Wood_Turtle_Report.Rmd

Julia Sirois

11/19/2022

Introduction

The goal of this project is to analyze telemetry data collected over the 2022 field season on six wood turtles (3 males and 3 females) in order to look for patterns and differences between their home ranges. There are three relationships I am curious to explore. The first relationship being explored is MCP home range area of males vs the MCP area of females. I hypothesize that male wood turtles will occupy a larger home range than female turtles do.

Why? What is different about their biology that might explain why males use more space?

The second relationship I am going to look at is the whether the home range of each individual is concentrated in a certain area or if there is a lot of variability in the home ranges.

I'm not sure I understand this one. Is this the clusthr? If so, how do you quantify it?

The third relationship I am looking at is the difference between how far away from the stream male turtles occupy compared to how far away female turtles occupy. I believe that female turtles will occupy areas further away from the stream than male turtles.

Why should females be further from the stream than males? Does time of year matter?

In the book Biology and Conservation of the Wood Turtle, Chapter 6: Spatial Ecology and Seasonal Behavior discusses previous examinations of home ranges with the results finding that male wood turtles had larger home ranges but whether it was significant varied between studies. This chapter also mentions that male wood turtles spend more time in streams than female turtles do during the active season and females generally move greater distances from the stream than males do.

Good, but maybe intergrate these findings right into your hyptotheses. Also, we need a formal citation of the book (maybe at the end of the document.)

Getting Started

```
rm(list = ls())
library(tidyverse)
library(ggfortify)
library(here)
library(rgdal)
library(adehabitatHR)
```

Analyses

Pull in data

```
turtles <- read.csv(here("Data", "updated_turtle_locations.csv"))
```

Question 1 Are the home ranges of male wood turtles larger than the home ranges of female wood turtles?

Create a subset consisting of the Turtle IDs, Lats and Longs.

```
turtles.sp <- turtles [c("Turtle_ID", "Latitude", "Longitude")]
```

R is reading the Lats and Longs as numeric data and needs to be told that those columns represent spacial data.

```
library(sp)
coordinates(turtles.sp)<- c("Longitude", "Latitude")
proj4string(turtles.sp)<-CRS("+proj=longlat +datum=WGS84") #sets projection
```

Now convert the spatial coordinates into UTM so the measurements are in meters and not degrees.

```
utm_sp <-spTransform(turtles.sp, CRS("+proj=utm +zone=19 ellps=WGS84")) #reprojects to UTM
```

```
## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition
```

```
turtles_mcp <- mcp(utm_sp, percent = 100)
```

```
## Warning in proj4string(xy): CRS object has comment, which is lost in output
```

```
## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition
```

```
turtles_mcp
```

```
## Object of class "SpatialPolygonsDataFrame" (package sp):
##
## Number of SpatialPolygons:  6
##
## Variables measured:
##        id      area
## L1R2 L1R2  5.002082
## L1R4 L1R4 22.510516
## L3R1 L3R1 43.528142
## L3R3 L3R3  5.874254
## L3R9 L3R9  9.354094
## L9R0 L9R0 15.451445
```

There should be 6 spatial polygons.. not sure what is wrong Ok - I wonder if there is a package conflict? One thing you could do is just refer to getting the mcps from your other code (reference the other file and write them from that file) and then just import them here, already "cooked".

Plot the MCPs

```r
plot(utm_sp, col = as.factor(utm_sp@data$Turtle_ID), pch = 16)
plot(turtles_mcp, col = alpha(1:30, 0.5), add = TRUE)
```

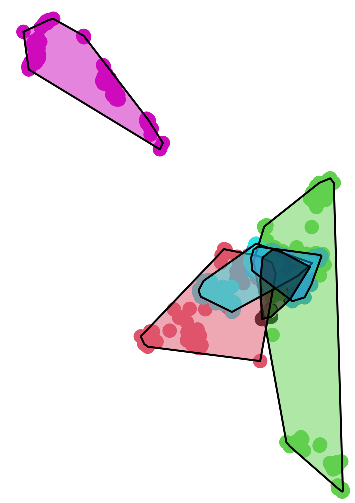


Figure 1: Figure 1. MCP Polygons

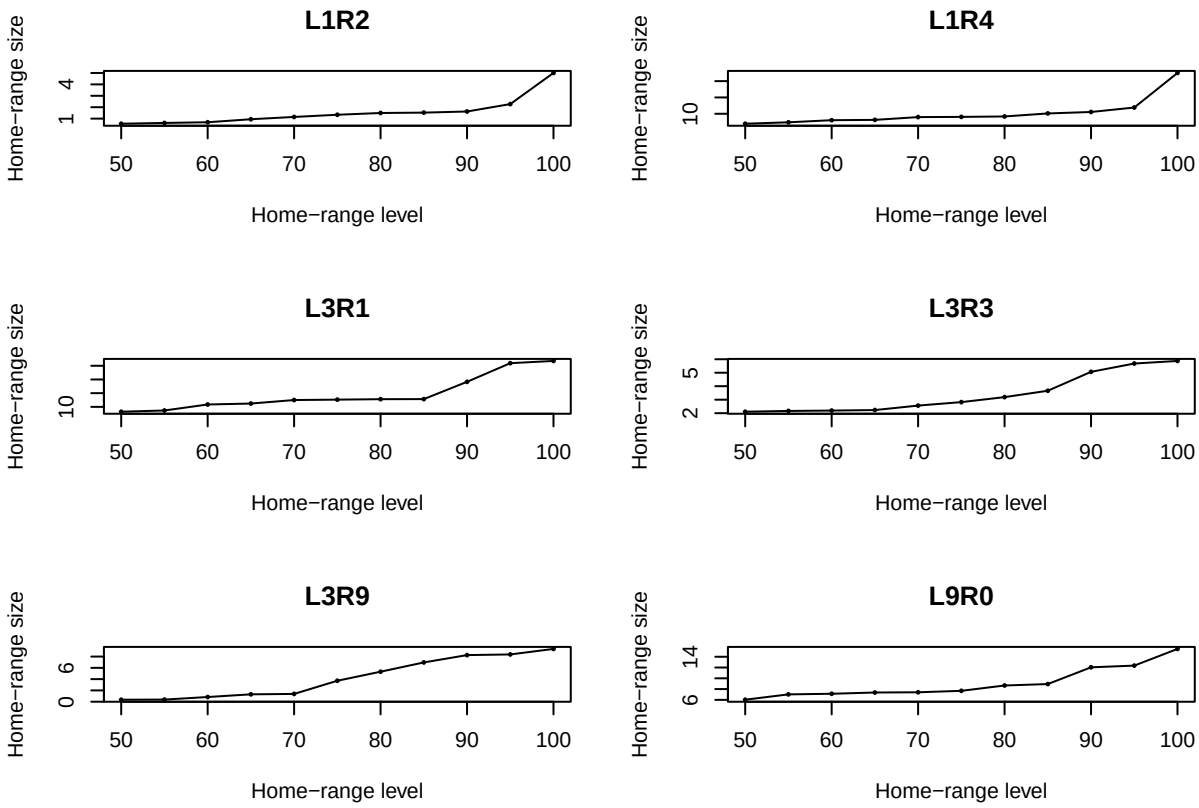
It will be cool to add a figure legend to this telling us what we're looking at.

Calculate the MCP areas

```r
areas <- mcp.area(utm_sp, percent = seq(50, 100, by = 5))
```

```
## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition

## Warning in proj4string(xy): CRS object has comment, which is lost in output
```

```
## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO", prefer_proj
## = prefer_proj): Discarded datum Unknown based on WGS84 ellipsoid in Proj4
## definition
```




```
areas
```

```
##          L1R2      L1R4      L3R1     L3R3      L3R9      L9R0
## 50  0.5542656  6.954713  6.459190 2.108646 0.3531711  6.026639
## 55  0.6204760  7.383645  7.341395 2.165634 0.3745302  7.017832
## 60  0.6830639  8.028361 11.729330 2.193428 0.8305926  7.127859
## 65  0.9474052  8.129758 12.409130 2.231128 1.3055407  7.359935
## 70  1.1468523  8.983196 14.988077 2.562724 1.3801160  7.407324
## 75  1.3376075  9.044232 15.273792 2.819362 3.7149304  7.673697
## 80  1.4935906  9.171617 15.611424 3.189612 5.3167632  8.665802
## 85  1.5250511 10.100023 15.669314 3.662311 6.9593996  8.929796
## 90  1.6243945 10.551843 28.271998 5.065163 8.2591966 12.038415
## 95  2.2725368 11.908777 41.881292 5.682521 8.3787605 12.355634
## 100 5.0020824 22.510516 43.528142 5.874254 9.3540939 15.451445
```

Save the 'areas' subset and import it to add the percentages as a variable.

```
write.csv(areas, here("Data", "turtle_areas.csv"))
```

Import the csv

```
turtle_areas <- read.csv(here("Data", "turtle_areas.csv"))
```

Statistical Analysis Preparation

```
areas_long <- pivot_longer(turtle_areas, cols = L1R2:L9R0, names_to = "Turtle_ID", values_to = "Area") 
areas_long
```

```
## # A tibble: 66 x 3
##        X Turtle_ID  Area
##    <int> <chr>     <dbl>
##  1    50 L1R2      0.554
##  2    50 L1R4      6.95
##  3    50 L3R1      6.46
##  4    50 L3R3      2.11
##  5    50 L3R9      0.353
##  6    50 L9R0      6.03
##  7    55 L1R2      0.620
##  8    55 L1R4      7.38
##  9    55 L3R1      7.34
## 10    55 L3R3      2.17
## # ... with 56 more rows
```

Now let's rename the 'X' column

```
colnames(areas_long) <- c("Area_percentage", "Turtle_ID", "Area")
```

