

589Project

Nowshaba Durrani, Ricky Heinrich, Viji Rajagopalan

2023-04-09

Libraries

```
#install.packages("mapcan")
#install.packages("bcmaps")
#install.packages("rgbif")

library(rgbif) #allows searching and retrieving data from GBIF
library(ggplot2) #use ggplot2 to add layer for visualization
library(sp) #Standardized Support for Spatial Vector Data
library(sf)
library(spatstat)
library(maptools)
#library(raster)
#library(mapcan)
#library(bcmaps)
#library(tidyverse)
#library(rgdal)
```

Introduction:

```
#occ_count() # occurrence count for all the species in GBIF (Global Biodiversity Information Facility) -
redFox <- name_backbone(name="Vulpes vulpes")
redFoxList <- occ_data(taxonKey = redFox$speciesKey, hasCoordinate=TRUE, stateProvince='British Columbia')
mydata <- redFoxList$data
n_row <- nrow(redFoxList$data)
n_col <- ncol(redFoxList$data)
#n_row
#n_col
```

For our Data 589 project, we have selected Red Fox (Scientific Name - *Vulpes Vulpes*) to do the analysis. In the GBIF database they have approximately, 610,958+ georeferences records for this species around the world, however for this project we have selected to do the analysis of the occurrence of Red Fox in BC only. So with the above function we have fetched the information for British Columbia only in 127 columns and 242 number of entries.

```

load("BC_Covariates.Rda")

# Create a spatial points data frame from the longitude and latitude columns
coordinates <- mydata[,c("decimalLongitude", "decimalLatitude")]
dat.sp <- SpatialPointsDataFrame(c(mydata[,c('decimalLongitude','decimalLatitude')]), data = mydata)

# Set the current CRS
proj4string(dat.sp)<- CRS("+proj=longlat +datum=WGS84")

# Define the new CRS you want to transform to
new_crs <- CRS("+proj=aea +lat_0=45 +lon_0=-126 +lat_1=50 +lat_2=58.5 +x_0=1000000
+y_0=0 +datum=NAD83 +units=m +no_defs")

# Transform the data to the new CRS
data.sp_trans <- spTransform(dat.sp, new_crs)

#data_transformed
#data.sp_trans

#plot(data.sp_trans, main = "Locations in BC", cex = 0.8, col ="blue")

```

```

library(sf)
lapply(DATA, FUN = class)

```

```

## $Window
## [1] "SpatialPolygons"
## attr(),"package")
## [1] "sp"
##
## $Elevation
## [1] "im"
##
## $Forest
## [1] "im"
##
## $HFI
## [1] "im"
##
## $Dist_Water
## [1] "im"

