation between the sexes in any mountain range (Marbles p-value = 0.4642, South Bristols p-value = 0.05788, Nopah p-value = 0.7963, Castle Piute = 0.8208).

I also examined whether there is a statistical different in visitation between bighorn of different ages. There was significantly different visitation between bighorn of different ages in the Marbles (p-value = 1.176e-05), South Bristols (p-value < 2.2e-16), and Nopahs (p-value = 0.01857), but not in the Castle Piutes (p-value = 0.1117). Looking more closely at the numbers, I do not see an age in the Marbles data that is super different than the expected ratio. In the South Bristols, it seems like the visitation of middle-aged sheep (ages 7 through 10) are noteably different, though different ages within this subset have greater or less visitation than what would be expected if all ages had equal visitation. Overall, I don’t think this significance is meaningful, as I don’t know biologically why a 7 year-old sheep would go into water way more often than an 8 year-old sheep.

Just checking, it does make sense to do these separately by mountain range???

# SEX  
  
# Marbles has 23 ewes and 8 rams of collared sheep  
nrow(Marbcutobsrd[Marbcutobsrd$KI\_Sex == "F", ])

## [1] 521

#544 rows for F  
nrow(Marbcutobsrd[Marbcutobsrd$KI\_Sex == "M", ])

## [1] 195

# 201 rows for M  
# nrow(Marbcutobs)  
# Checked that this was the total number of rows - it was.  
chisq.test(x = c(544, 201), p = c(23/31, 8/31))

##   
## Chi-squared test for given probabilities  
##   
## data: c(544, 201)  
## X-squared = 0.53575, df = 1, p-value = 0.4642

# X-squared = 0.53575, df = 1, p-value = 0.4642  
# No significant difference between males and females in Marbles.  
  
# South Bristols has 16 ewes and 8 rams.  
nrow(BrsScutobsrdwithdups[BrsScutobsrdwithdups$KI\_Sex == "F", ])

## [1] 637

# 635 rows for F  
nrow(BrsScutobsrdwithdups[BrsScutobsrdwithdups$KI\_Sex == "M", ])

## [1] 279

# 277 rows for M  
# nrow(BrsScutobsrdwithdups)  
# Checked that this was the total number of rows - it was.  
chisq.test(x = c(635, 277), p = c(16/24, 8/24))

##   
## Chi-squared test for given probabilities  
##   
## data: c(635, 277)  
## X-squared = 3.597, df = 1, p-value = 0.05788

# X-squared = 3.597, df = 1, p-value = 0.05788  
# Though close to significant, still not significant  
  
# Nopah has 5 females and 1 male.  
nrow(nopacutobsrd[nopacutobsrd$KI\_Sex == "F", ])

## [1] 20

# 22 rows for F  
nrow(nopacutobsrd[nopacutobsrd$KI\_Sex == "M", ])

## [1] 5

# 5 rows for M  
# nrow(nopacutobs)  
# correct number of rows  
chisq.test(x = c(22, 5), p = c(5/6, 1/6))

## Warning in chisq.test(x = c(22, 5), p = c(5/6, 1/6)): Chi-squared  
## approximation may be incorrect

##   
## Chi-squared test for given probabilities  
##   
## data: c(22, 5)  
## X-squared = 0.066667, df = 1, p-value = 0.7963

# There is a warning associated with the small sample size but:  
# X-squared = 0.066667, df = 1, p-value = 0.7963  
  
# Castle Piutes has 3 females and 1 male.  
View(cmprcutobs)  
cmprcutobs1 <- cmprcutobs[(cmprcutobs$KI\_Known\_ID == "BHS\_1730") | (cmprcutobs$KI\_Known\_ID == "BHS\_1733") | (cmprcutobs$KI\_Known\_ID == "BHS\_1734") | (cmprcutobs$KI\_Known\_ID == "BHS\_1728"), ]  
nrow(cmprcutobs1[cmprcutobs1$KI\_Sex == "F", ])

