```
library(dplyr)

setwd("C:\\Users\\DJ\\Desktop\\fake_occu_data") #set working directory
frogs <- read.csv("raw_survey_data.csv") # read raw frog data into R
sitecoords1 <- read.csv("point_locations.csv") # read coordinate data into R
View(frogs)
unique(frogs$species)

#################################################################################################
#################################################################################################
#                           Preparing the detection covariates                                 #
#################################################################################################
#################################################################################################

# first visit

# removing all data except from visit 1:
frogsV1 <- subset(frogs, visit == 1)

# merge is essentially a vlookup - this links the detection data with the coordinate data:
lookup_1a <- merge(x = sitecoords1, by.x = "point_id", y = frogsV1, by.y = "pointID", all.x = TRUE)
#vlookup

# then sites must be identified as a group - this does not change the APPEARANCE of the
# file, however, is important for summarising ('summarise()') each visit's detection data:
lookup_1b <- group_by(lookup_1a, point_id)

# summarise gives you summary data for each group. In this case, you can choose anything -- all of
# the values are the same; you could choose min(), max(), or whatever - I chose mean() here.
# In any case, this bit of code gives you each detection covariate needed for occupancy modeling
lookup_1cloud <- data.frame(summarise(lookup_1b, cloud = mean(cloud)))
lookup_1julian <- data.frame(summarise(lookup_1b, julian = mean(date)))
lookup_1temp <- data.frame(summarise(lookup_1b, temp = mean(temp)))
lookup_1wind <- data.frame(summarise(lookup_1b, wind = mean(wind)))
lookup_1msss <- data.frame(summarise(lookup_1b, msss = mean(minutes)))
lookup_1noise <- data.frame(summarise(lookup_1b, noise = mean(noise_index)))

# now let's repeat all of the above steps for visits 2 and 3

# second visit

frogsV2 <- subset(frogs, visit == 2) # removing all data except from visit 2

lookup_2a <- merge(x = sitecoords1, by.x = "point_id", y = frogsV2, by.y = "pointID", all.x = TRUE)
#vlookup
lookup_2b <- group_by(lookup_2a, point_id)

lookup_2cloud <- data.frame(summarise(lookup_2b, cloud = mean(cloud)))
lookup_2julian <- data.frame(summarise(lookup_2b, julian = mean(date)))
lookup_2temp <- data.frame(summarise(lookup_2b, temp = mean(temp)))
lookup_2wind <- data.frame(summarise(lookup_2b, wind = mean(wind)))
lookup_2msss <- data.frame(summarise(lookup_2b, msss = mean(minutes)))
lookup_2noise <- data.frame(summarise(lookup_2b, noise = mean(noise_index)))

# third visit

frogsV3 <- subset(frogs, visit == 3) # removing all data except from visit 3

lookup_3a <- merge(x = sitecoords1, by.x = "point_id", y = frogsV3, by.y = "pointID", all.x = TRUE)
#vlookup
lookup_3b <- group_by(lookup_3a, point_id)

lookup_3cloud <- data.frame(summarise(lookup_3b, cloud = mean(cloud)))
```

```
lookup_3julian <- data.frame(summarise(lookup_3b, julian = mean(date)))
lookup_3temp <- data.frame(summarise(lookup_3b, temp = mean(temp)))
lookup_3wind <- data.frame(summarise(lookup_3b, wind = mean(wind)))
lookup_3msss <- data.frame(summarise(lookup_3b, msss = mean(msss)))
lookup_3noise <- data.frame(summarise(lookup_3b, noise = mean(noise_index)))

##################################################

# combining each visit's covariates of each type

point_idVals <- data.frame("point_id" = lookup_1wind$point_id)
windVals <- cbind("wind1" = lookup_1wind$wind, "wind2" = lookup_2wind$wind, "wind3" =
lookup_3wind$wind)
cloudVals <- cbind("cloud1" = lookup_1cloud$cloud, "wind2" = lookup_2cloud$cloud, "wind3" =
lookup_3cloud$cloud)
julianVals <- cbind("julian1" = lookup_1julian$julian, "julian2" = lookup_2julian$julian, "julian3" =
lookup_3julian$julian)
tempVals <- cbind("temp1" = lookup_1temp$temp, "temp2" = lookup_2temp$temp, "temp3" =
lookup_3temp$temp)
msssVals <- cbind("msss1" = lookup_1msss$msss, "msss2" = lookup_2msss$msss, "msss3" =
lookup_3msss$msss)
noiseVals <- cbind("noise1" = lookup_1noise$noise, "noise2" = lookup_2noise$noise, "noise3" =
lookup_3noise$noise)

##################################################

# combining into a single file of detection variables

AllDetCovs <- cbind(point_idVals, windVals, cloudVals, julianVals, tempVals, msssVals, noiseVals)
View(AllDetCovs)

#################################################################################################
#################################################################################################
#                  Creating the frog observation detection history                           #
#################################################################################################
#################################################################################################

frogs100 <- subset(frogs, distance <= 100) # removing detections beyond 100m