```

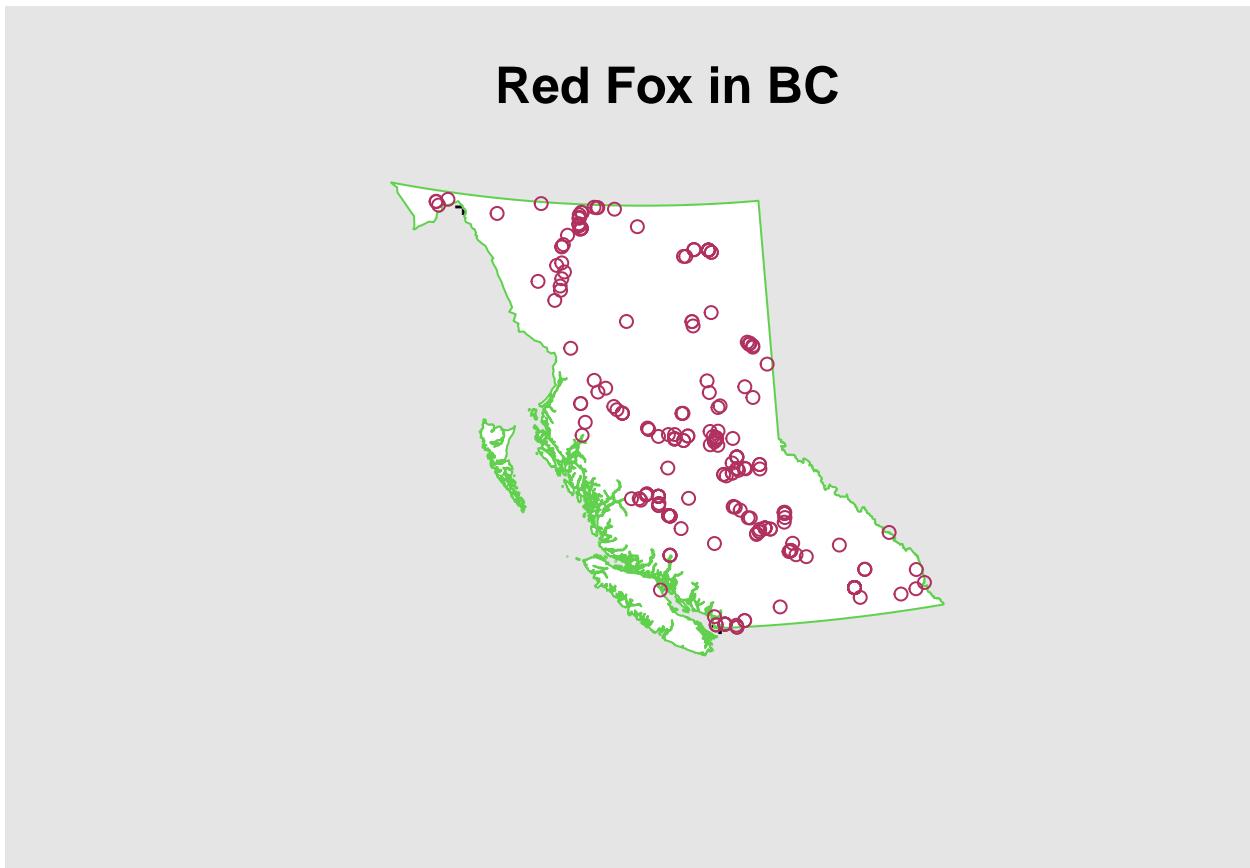
```

parks_ppp <- ppp(x = data.sp_trans@coords[,1], # X coordinates
                    y = data.sp_trans@coords[,2], # Y coordinates
                    window = as.owin( DATA[["Window"]]),# Observation window
                    )

col_pal <- c("maroon")
plot(parks_ppp,
      main = "Red Fox in BC",
      cex = 0.9,
      col ="white",
      border = 3,

```

```
cols = col_pal,  
par(bg = "grey90", cex.main = 1.6))
```



Here we have plotted all the occurrences of Red Fox in the BC region and we can see that the species are scattered in the region specially in the upper and middle part of the province. Now we will be exploring what is contributing to the occurrences of the species in the specific places based on various factors like elevation, close to water bodies, forests, human habitats, etc.

Methods:

Briefly describe the data and what variables are included. Provide a detailed description of the analytical workflow that was applied to the data, citing any relevant literature and statistical packages employed. There should be enough information that anyone can reproduce the workflow if they had access to the data. Length: As long as necessary.

First Moment Analysis

```
#summary(parks_ppp)  
intensity(parks_ppp)
```

```
## [1] 2.509854e-10
```

Per the summary, Average intensity 5.063089e-10 points per square unit which is 0.0000000005063089 per square unit and this does not explain the observance of Vulpes Vulpes in a meaningful way.

Quadratcount: 5 by 5 and 10 by 10 - Both convey different view points on the intensity of the observance. According to plot 1, most of the Vulpes Vulpes are spotted in the South West areas around Vancouver.

The 10X10 figure shows the intensity is high in the coastal areas with higher density in the South West region.

```
#Split into a 5 by 5 quadrat and count points
Q <- quadratcount(parks_ppp,
                    nx = 5,
                    ny = 5)

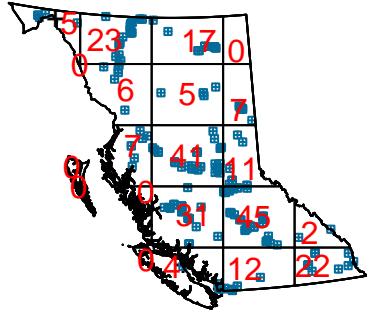
#Plot the output
par(mfrow=c(1,2))
plot(parks_ppp,
      pch = 12,
      cex = 0.5,
      cols = "#046C9A",
      main = "Vulpes Vulpes locations")

plot(Q, cex = 1, col = "red", add = T)

Q <- quadratcount(parks_ppp,
                    nx = 10,
                    ny = 10)

#Plot the output
par(mfrow=c(1,2))
```