The sex of each turtle is important moving forward. Let's add a column with the appropriate sex for each individual.

```
Turtle_sex <- data.frame(Turtle_sex = c('male', 'female', 'male', 'female', 'female', 'male')) #creates
areas_new <- cbind(areas_long, Turtle_sex) #merges the two dataframes
areas_new
```

```
##    Area_percentage Turtle_ID       Area Turtle_sex
## 1               50      L1R2  0.5542656       male
## 2               50      L1R4  6.9547131     female
## 3               50      L3R1  6.4591897       male
## 4               50      L3R3  2.1086460     female
## 5               50      L3R9  0.3531711     female
## 6               50      L9R0  6.0266392       male
## 7               55      L1R2  0.6204760       male
## 8               55      L1R4  7.3836445     female
## 9               55      L3R1  7.3413952       male
## 10              55      L3R3  2.1656340     female
## 11              55      L3R9  0.3745302     female
## 12              55      L9R0  7.0178324       male
## 13              60      L1R2  0.6830639       male
## 14              60      L1R4  8.0283612     female
## 15              60      L3R1 11.7293299       male
## 16              60      L3R3  2.1934281     female
## 17              60      L3R9  0.8305926     female
## 18              60      L9R0  7.1278586       male
## 19              65      L1R2  0.9474052       male
## 20              65      L1R4  8.1297575     female
## 21              65      L3R1 12.4091303       male
## 22              65      L3R3  2.2311281     female
## 23              65      L3R9  1.3055407     female
## 24              65      L9R0  7.3599349       male
## 25              70      L1R2  1.1468523       male
## 26              70      L1R4  8.9831958     female
## 27              70      L3R1 14.9880770       male
## 28              70      L3R3  2.5627245     female
## 29              70      L3R9  1.3801160     female
```

```
## 30              70      L9R0  7.4073242       male
## 31              75      L1R2  1.3376075       male
## 32              75      L1R4  9.0442319     female
## 33              75      L3R1 15.2737917       male
## 34              75      L3R3  2.8193621     female
## 35              75      L3R9  3.7149304     female
## 36              75      L9R0  7.6736965       male
## 37              80      L1R2  1.4935906       male
## 38              80      L1R4  9.1716168     female
## 39              80      L3R1 15.6114243       male
## 40              80      L3R3  3.1896119     female
## 41              80      L3R9  5.3167632     female
## 42              80      L9R0  8.6658025       male
## 43              85      L1R2  1.5250511       male
## 44              85      L1R4 10.1000234     female
## 45              85      L3R1 15.6693137       male
## 46              85      L3R3  3.6623109     female
## 47              85      L3R9  6.9593996     female
## 48              85      L9R0  8.9297956       male
## 49              90      L1R2  1.6243945       male
## 50              90      L1R4 10.5518430     female
## 51              90      L3R1 28.2719983       male
## 52              90      L3R3  5.0651633     female
## 53              90      L3R9  8.2591966     female
## 54              90      L9R0 12.0384154       male
## 55              95      L1R2  2.2725368       male
## 56              95      L1R4 11.9087775     female
## 57              95      L3R1 41.8812925       male
## 58              95      L3R3  5.6825214     female
## 59              95      L3R9  8.3787605     female
## 60              95      L9R0 12.3556338       male
## 61             100      L1R2  5.0020824       male
## 62             100      L1R4 22.5105163     female
## 63             100      L3R1 43.5281423       male
## 64             100      L3R3  5.8742538     female
## 65             100      L3R9  9.3540939     female
## 66             100      L9R0 15.4514453       male
```

The male turtles are L1R2, L3R1 and L9R0 The female turtles are L1R4, L3R3 and L3R9

Let's separate the areas by the sex of the turtles

```
male_areas <- filter(areas_new, Turtle_ID == "L1R2" | Turtle_ID == "L3R1"| Turtle_ID == "L9R0")
```

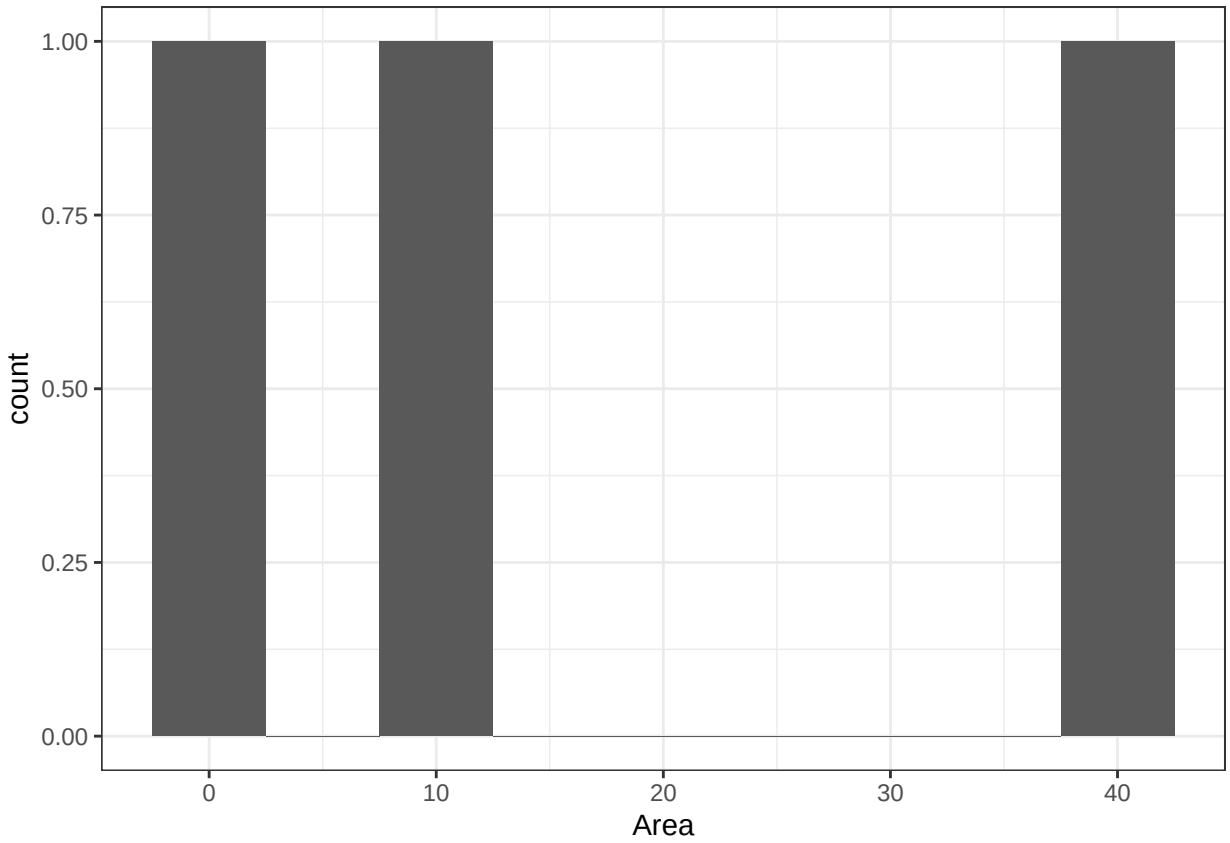
Repeat with the female turtles

```
female_areas <- filter(areas_new, Turtle_ID == "L1R4" | Turtle_ID == "L3R3"| Turtle_ID == "L3R9")
```

Analysis

Visualize the data

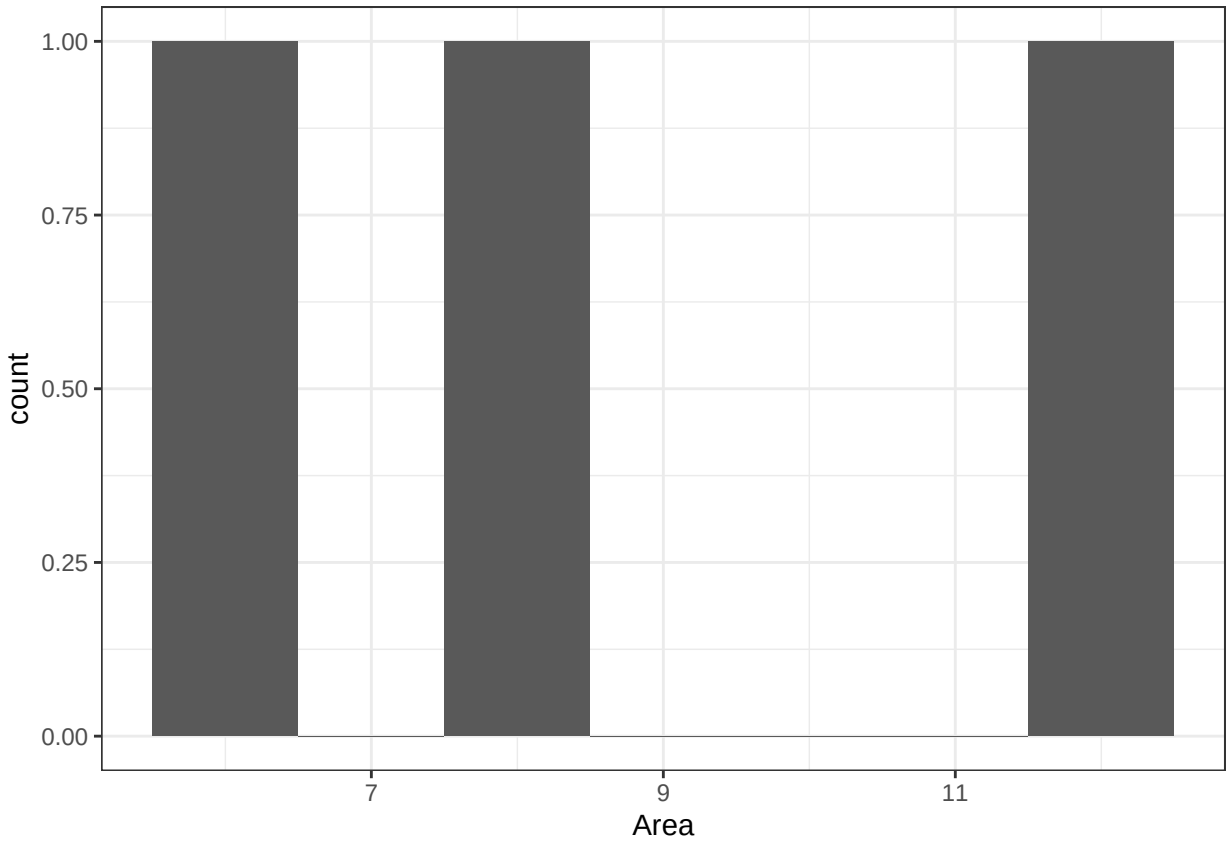
```
male_areas_95 <- filter(male_areas, Area_percentage == "95")
ggplot(male_areas_95, aes(Area)) + geom_histogram(binwidth = 5) + theme_bw()
```




```
summary(male_areas_95)
```

```
##  Area_percentage  Turtle_ID             Area         Turtle_sex
##  Min.   :95       Length:3           Min.   : 2.273   Length:3
##  1st Qu.:95       Class :character   1st Qu.: 7.314   Class :character
##  Median :95       Mode  :character   Median :12.356   Mode  :character
##  Mean   :95                          Mean   :18.836
##  3rd Qu.:95                          3rd Qu.:27.118
##  Max.   :95                          Max.   :41.881
```

```
female_areas_95 <- filter(female_areas, Area_percentage == "95")
ggplot(female_areas_95, aes(Area)) + geom_histogram(binwidth = 1) + theme_bw()
```




```
summary(female_areas_95)
```

```
##  Area_percentage  Turtle_ID             Area         Turtle_sex
##  Min.   :95       Length:3           Min.   : 5.683   Length:3
##  1st Qu.:95       Class :character   1st Qu.: 7.031   Class :character
##  Median :95       Mode  :character   Median : 8.379   Mode  :character
##  Mean   :95                          Mean   : 8.657
##  3rd Qu.:95                          3rd Qu.:10.144
##  Max.   :95                          Max.   :11.909
```