frogsV1 <- subset(frogs100, visit == 1) # removing all data except from visit 1
frogsV2 <- subset(frogs100, visit == 2) #  removing all data except from visit 2
frogsV3 <- subset(frogs100, visit == 3) #  removing all data except from visit 3

# begin using the sitecoords dataframe as the "base" and then build in each  visit's detections

focalspecies <- "wood frog"

frogsV1_foc <- subset(frogsV1, species == focalspecies) # subsetting focal species from first visit
frogsV2_foc <- subset(frogsV2, species == focalspecies) # subsetting focal species from second visit
frogsV3_foc <- subset(frogsV3, species == focalspecies) # subsetting focal species from third visit

# first visit
lookup1 <- merge(x = sitecoords1, by.x = "point_id", y = frogsV1_foc, by.y = "pointID", all.x = TRUE)
#vlookup
lookup1$abundance[is.na(lookup1$abundance)] <- 0 # convert non-detections to zero in 'abundance'1
lookup1 <- mutate(lookup1, vis1 = if_else(abundance == 0, 0, 1))
visit1data <- data.frame("v1" = lookup1$vis1, "point1" = lookup1$point_id) # point id added as a
double-check

# second visit
lookup2 <- merge(x = sitecoords1, by.x = "point_id", y = frogsV2_foc, by.y = "pointID", all.x = TRUE)
#vlookup
lookup2$abundance[is.na(lookup2$abundance)] <- 0 # convert non-detections to zero in 'abundance'
lookup2 <- mutate(lookup2, vis2 = if_else(abundance == 0, 0, 1))
visit2data <- data.frame("v2" = lookup2$vis2, "point2" = lookup1$point_id) # point id added as a
double-check
```

```
# third visit
lookup3 <- merge(x = sitecoords1, by.x = "point_id", y = frogsV3_foc, by.y = "pointID", all.x = TRUE)
#vlookup
lookup3$abundance[is.na(lookup3$abundance)] <- 0 # convert non-detections to zero in 'abundance'
lookup3 <- mutate(lookup3, vis3 = if_else(abundance == 0, 0, 1))
visit3data <- data.frame("v3" = lookup3$vis3, "point3" = lookup1$point_id) # point id added as a
double-check

##

# combining detection history
dethistory <- cbind(visit1data, visit2data, visit3data)
sitehistory <- merge(x = sitecoords1, by.x = "point_id", y = dethistory, by.y = "point1", all.x =
TRUE) #vlookup
sitehistory <- select(sitehistory, -pointid, -point2, -point3) # delete trash columns
View(sitehistory)

#################################################################################################
#################################################################################################
#                                           combining                                           #
#################################################################################################
#################################################################################################

FrogOccupancyData <- merge(x = sitehistory, by.x = "point_id", y = AllDetCovs, by.y = "point_id",
all.x = TRUE) #vlookup
View(FrogOccupancyData)

##################
##################
##################
##################
##################

# good info: https://www.neonscience.org/raster-data-r

library(raster)

nlcd_data <- raster(".\\state_college_nlcd.tif") # read raster
nlcd_data <- setMinMax(nlcd_data) # set extent of raster

# re-project the raster
nlcd_data_a <- projectRaster(from = nlcd_data, crs = "+proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23
+lon_0=-96 +x_0=0
                   +y_0=0 +ellps=GRS80 +datum=NAD83 +units=m +no_defs") # reproject into Albers Equal
Area Conic

coords1 <- data.frame(cbind("long" = sitehistory$long, "lat" = sitehistory$lat)) # read in coords of
survey locs

sites1 <- sf::st_as_sf(coords1, coords = c("long", "lat"), crs = 4269) # turn the coords into a
spatial object
# used crs = 4269 because this represents NAD83

sites1 <- sf::st_transform(sites1, crs = "+proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96
+x_0=0
                   +y_0=0 +ellps=GRS80 +datum=NAD83 +units=m +no_defs") # reproject the points into
the same as the raster

#### plotting site

plot(nlcd_data_a, xlim = c(xmin(nlcd_data_a) + 16000, xmax(nlcd_data_a) - 15000),
     ylim = c(ymin(nlcd_data_a) + 11000, ymax(nlcd_data_a)))
plot(nlcd_data_a, add = TRUE)
plot(sites1, add = TRUE)
```

```
#### extracting

extract1 <- raster::extract(x = nlcd_data_a, y = sites1, buffer = 500, df = TRUE)
extract1 <- data.frame(extract1)

#### summarizing the data to be usable

NLCD_freq <- mutate(extract1, NLCD = as.integer(state_college_nlcd)) %>%
        group_by(ID, NLCD) %>%
        summarise(freq = n()) %>%
        ungroup() %>%
        group_by(ID) %>%
        mutate(ncells= sum(freq), prop_land = round(freq/ncells, 2) * 100) %>%
        ungroup()

#### re-assinging site names

sitesPicklist <- data.frame(FalseName= seq(from = 1, to = 41, length.out = 41), OrigName =
sitehistory$point_id)

#### restructuring the dataset

wide <- full_join(NLCD_freq, sitesPicklist, by=c("ID" = "FalseName")) %>%
        dplyr::select(-ID, -freq, -ncells) %>%
        tidyr::spread(NLCD, prop_land, fill = 0)

### Still need to link this back to DetectionData

FinalData <- as.data.frame(full_join(wide, DetectionData, by = c("OrigName" = "point_id")))
View(FinalData)
```