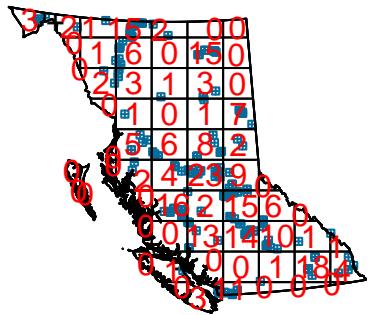
Vulpes Vulpes locations



```
plot(parks_ppp,
  pch = 12,
  cex = 0.5,
  cols = "#046C9A",
  main = "Beilschmiedia pendula locations")

plot(Q, cex = 1, col = "red", add = T)
```

Beilschmiedia pendula location:



Quadrat counting suggests varying intensity and to confirm that the variation is not due to chance alone, we conduct an objective test for spatial (in)homogeneity. We do a Chi-square test to validate if the deviations are significant.

```
#Quadrat test of homogeneity
quadrat.test(Q)
```

```
##
##  Chi-squared test of CSR using quadrat counts
##
## data:
## X2 = 352.69, df = 63, p-value < 2.2e-16
## alternative hypothesis: two.sided
##
## Quadrats: 64 tiles (irregular windows)
```

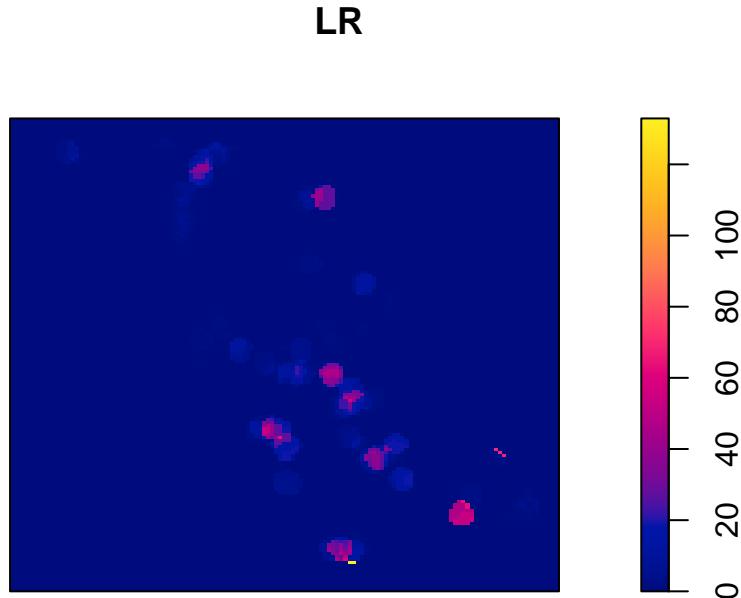
The null hypothesis of the test suggests homogeneity in the process and as the p-value is very small, the null hypothesis is rejected and its confirmed there is significant deviation from homogeneity.

Hot spot analysis: As the next step, we analyze for any hot spots in the south west coastal areas of BC.

```
# Estimate R
R <- bw.ppl(parks_ppp)

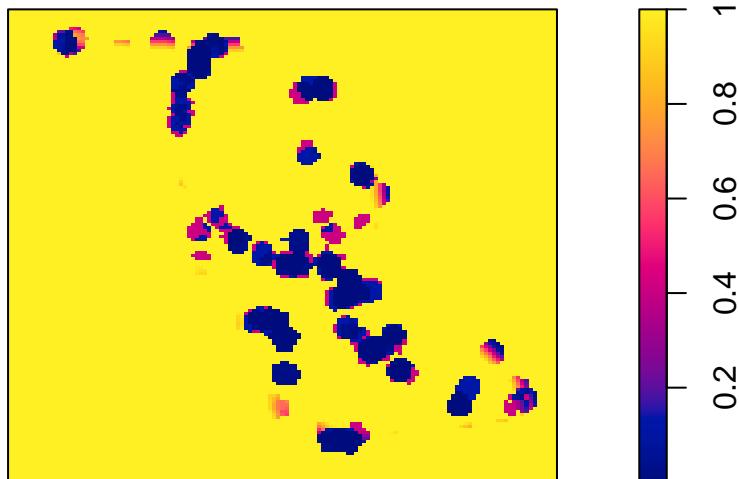
#Calculate test statistic
LR <- scanLRTS(parks_ppp, r = R)
```

```
#Plot the output  
plot(LR)
```



```
#Compute local p-values  
pvals <- eval.im(pchisq(LR,  
                         df = 1,  
                         lower.tail = FALSE))  
  
#Plot the output  
plot(pvals, main = "Local p-values")
```