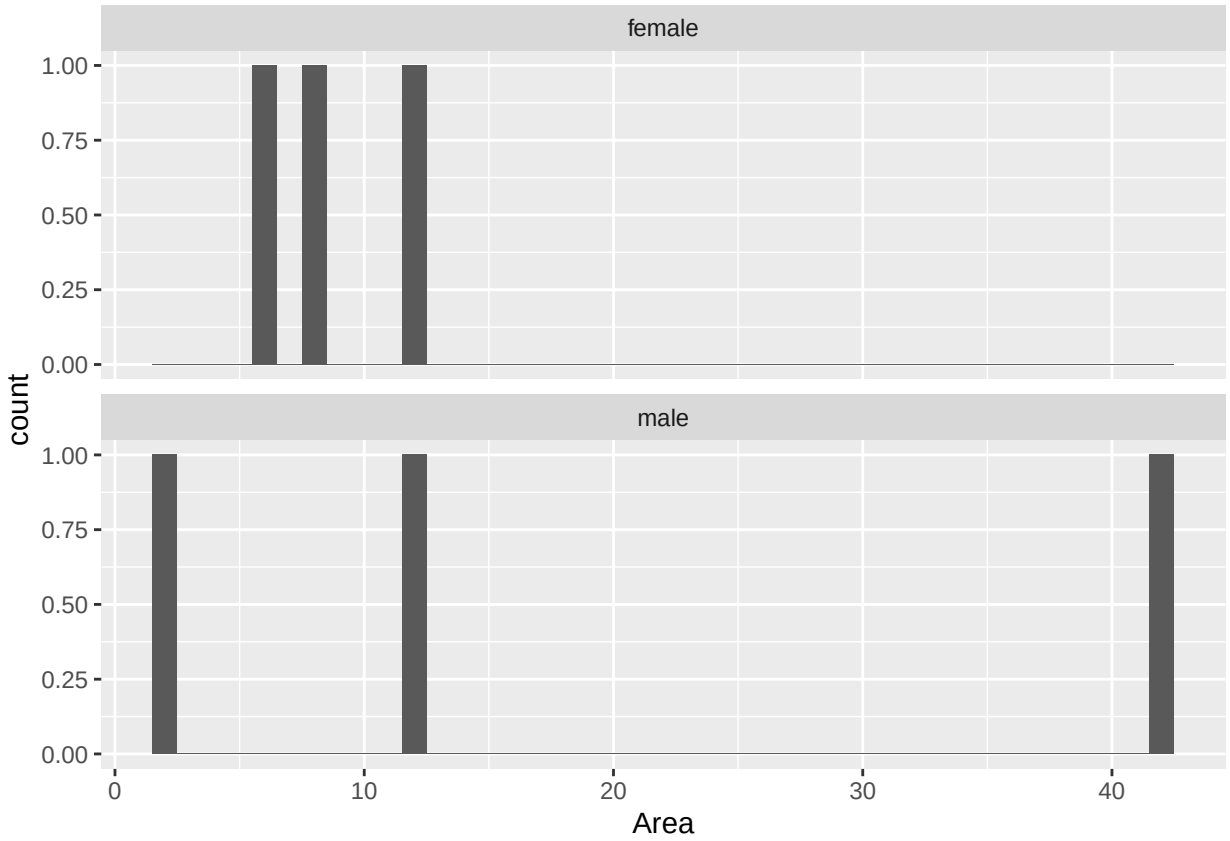
Filter the data so that the only home range area is 95%

```
areas_test <- filter(areas_new, Area_percentage == 95)
```

I'd like to compare the home range size of males vs females. Let's use a t-test

Check out the home ranges of each sex using histograms and the facet_wrap() function.

```
ggplot(areas_test, aes(x = Area)) +
  geom_histogram(binwidth = 1) +
  facet_wrap(~Turtle_sex, ncol = 1) #generates two histograms +
```




```
  theme_bw()
```

```
## List of 94
##  $ line                      :List of 6
##   ..$ colour       : chr "black"
##   ..$ linewidth    : num 0.5
##   ..$ linetype     : num 1
##   ..$ lineend      : chr "butt"
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ rect                      :List of 5
##   ..$ fill         : chr "white"
##   ..$ colour       : chr "black"
##   ..$ linewidth    : num 0.5
##   ..$ linetype     : num 1
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ text                      :List of 11
##   ..$ family       : chr ""
##   ..$ face         : chr "plain"
##   ..$ colour       : chr "black"
##   ..$ size         : num 11
##   ..$ hjust        : num 0.5
##   ..$ vjust        : num 0.5
##   ..$ angle        : num 0
```

```
##   ..$ lineheight   : num 0.9
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ title                     : NULL
##  $ aspect.ratio              : NULL
##  $ axis.title                : NULL
##  $ axis.title.x              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 2.75points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x.top          :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 2.75points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x.bottom       : NULL
##  $ axis.title.y              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : num 90
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 2.75points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.y.left         : NULL
##  $ axis.title.y.right        :List of 11
```

```
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : num -90
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 2.75points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text                :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : chr "grey30"
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 2.2points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x.top          :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 2.2points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
```

```
##  $ axis.text.x.bottom        : NULL
##  $ axis.text.y               :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 1
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 2.2points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.y.left          : NULL
##  $ axis.text.y.right         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 2.2points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.ticks                :List of 6
##   ..$ colour       : chr "grey20"
##   ..$ linewidth    : NULL
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ axis.ticks.x              : NULL
##  $ axis.ticks.x.top          : NULL
##  $ axis.ticks.x.bottom       : NULL
##  $ axis.ticks.y              : NULL
##  $ axis.ticks.y.left         : NULL
##  $ axis.ticks.y.right        : NULL
##  $ axis.ticks.length         : 'simpleUnit' num 2.75points
##   ..- attr(*, "unit")= int 8
##  $ axis.ticks.length.x       : NULL
##  $ axis.ticks.length.x.top   : NULL
##  $ axis.ticks.length.x.bottom: NULL
##  $ axis.ticks.length.y       : NULL
##  $ axis.ticks.length.y.left  : NULL
##  $ axis.ticks.length.y.right : NULL
##  $ axis.line                 : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
```

```
##  $ axis.line.x              : NULL
##  $ axis.line.x.top          : NULL
##  $ axis.line.x.bottom       : NULL
##  $ axis.line.y              : NULL
##  $ axis.line.y.left         : NULL
##  $ axis.line.y.right        : NULL
##  $ legend.background        :List of 5
##   ..$ fill         : NULL
##   ..$ colour       : logi NA
##   ..$ linewidth    : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ legend.margin            : 'margin' num [1:4] 5.5points 5.5points 5.5points 5.5points
##   ..- attr(*, "unit")= int 8
##  $ legend.spacing           : 'simpleUnit' num 11points
##   ..- attr(*, "unit")= int 8
##  $ legend.spacing.x         : NULL
##  $ legend.spacing.y         : NULL
##  $ legend.key               :List of 5
##   ..$ fill         : chr "white"
##   ..$ colour       : logi NA
##   ..$ linewidth    : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ legend.key.size          : 'simpleUnit' num 1.2lines
##   ..- attr(*, "unit")= int 3
##  $ legend.key.height        : NULL
##  $ legend.key.width         : NULL
##  $ legend.text              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.text.align        : NULL
##  $ legend.title             :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
```

```
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.title.align        : NULL
##  $ legend.position           : chr "right"
##  $ legend.direction          : NULL
##  $ legend.justification      : chr "center"
##  $ legend.box                : NULL
##  $ legend.box.just           : NULL
##  $ legend.box.margin         : 'margin' num [1:4] 0cm 0cm 0cm 0cm
##   ..- attr(*, "unit")= int 1
##  $ legend.box.background     : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ legend.box.spacing        : 'simpleUnit' num 11points
##   ..- attr(*, "unit")= int 8
##  $ panel.background          :List of 5
##   ..$ fill         : chr "white"
##   ..$ colour       : logi NA
##   ..$ linewidth    : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ panel.border              :List of 5
##   ..$ fill         : logi NA
##   ..$ colour       : chr "grey20"
##   ..$ linewidth    : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ panel.spacing             : 'simpleUnit' num 5.5points
##   ..- attr(*, "unit")= int 8
##  $ panel.spacing.x           : NULL
##  $ panel.spacing.y           : NULL
##  $ panel.grid                :List of 6
##   ..$ colour       : chr "grey92"
##   ..$ linewidth    : NULL
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ panel.grid.major          : NULL
##  $ panel.grid.minor          :List of 6
##   ..$ colour       : NULL
##   ..$ linewidth    : 'rel' num 0.5
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ panel.grid.major.x        : NULL
##  $ panel.grid.major.y        : NULL
##  $ panel.grid.minor.x        : NULL
##  $ panel.grid.minor.y        : NULL
```

```
##  $ panel.ontop               : logi FALSE
##  $ plot.background           :List of 5
##   ..$ fill         : NULL
##   ..$ colour       : chr "white"
##   ..$ linewidth    : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ plot.title                :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 1.2
##   ..$ hjust        : num 0
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 5.5points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.title.position       : chr "panel"
##  $ plot.subtitle             :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 5.5points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.caption              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : num 1
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 5.5points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.caption.position     : chr "panel"
##  $ plot.tag                  :List of 11
##   ..$ family       : NULL
```

```
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 1.2
##   ..$ hjust        : num 0.5
##   ..$ vjust        : num 0.5
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.tag.position         : chr "topleft"
##  $ plot.margin               : 'margin' num [1:4] 5.5points 5.5points 5.5points 5.5points
##   ..- attr(*, "unit")= int 8
##  $ strip.background          :List of 5
##   ..$ fill         : chr "grey85"
##   ..$ colour       : chr "grey20"
##   ..$ linewidth    : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ strip.background.x        : NULL
##  $ strip.background.y        : NULL
##  $ strip.clip                : chr "inherit"
##  $ strip.placement           : chr "inside"
##  $ strip.text                :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : chr "grey10"
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 4.4points 4.4points 4.4points 4.4points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.text.x              : NULL
##  $ strip.text.y              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : num -90
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.switch.pad.grid     : 'simpleUnit' num 2.75points
```

```
##   ..- attr(*, "unit")= int 8
##  $ strip.switch.pad.wrap     : 'simpleUnit' num 2.75points
##   ..- attr(*, "unit")= int 8
##  $ strip.text.y.left         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : num 90
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi TRUE
##  - attr(*, "validate")= logi TRUE
```

Complete a t-test

```
ttest_MCP <- t.test(Area ~ Turtle_sex, data = areas_test)
ttest_MCP
```

```
##
##  Welch Two Sample t-test
##
## data:  Area by Turtle_sex
## t = -0.84688, df = 2.092, p-value = 0.4828
## alternative hypothesis: true difference in means between group female and group male is not equal to
## 95 percent confidence interval:
##  -59.77966  39.42006
## sample estimates:
## mean in group female   mean in group male
##             8.656686            18.836488
```

Results

There is no significant difference between the home range sizes of male and female turtles in this sample p = 0.4756

Question 2 Do individuals have consistent home range sizes or are they greatly variable?