## [1] 20

# 22 rows for F  
nrow(cmprcutobs1[cmprcutobs1$KI\_Sex == "M", ])

## [1] 7

# 7 rows for M  
nrow(cmprcutobs1)

## [1] 24

# ONLY 26 ROWS - there are some NA's in sex here!  
# but there really is: 20 F, 6 M  
chisq.test(x = c(20, 6), p = c(3/4, 1/4))

##   
## Chi-squared test for given probabilities  
##   
## data: c(20, 6)  
## X-squared = 0.051282, df = 1, p-value = 0.8208

# X-squared = 0.051282, df = 1, p-value = 0.8208  
  
# There is not a significant difference in visitation by sex in any mountain range.  
  
# AGE  
  
# See BighornAgeSummer2019 (in Danielle's google drive) for age calculations  
Marbcutobsrd$Age2019 <- NA  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1365"] <- 9  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1361"] <- 9  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1362"] <- 8  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1357"] <- 8  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1416"] <- 11  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1421"] <- 9  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1581"] <- 7  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1577"] <- 7  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1574"] <- 12  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1576"] <- 10  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1583"] <- 8  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1584"] <- 9  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1585"] <- 7  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1587"] <- 8  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1582"] <- 7  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1575"] <- 7  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1592"] <- 7  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1594"] <- 9  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1590"] <- 12  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1591"] <- 9  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1682"] <- 4  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1683"] <- 4  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1684"] <- 13  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1588"] <- 8  
Marbcutobsrd$Age2019[Marbcutobsrd$KI\_Known\_ID == "BHS\_1595"] <- 8  
# 1367, 1366, and 1497 were of unknown age  
table(Marbcutobsrd$Age2019[(Marbcutobsrd$KI\_Known\_ID != "BHS\_1367") & (Marbcutobsrd$KI\_Known\_ID != "BHS\_1366") & (Marbcutobsrd$KI\_Known\_ID != "BHS\_1497")])

##   
## 4 7 8 9 10 11 12 13   
## 72 172 128 140 25 16 50 10

# 4 7 8 9 10 11 12 13  
# 76 184 132 150 25 15 51 10  
chisq.test(x = c(76, 184, 132, 150, 25, 15, 51, 10), p = c(2/25, 6/25, 6/25, 6/25, 1/25, 1/25, 2/25, 1/25))

##   
## Chi-squared test for given probabilities  
##   
## data: c(76, 184, 132, 150, 25, 15, 51, 10)  
## X-squared = 34.884, df = 7, p-value = 1.176e-05

# p-value = 1.176e-05  
# But is this really a significant finding? I'm question it.  
  
BrsScutobsrdwithdups$Age2019 <- NA  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1330"] <- 8  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1334"] <- 8  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1333"] <- 9  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1425"] <- 10  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1489"] <- 7  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1491"] <- 10  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1578"] <- 7  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1579"] <- 3  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1596"] <- 7  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1599"] <- 7  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1687"] <- 9  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1486"] <- 8  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1490"] <- 12  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1492"] <- 10  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1428"] <- 8  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1685"] <- 3  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1686"] <- 6  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1688"] <- 2  
BrsScutobsrdwithdups$Age2019[BrsScutobsrdwithdups$KI\_Known\_ID == "BHS\_1689"] <- 10  
table(BrsScutobsrdwithdups$Age2019)

##   
## 2 3 6 7 8 9 10 12   
## 58 103 23 239 120 48 104 54

# 2 3 6 7 8 9 10 12  
 # 58 101 23 239 120 48 103 54  
chisq.test(x = c(58, 101, 23, 239, 120, 48, 103, 54), p = c(1/19, 2/19, 1/19, 4/19, 4/19, 2/19, 4/19, 1/19))

##   
## Chi-squared test for given probabilities  
##   
## data: c(58, 101, 23, 239, 120, 48, 103, 54)  
## X-squared = 109.61, df = 7, p-value < 2.2e-16