Local p-values



Question: Do we need p -value intensity analysis? Also, is it possible to add the window for better observation window boundary (shape of BC)?

```
#add marks and relationship with one covariate to start with  
#parks_ppp <- ppp(x = data.sp_trans$decimalLatitude, # X coordinates  
                      y = data.sp_trans$decimalLongitude)  
  
#.....
```

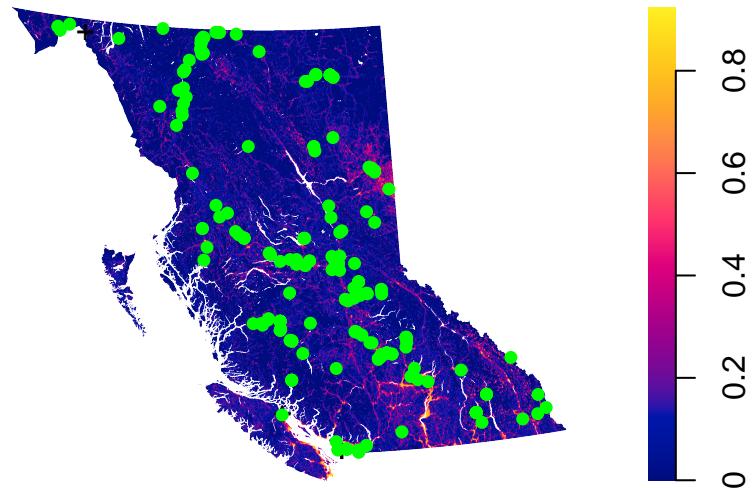
Covariates Analysis

Our data includes 4 covariates we can explore: the elevation, the forest cover, the human footprint inventory (HFI), and the distance to water. Given our research questions, we will start with investigating the HFI and the forest cover.

HFI

```
plot(DATA$HFI, box = F, par(cex.main = 2), main = "HFI")  
plot(parks_ppp, pch = 16, cex = 0.9, col = "green", use.marks = F, add = T)
```

HFI

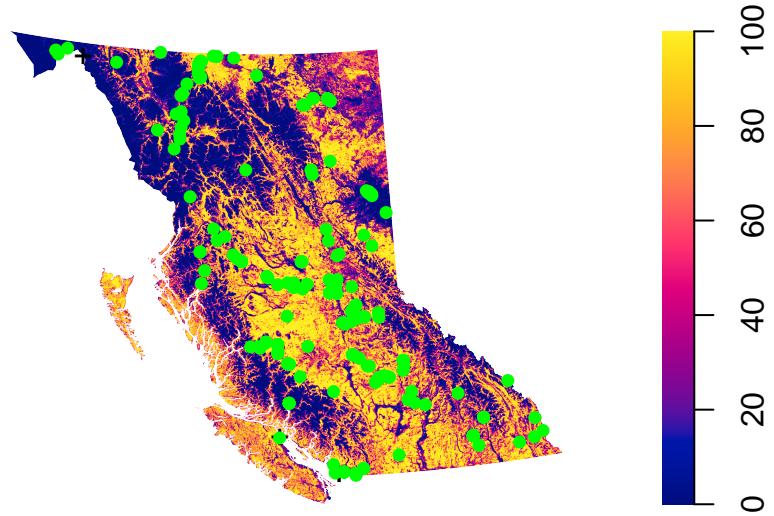


From this plot, it is hard to tell if there is a possible relationship between a red fox occurrence and the HFI.

Forest Cover

```
plot(DATA$Forest, box = F, par(cex.main = 2), main = "Forest")
plot(parks_ppp, pch = 16, cex = 0.9, col = "green", use.marks = F, add = T)
```

Forest

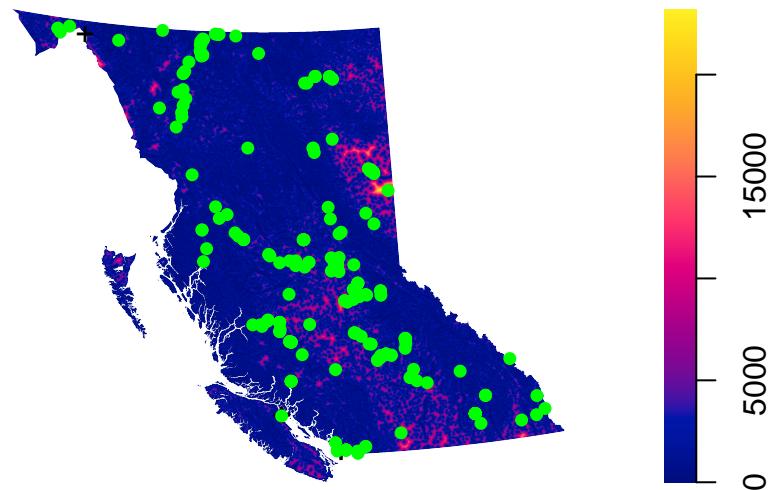


It is also very hard to see if there is a relationship, this time because there are a lot of high values for forest cover all over the province.