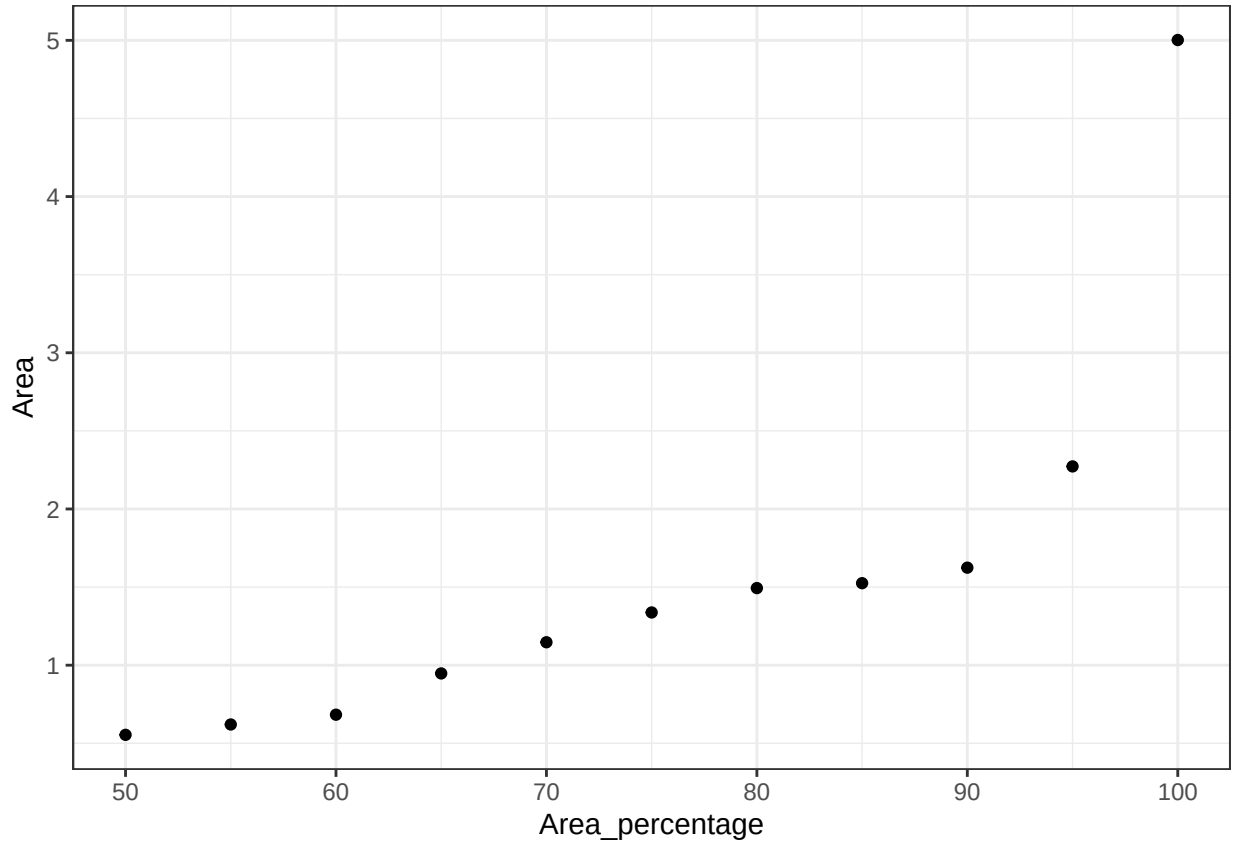
Generate subsets for each individual

```
L1R2 <- filter(areas_new, Turtle_ID == 'L1R2')
L1R4 <- filter(areas_new, Turtle_ID == 'L1R4')
L3R1 <- filter(areas_new, Turtle_ID == 'L3R1')
L3R3 <- filter(areas_new, Turtle_ID == 'L3R3')
L3R9 <- filter(areas_new, Turtle_ID == 'L3R9')
L9R0 <- filter(areas_new, Turtle_ID == 'L9R0')
```

Plot the relationships

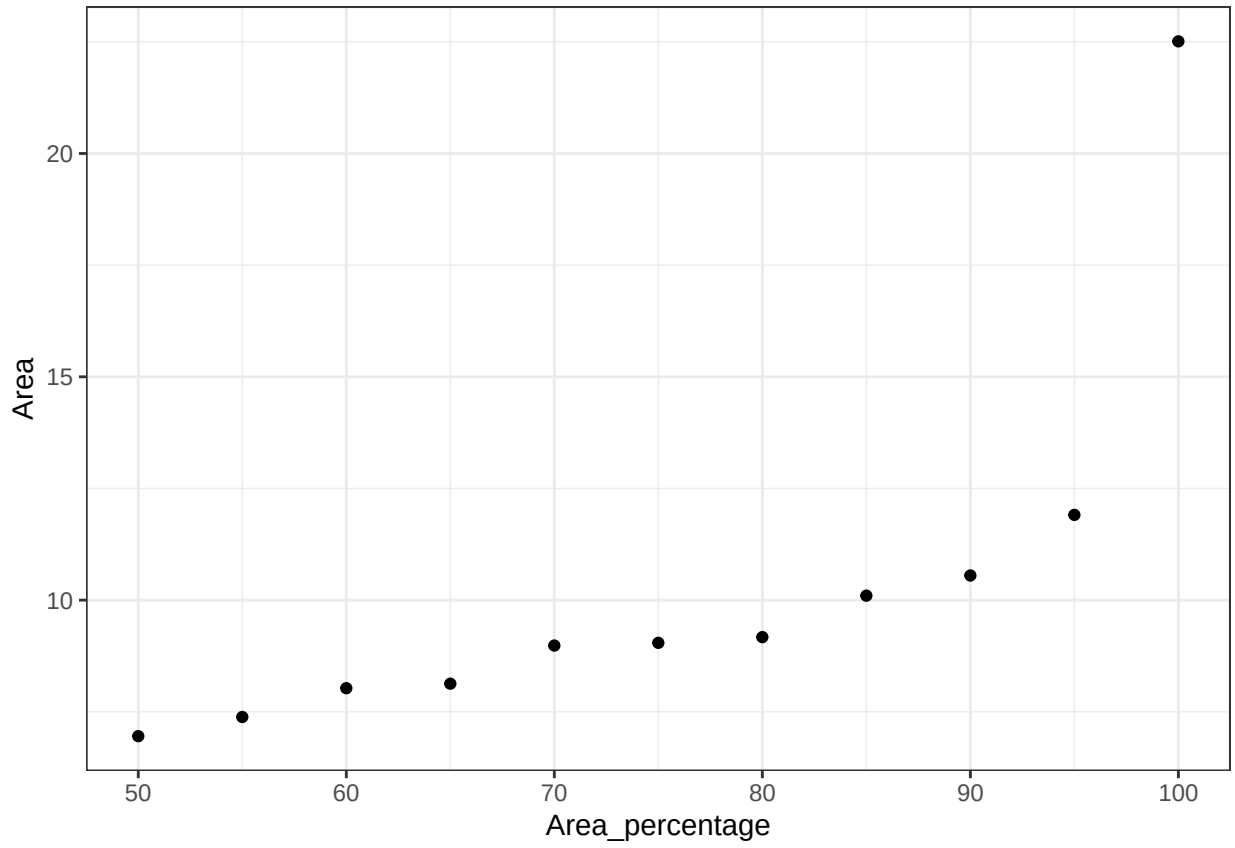
L1R2:

```
ggplot2::ggplot(L1R2, aes(Area_percentage, Area)) +
         geom_point() +
         theme_bw()
```



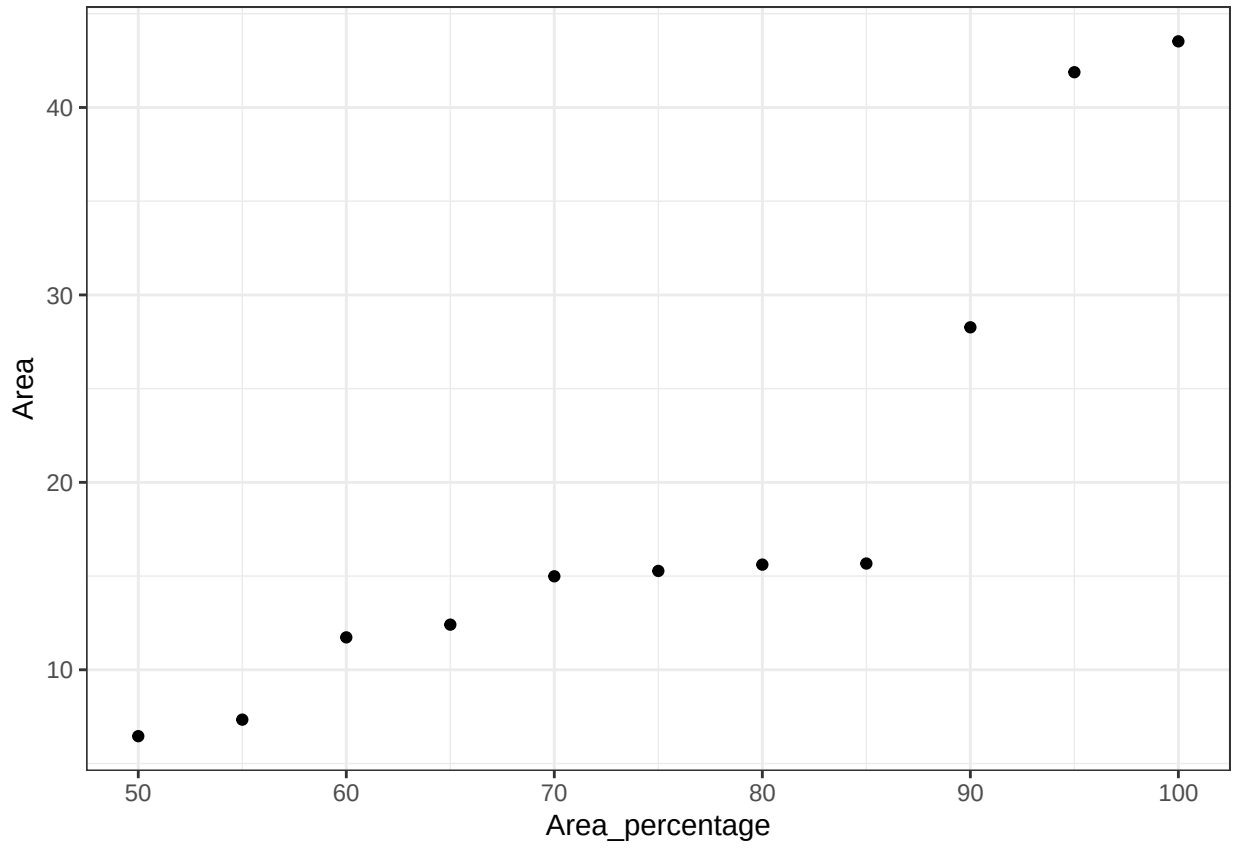
L1R4:

```
ggplot2::ggplot(L1R4, aes(Area_percentage, Area)) +
         geom_point() +
         theme_bw()
```