# X-squared = 109.61, df = 7, p-value < 2.2e-16  
# Could it be that middle-aged sheep are visiting water more often b/c they are more likely to have lambs make it to the summer? Is that even true???? Though ewes did not visit more than rams  
  
nopacutobsrd$Age2019 <- NA  
nopacutobsrd$Age2019[nopacutobsrd$KI\_Known\_ID == "BHS\_1737"] <- 3  
nopacutobsrd$Age2019[nopacutobsrd$KI\_Known\_ID == "BHS\_1740"] <- 2  
nopacutobsrd$Age2019[nopacutobsrd$KI\_Known\_ID == "BHS\_1741"] <- 8  
nopacutobsrd$Age2019[nopacutobsrd$KI\_Known\_ID == "BHS\_1738"] <- 2  
nopacutobsrd$Age2019[nopacutobsrd$KI\_Known\_ID == "BHS\_1736"] <- 3  
nopacutobsrd$Age2019[nopacutobsrd$KI\_Known\_ID == "BHS\_1739"] <- 5  
table(nopacutobsrd$Age2019)

##   
## 2 3 5 8   
## 14 5 5 1

# 2 3 5 8  
# 16 5 5 1  
chisq.test( x = c(16, 5, 5, 1), p = c(2/6, 2/6, 1/6, 1/6))

## Warning in chisq.test(x = c(16, 5, 5, 1), p = c(2/6, 2/6, 1/6, 1/6)): Chi-  
## squared approximation may be incorrect

##   
## Chi-squared test for given probabilities  
##   
## data: c(16, 5, 5, 1)  
## X-squared = 10, df = 3, p-value = 0.01857

# X-squared = 10, df = 3, p-value = 0.01857  
# warning: Chi-squared approximation may be incorrect  
  
cmprcutobs1$Age2019 <- NA  
cmprcutobs1$Age2019[cmprcutobs1$KI\_Known\_ID == "BHS\_1730"] <- 5  
cmprcutobs1$Age2019[cmprcutobs1$KI\_Known\_ID == "BHS\_1733"] <- 9  
cmprcutobs1$Age2019[cmprcutobs1$KI\_Known\_ID == "BHS\_1734"] <- 7  
cmprcutobs1$Age2019[cmprcutobs1$KI\_Known\_ID == "BHS\_1728"] <- 9  
table(cmprcutobs1$Age2019)

##   
## 5 7 9   
## 4 11 9

# 5 7 9  
# 4 11 11  
chisq.test(x = c(4, 11, 11), p = c(1/4, 1/4, 2/4))

##   
## Chi-squared test for given probabilities  
##   
## data: c(4, 11, 11)  
## X-squared = 4.3846, df = 2, p-value = 0.1117

# X-squared = 4.3846, df = 2, p-value = 0.1117

I looked at what time of day sheep come in to water. I decided to break down the day into comparable four-hour chunks: 1-5, 5-9, 9-13, 13-17, 17-21, 21-1. I decided on these specific time periods (1-5 instead of 0-4 for example) for the following reasons: 1) the hottest time of the day is 3 pm, 2) sunset during the midpoint during the study period (June 23rd) in the Mojave is at 8:05 pm, and 3) sunrise during the midpoint during the study period in the Mojave is at 5:35 am. Visitation to water kept line with bighorn diurnality, with 97% of visits (1652/1698) occurring between 5 am and 9 pm. 45% of visits (773/1698) occurred between 9 am and 1 pm, while another 26% of visits (453/1698) occurred between 5 am and 9 pm. Needless to say, there was a significant difference in the number of visits to water among these time chunks (p-value < 2.2e-16).

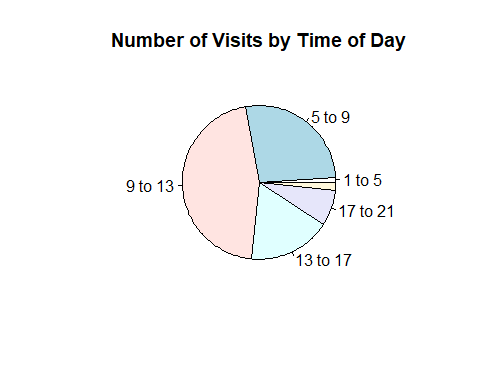
# Decided to break the day down into 1-5, 5-9, 9-13, 13-17, 17-21, 21-1. This is b/c hottest time of day is 3pm, sunset during the midpoint during the study period (June 23rd) in the Mojave is at 8:05 pm, and sunrise during the midpoint during the study period in the Mojave is at 5:35 am.   
  