Distance to water

```
plot(DATA$Dist_Water, box = F, par(cex.main = 2), main = "Distance to water")
plot(parks_ppp, pch = 16, cex = 0.9, col = "green", use.marks = F, add = T)
```

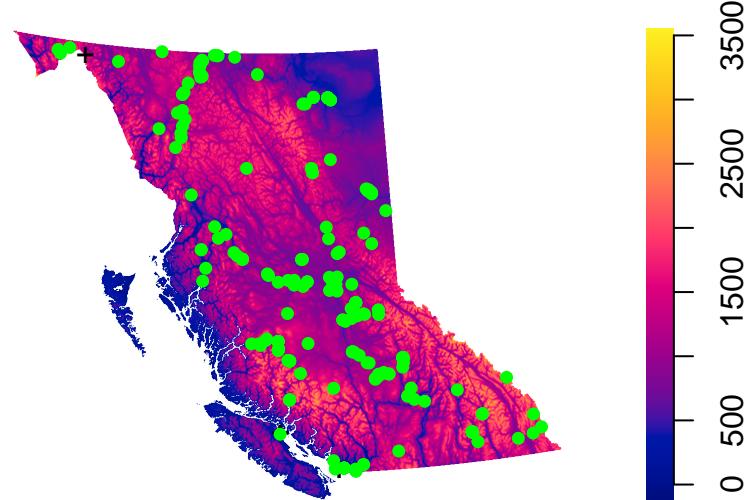
Distance to water



Elevation

```
plot(DATA$Elevation, box = F, par(cex.main = 2), main = "Elevation")
plot(parks_ppp, pch = 16, cex = 0.9, col = "green", use.marks = F, add = T)
```

Elevation



Conclusion: maybe need to segment the continuous values of covariates so that it is easier to see trend, like histogram in lab 1.

2nd Moment Analysis

Morisita's Index plot

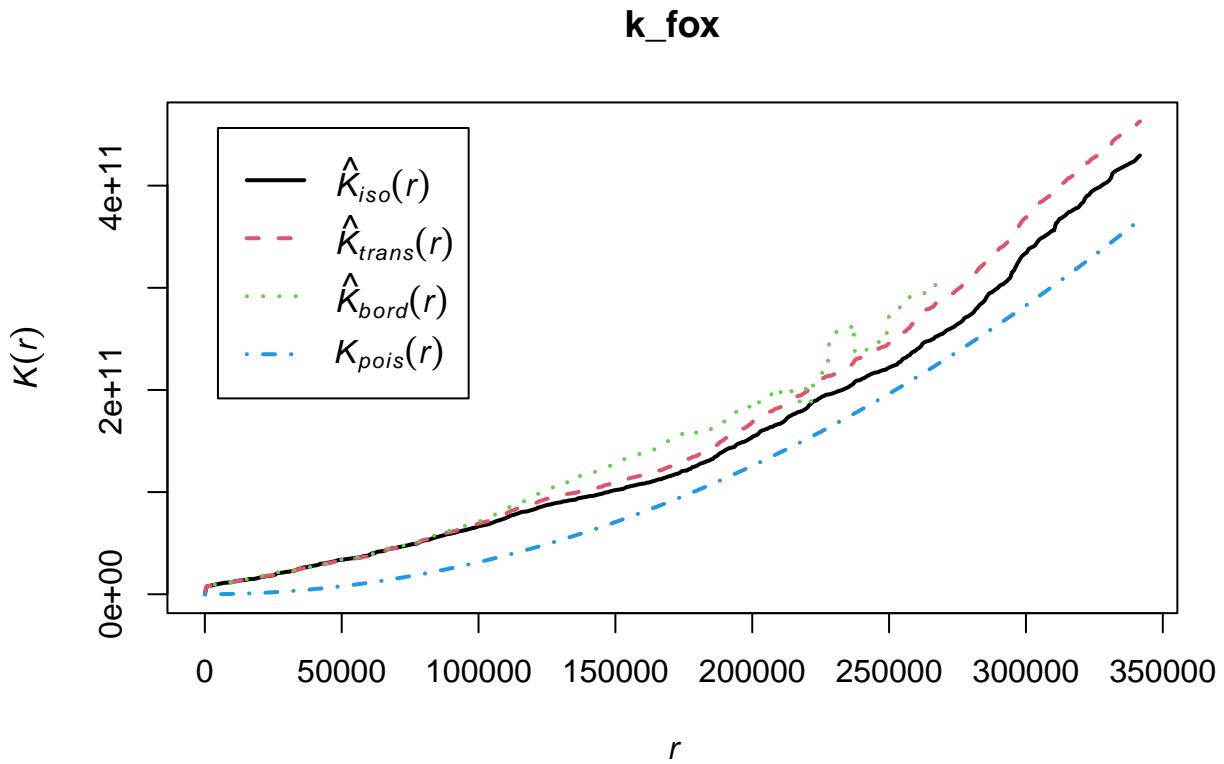
this produces an error, and we don't need to include it

```
miplot(parks_ppp,  
       main = "",  
       pch = 16,  
       col = "maroon")
```