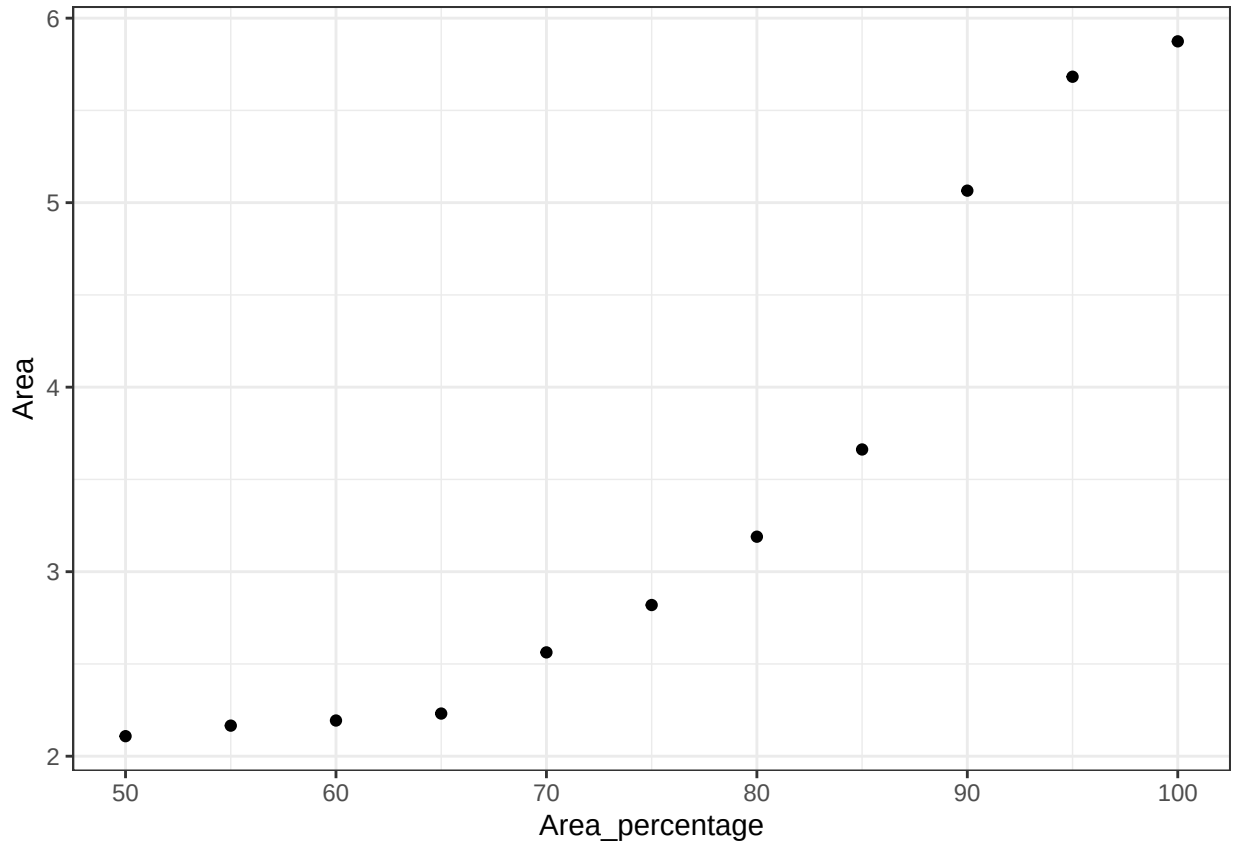
L3R1:

```
ggplot2::ggplot(L3R1, aes(Area_percentage, Area)) +
         geom_point() +
         theme_bw()
```



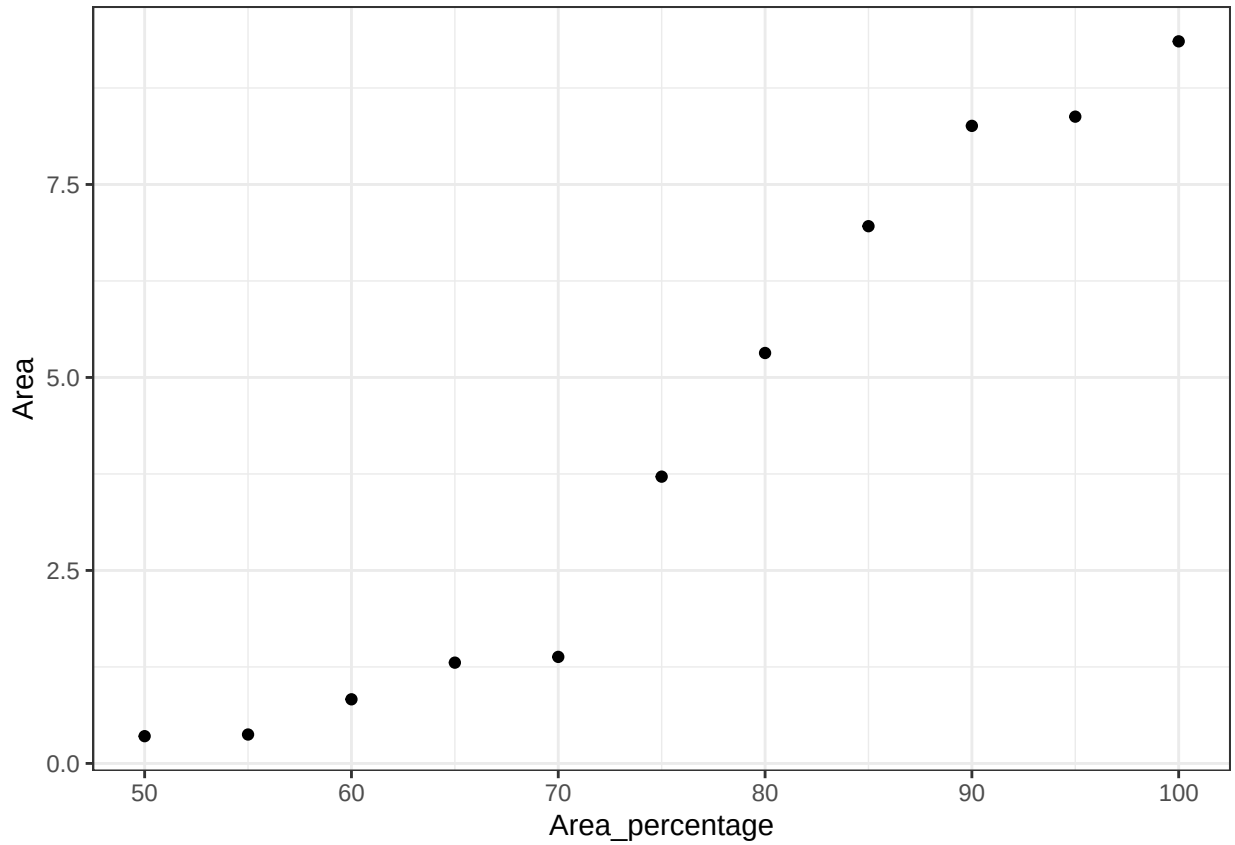
L3R3:

```
ggplot2::ggplot(L3R3, aes(Area_percentage, Area)) +
         geom_point() +
         theme_bw()
```



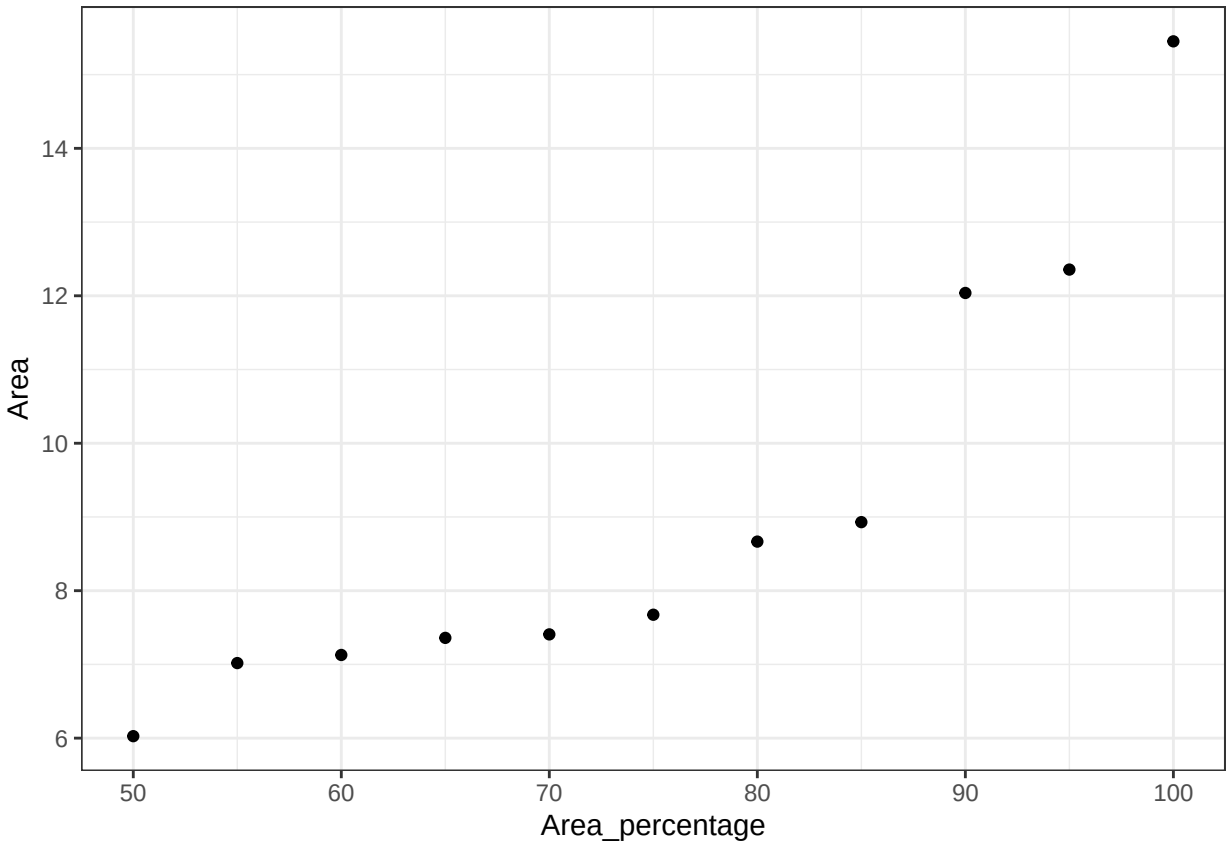
L3R9:

```
ggplot2::ggplot(L3R9, aes(Area_percentage, Area)) +
         geom_point() +
         theme_bw()
```



L9R0:

```
ggplot2::ggplot(L9R0, aes(Area_percentage, Area)) +
         geom_point() +
         theme_bw()
```