# First need to put  
  
allrangetime <- rbind(Marbcutobsrd, BrsScutobsrdwithdups, nopacutobsrd, cmprcutobs1)  
allrangetime$timeofday <- allrangetime$KI\_DG\_FirstPhotoTime  
write.csv(allrangetime, 'allrangetime.csv')  
# Because it seemed easier to just transfer the times of the 2nd/3rd/4th visits in the few observations with multiple visits/observations into the same column by hand than write out a complicated code, I did just that.  
allrangetime1 <- read.csv("allrangetime.csv", header = TRUE, sep = ",")  
# View(allrangetime1)  
# Double-checked that everything was correct - it was  
allrangetime1$X.1 <- NULL  
allrangetime1$X.2 <- NULL  
allrangetime1$X.3 <- NULL  
allrangetime1$X.4 <- NULL  
allrangetime1$TimeBreaks <- NA  
allrangetime1$TimeBreaks <- as.character(allrangetime1$TimeBreaks)  
# There are empty cells for allrangetime1$Hour - need to go back through photos and fix these entries!  
library(lubridate)  
allrangetime1$timeofday <- hm(allrangetime1$timeofday)

## Warning in .parse\_hms(..., order = "HM", quiet = quiet): Some strings  
## failed to parse, or all strings are NAs

allrangetime1$Hour <- NA  
allrangetime1$Hour <- hour(allrangetime1$timeofday)  
allrangetime1$TimeBreaks[(allrangetime1$Hour >= 1) & (allrangetime1$Hour <= 4)] <- "1to5"  
allrangetime1$TimeBreaks[(allrangetime1$Hour >= 5) & (allrangetime1$Hour <= 8)] <- "5to9"  
allrangetime1$TimeBreaks[(allrangetime1$Hour >= 9) & (allrangetime1$Hour <= 12)] <- "9to13"  
allrangetime1$TimeBreaks[(allrangetime1$Hour >= 13) & (allrangetime1$Hour <= 16)] <- "13to17"  
allrangetime1$TimeBreaks[(allrangetime1$Hour >= 17) & (allrangetime1$Hour <= 20)] <- "17to21"  
allrangetime1$TimeBreaks[(allrangetime1$Hour >= 21) | (allrangetime1$Hour <= 0)] <- "21to1"  
allrangetime1$TimeBreaks <- as.factor(allrangetime1$TimeBreaks)  
table(allrangetime1$TimeBreaks)

##   
## 13to17 17to21 1to5 21to1 5to9 9to13   
## 294 123 17 25 459 749

# 13to17 17to21 1to5 21to1 5to9 9to13  
# 297 129 19 27 453 773  
slices <- c(19, 453, 773, 297, 129, 27)  
lbls <- c("1 to 5", "5 to 9", "9 to 13", "13 to 17", "17 to 21")  
pie(slices, labels = lbls, main = "Number of Visits by Time of Day")



chisq.test(x = c(19, 453, 773, 297, 129, 27), p = c(4/24, 4/24, 4/24, 4/24, 4/24, 4/24))

##   
## Chi-squared test for given probabilities  
##   
## data: c(19, 453, 773, 297, 129, 27)  
## X-squared = 1512.9, df = 5, p-value < 2.2e-16

# X-squared = 1512.9, df = 5, p-value < 2.2e-16

This is where I am going to leave off my basic analyses of the visitation data for the moment. When I have time, I still want to do the following analyses.

* how much time individual bighorn spend drinking water when they visit (and whether this differs by age, sex)
* whether there is a significant difference in how often ewes with lambs come in vs. ewes without lambs (or ewe groups with lambs come in vs. ewe groups without lambs)
* whether there is a significant difference in how often sheep in large groups come in vs. sheep in small groups come in