Set up a linear model

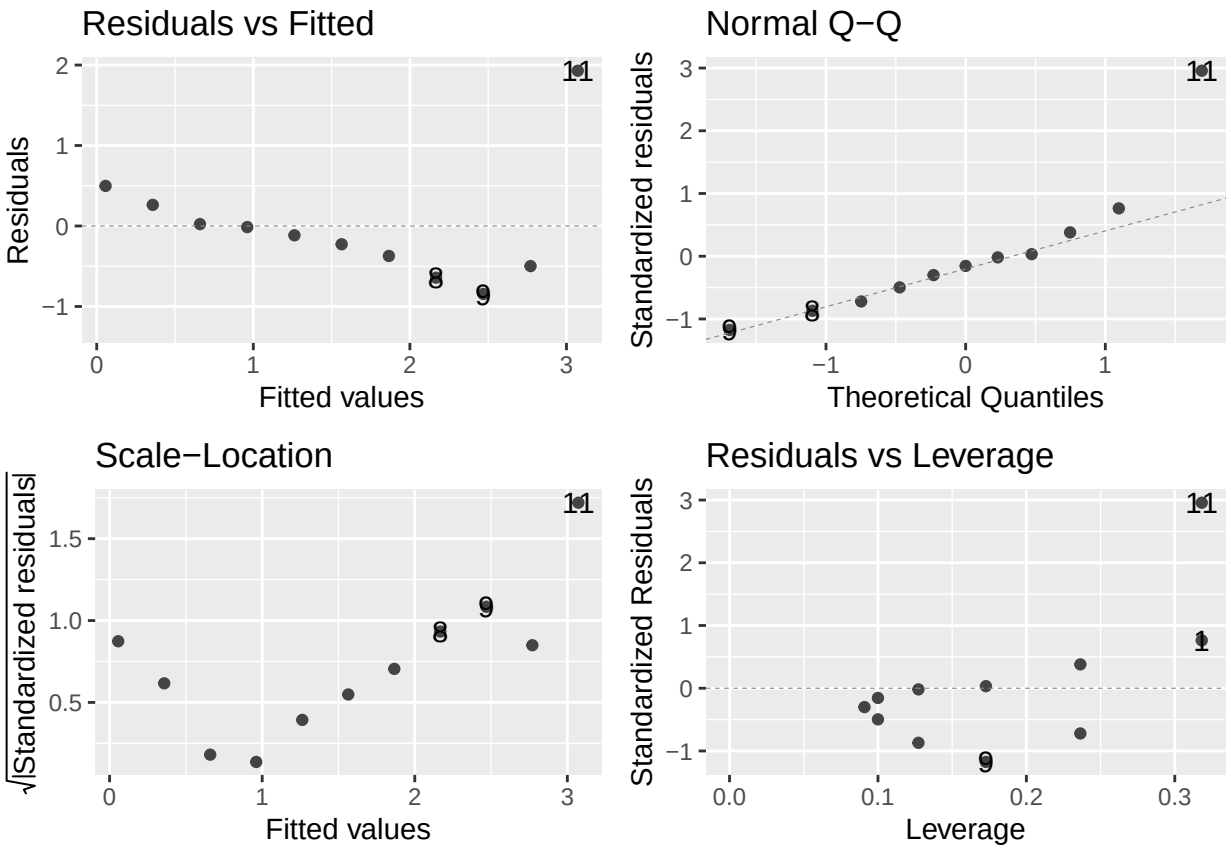
```
L1R2_lm <- lm(Area ~ Area_percentage, data = L1R2)
L1R4_lm <- lm(Area ~ Area_percentage, data = L1R4)
L3R1_lm <- lm(Area ~ Area_percentage, data = L3R1)
L3R3_lm <- lm(Area ~ Area_percentage, data = L3R3)
L3R9_lm <- lm(Area ~ Area_percentage, data = L3R9)
L9R0_lm <- lm(Area ~ Area_percentage, data = L9R0)
```

I'm not sure I get what this is testing. Area is the home range area , right? Are you trying to see how much different the 50% home range is to the 75% for example? If so, it might make sense to determine the difference in areas with each step and then compare the mean differences, rather than the areas themselves. Let's talk about this. Have you seen an example of this sort of analysis someplace?

Check the assumptions

```
autoplot(L1R2_lm, smooth.colour = NA)
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).

## Warning: Removed 11 rows containing missing values (`geom_line()`).

## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

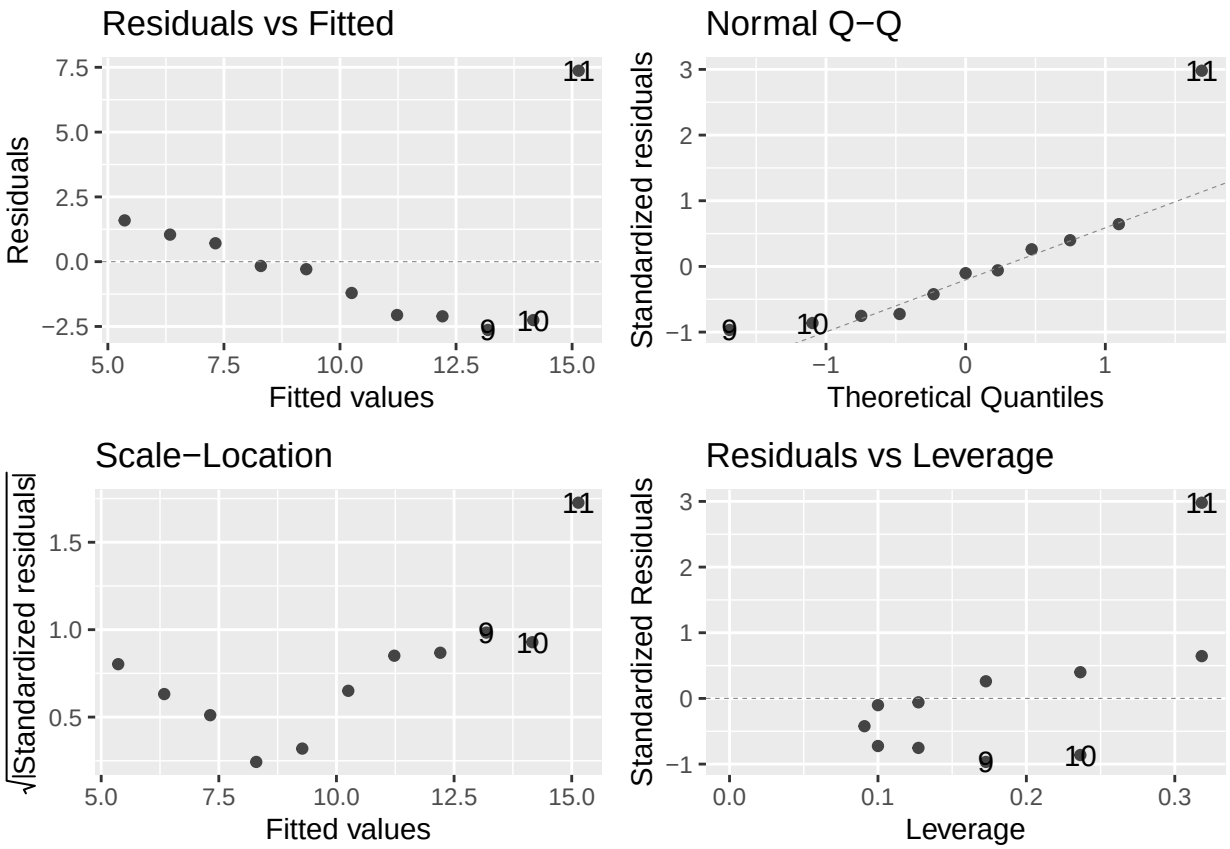



```
autoplot(L1R4_lm, smooth.colour = NA)
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

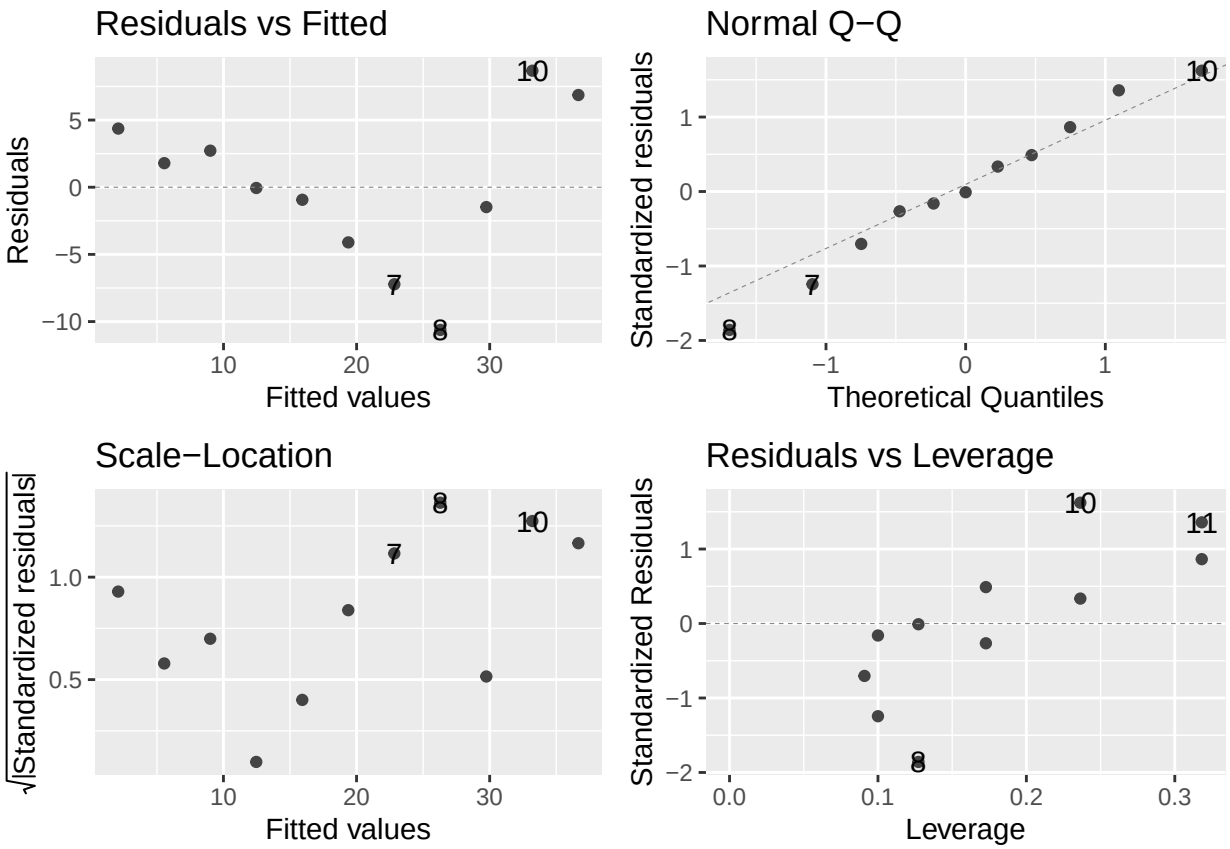



```
autoplot(L3R1_lm, smooth.colour = NA)
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

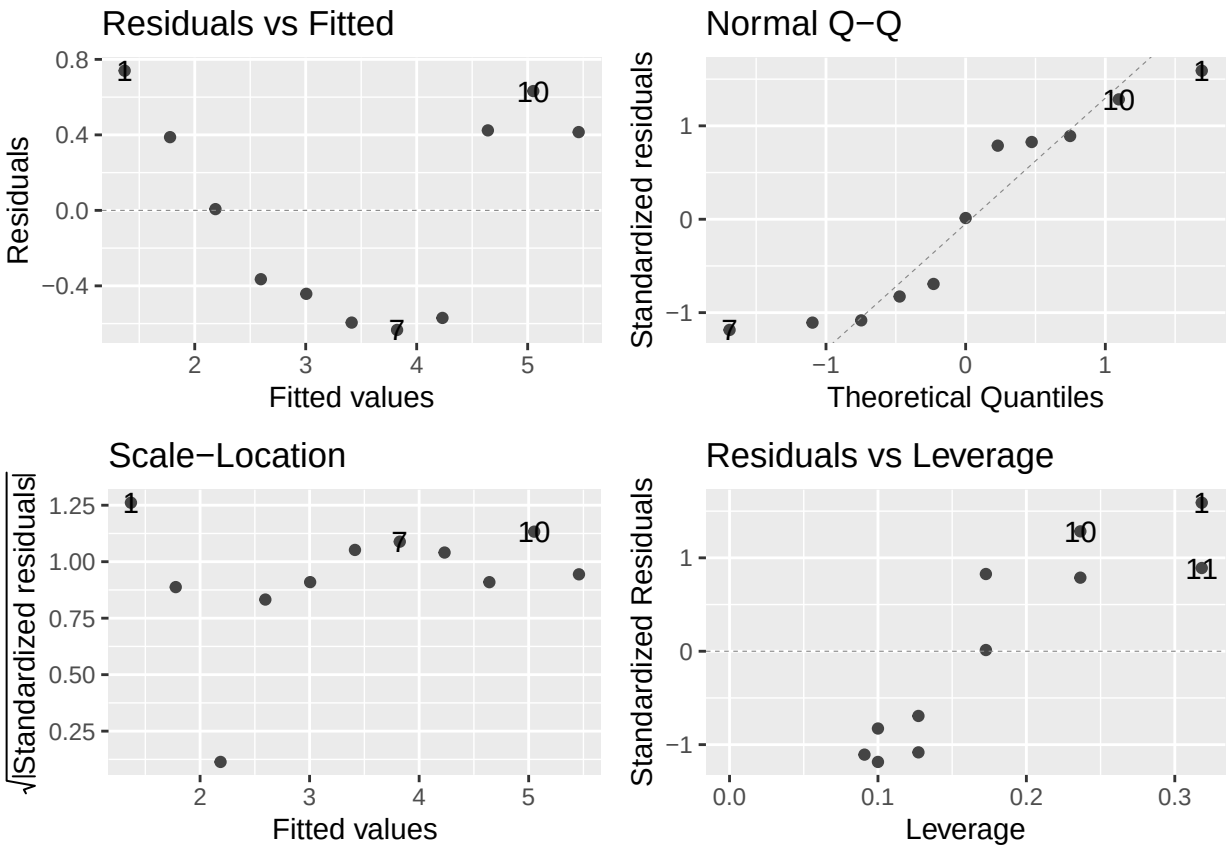



```
autoplot(L3R3_lm, smooth.colour = NA)
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

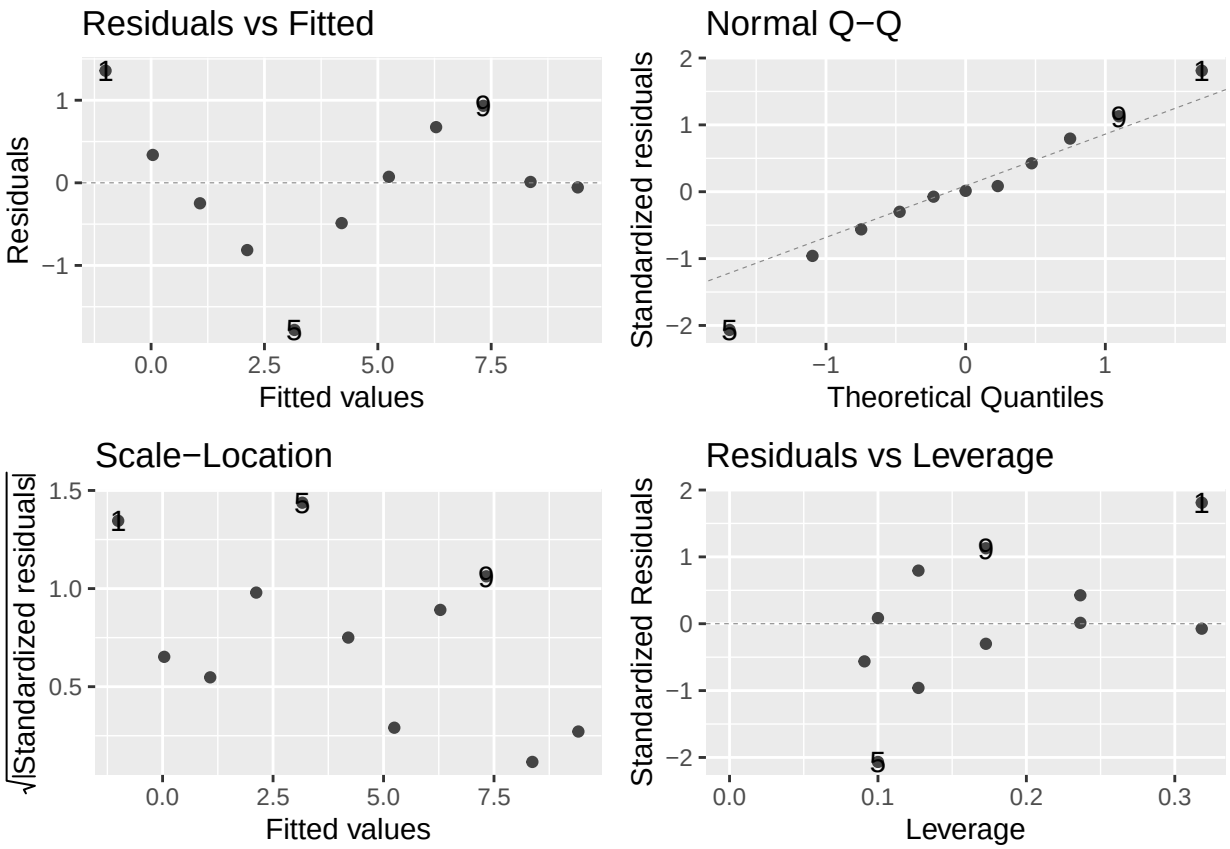



```
autoplot(L3R9_lm, smooth.colour = NA)
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

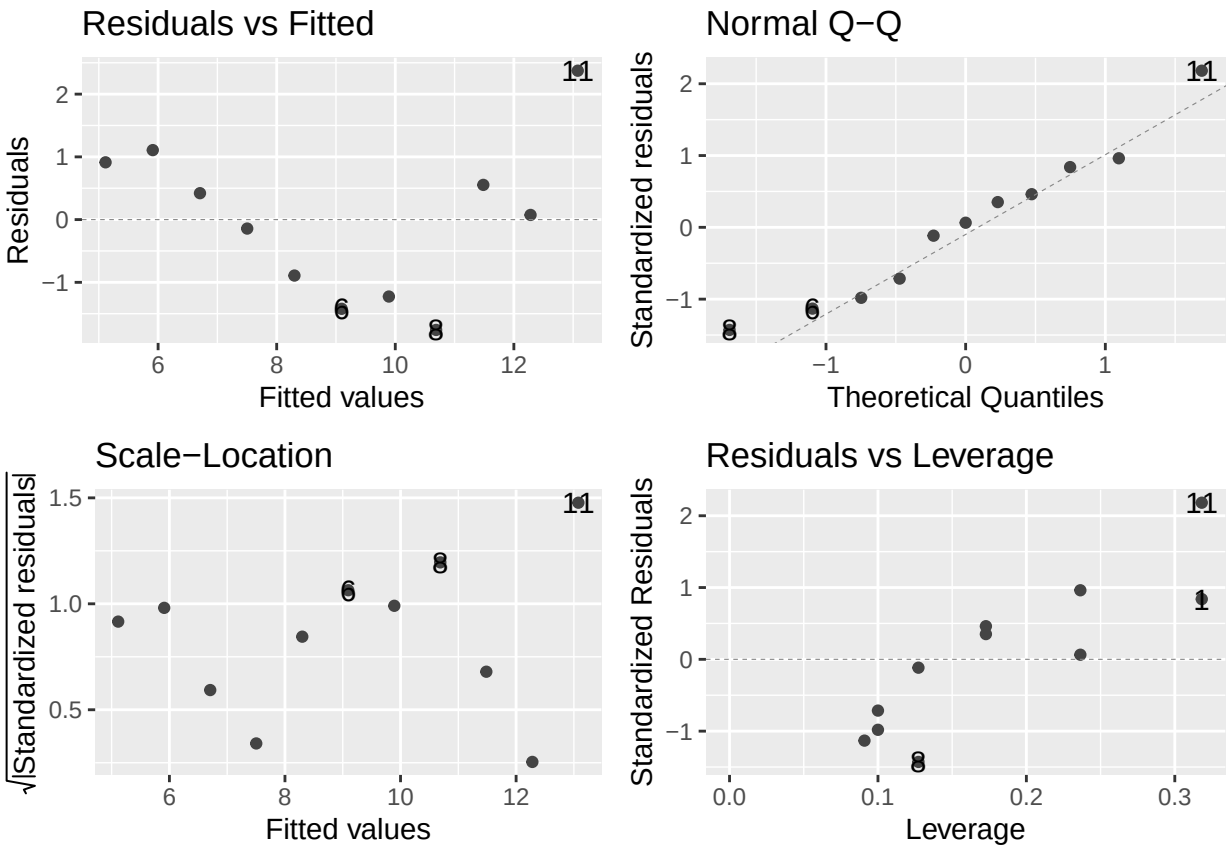



```
autoplot(L9R0_lm, smooth.colour = NA)
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 11 rows containing missing values (`geom_line()`).
```



Interpret

L1R2:

```
anova(L1R2_lm)
```

```
## Analysis of Variance Table
##
## Response: Area
##                 Df  Sum Sq Mean Sq F value   Pr(>F)
## Area_percentage  1 10.0043 10.0043  16.022 0.003097 **
## Residuals        9  5.6196  0.6244
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

L1R4:

```
anova(L1R4_lm)
```

```
## Analysis of Variance Table
##
## Response: Area
##                 Df  Sum Sq Mean Sq F value   Pr(>F)
## Area_percentage  1 105.211 105.211  11.733 0.007563 **
## Residuals        9  80.702   8.967
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

L3R1:

```
anova(L3R1_lm)
```

```
## Analysis of Variance Table
##
## Response: Area
##                 Df Sum Sq Mean Sq F value   Pr(>F)
## Area_percentage  1 1314.6 1314.64  35.109 0.000222 ***
## Residuals        9  337.0   37.44
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

L3R3:

```
anova(L3R3_lm)
```

```
## Analysis of Variance Table
##
## Response: Area
##                 Df  Sum Sq Mean Sq F value    Pr(>F)
## Area_percentage  1 18.4091 18.4091  57.974 3.278e-05 ***
## Residuals        9  2.8579  0.3175
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

L3R9:

```
anova(L3R9_lm)
```

```
## Analysis of Variance Table
##
## Response: Area
##                 Df  Sum Sq Mean Sq F value    Pr(>F)
## Area_percentage  1 119.292 119.292  144.62 7.559e-07 ***
## Residuals        9   7.424   0.825
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

L9R0:

```
anova(L9R0_lm)
```

```
## Analysis of Variance Table
##
## Response: Area
##                 Df Sum Sq Mean Sq F value    Pr(>F)
## Area_percentage  1  69.77  69.770  40.201 0.0001344 ***
## Residuals        9  15.62   1.736
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Summary tables:

L1R2

```
summary(L1R2_lm)
```

```
##
## Call:
## lm(formula = Area ~ Area_percentage, data = L1R2)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8446 -0.4352 -0.1159  0.1430  1.9299
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -2.95934    1.15496  -2.562   0.0306 *
## Area_percentage  0.06032    0.01507   4.003   0.0031 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7902 on 9 degrees of freedom
## Multiple R-squared:  0.6403, Adjusted R-squared:  0.6004
## F-statistic: 16.02 on 1 and 9 DF,  p-value: 0.003097
```

L1R4

```
summary(L1R4_lm)
```

```
##
## Call:
## lm(formula = Area ~ Area_percentage, data = L1R4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6336 -2.0827 -0.2903  0.8775  7.3690
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -4.4183     4.3768  -1.009  0.33911
## Area_percentage   0.1956     0.0571   3.425  0.00756 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.994 on 9 degrees of freedom
## Multiple R-squared:  0.5659, Adjusted R-squared:  0.5177
## F-statistic: 11.73 on 1 and 9 DF,  p-value: 0.007563
```

L3R1

```
summary(L3R1_lm)
```

```
##
## Call:
## lm(formula = Area ~ Area_percentage, data = L3R1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.6233  -2.7912  -0.0552   3.5440   8.6746
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -32.4774     8.9439  -3.631 0.005475 **
## Area_percentage   0.6914     0.1167   5.925 0.000222 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.119 on 9 degrees of freedom
## Multiple R-squared:  0.796,  Adjusted R-squared:  0.7733
## F-statistic: 35.11 on 1 and 9 DF,  p-value: 0.000222
```

L3R3

```
summary(L3R3_lm)
```

```
##
## Call:
## lm(formula = Area ~ Area_percentage, data = L3R3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.63355 -0.50610  0.00663  0.41927  0.74003
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -2.72230    0.82364  -3.305  0.00915 **
## Area_percentage  0.08182    0.01075   7.614 3.28e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5635 on 9 degrees of freedom
## Multiple R-squared:  0.8656, Adjusted R-squared:  0.8507
## F-statistic: 57.97 on 1 and 9 DF,  p-value: 3.278e-05
```

L3R9

```
summary(L3R9_lm)
```

```
##
## Call:
## lm(formula = Area ~ Area_percentage, data = L3R9)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.78097 -0.36763  0.01078  0.50588  1.35760
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -11.41822    1.32746  -8.602 1.23e-05 ***
## Area_percentage   0.20828    0.01732  12.026 7.56e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9082 on 9 degrees of freedom
## Multiple R-squared:  0.9414, Adjusted R-squared:  0.9349
## F-statistic: 144.6 on 1 and 9 DF,  p-value: 7.559e-07
```

L9R0

```
summary(L9R0_lm)
```

```
##
## Call:
```

```
## lm(formula = Area ~ Area_percentage, data = L9R0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.75888 -1.05929  0.07414  0.73308  2.37354
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -2.85030    1.92554   -1.48 0.172937
## Area_percentage  0.15928    0.02512    6.34 0.000134 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.317 on 9 degrees of freedom
## Multiple R-squared:  0.8171, Adjusted R-squared:  0.7968
## F-statistic:  40.2 on 1 and 9 DF,  p-value: 0.0001344
```

Results: All individuals hava a homerange with a significant amount of variability (p = 0.00)

Question 3 Do female wood turtles occupy areas further from the stream than males do?

Check column names

```
names(turtles)
```

```
##  [1] "OID_"        "Date"        "Time"        "Latitude"    "Longitude"
##  [6] "Turtle_ID"   "Turtle_sex"  "Altitude"    "Duration"    "Temperature"
## [11] "Voltage"     "DOP"         "Satellites"  "EST_Time"    "NEAR_FID"
## [16] "NEAR_DIST"
```

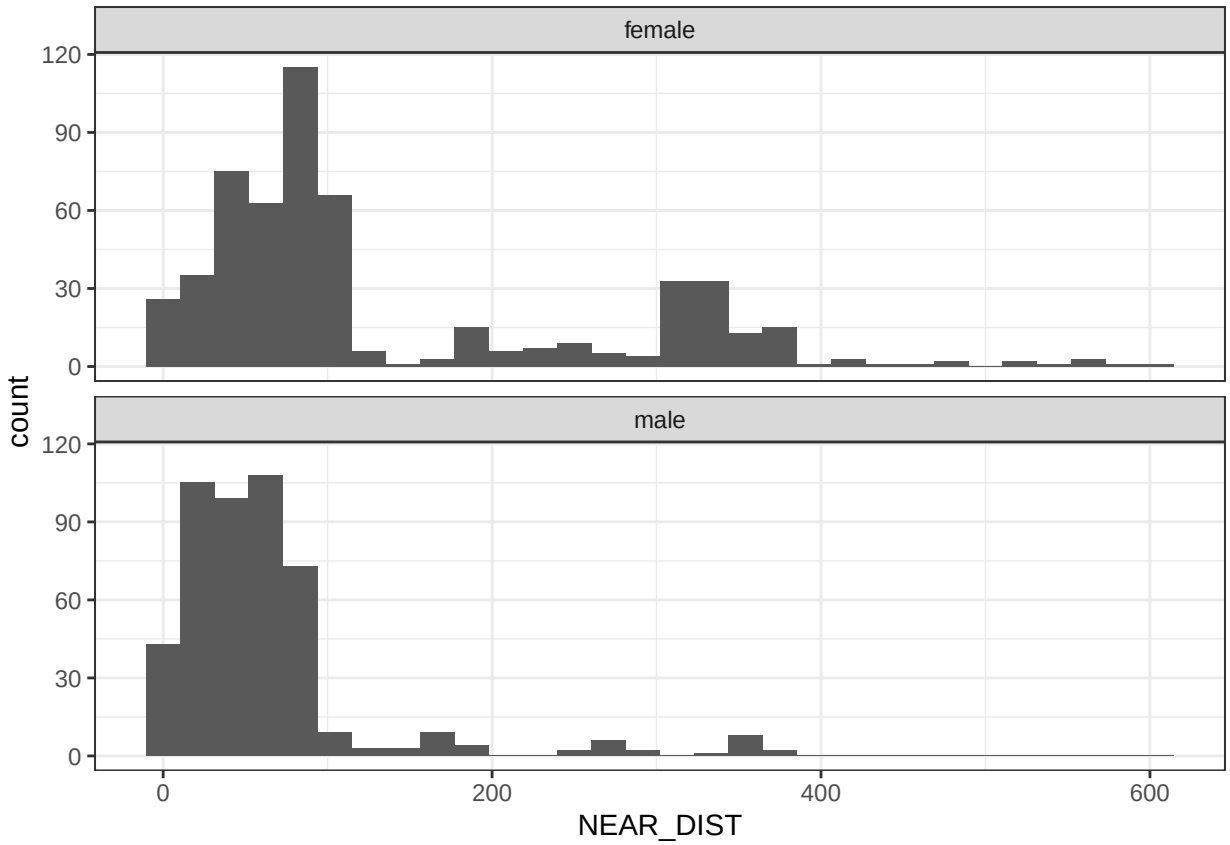
Create subset

```
df <- dplyr::select(turtles, Turtle_sex, NEAR_DIST)
```

Plot the relationship

```
ggplot(df, aes(NEAR_DIST))+
  geom_histogram() +
  facet_wrap(~Turtle_sex, ncol = 1) +
  theme_bw()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Summarise the data

```
turtlemeans <- summarise(
  group_by(df, Turtle_sex),
  meanNearDist = mean(NEAR_DIST))
```

Run ttest

```
turtle_ttest <- t.test(NEAR_DIST ~ Turtle_sex, data = df)
turtle_ttest
```

```
##
##  Welch Two Sample t-test
##
## data:  NEAR_DIST by Turtle_sex
## t = 12.246, df = 837.79, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group female and group male is not equal to
## 95 percent confidence interval:
##  63.36131 87.54920
## sample estimates:
## mean in group female   mean in group male
##            138.44754             62.99229
```

There is a significant difference between the distance from the stream occupied by males vs females (n=6)(p=0.00)

Biological Summary

I found that there was not a significant difference between WHAT - between WHAT measure compared between males and females? the male and female wood turtles in this sample (n=6)(p=0.4756). This was different than my original hypothesis and therefore, I cannot reject the null hypothesis.

There was a significant difference in home range point variability for all individuals (n=6)(p=0.00). Instead of saying "there was a significant difference" which tells me some information, what about "Males had significantly larger home ranges than females" which tells me more.

Finally, there was a significant difference between the distance from the stream occupied by female turtles (mean = 138m) than male turtles (mean = 63m)(p=0.00). I can reject the null hypothesis. Again, tell me the difference. Males were found signfiicantly closer to streams than females...

Challenges

I needed to learn how to manipulate numeric values into usable coordinates for spatial data analyses. One challenge in this was learning how to project coordinates into UTM so that instead of having decimal degrees, I would be working with meters. This was important for calculating the areas of the MCPs and the distance the turtles were from the stream.

I think moving forward I would find a better test and angle to look at the home range variability question. I also need to troubleshoot and figure out why my code stopped providing 6 spatial polygons for the mcp (and therefore affected the rest of my code).

Ideally we'll fix this before the final version.

Learning about spatial data analysis has been a lot of fun because I am able to contrast the process with my GIS class and I am excited to implement these skills moving forwards in my studies and my career.

this has been a great project!