Ripley's K function

Ripley's K-function provides information on whether there are significant deviations from independence between points.

```
k_fox <- Kest(parks_ppp)  
plot(k_fox, lwd=2)
```



The blue line is the theoretical line. (I don't know what the other lines mean)

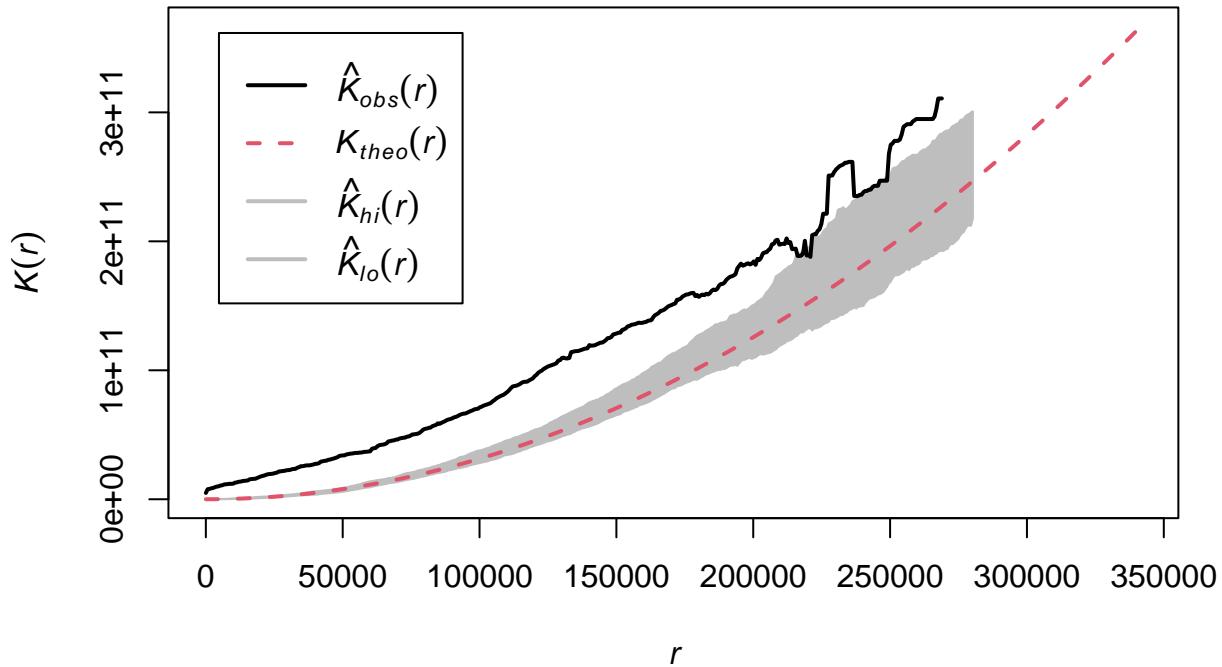
Adding confidence interval at a significance level of 0.05:

```
E_fox_homo <- envelope(parks_ppp,
    Kest,
    correction="border",
    rank = 1,
    nsim = 19, # aka alpha of 0.05
    fix.n = T)

## Generating 19 simulations of CSR with fixed number of points ...
## 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19.
##
## Done.

plot(E_fox_homo, lwd = 2)
```

E_fox_homo



(I don't know why it only plotted these lines and not the other ones) We see that the effect appears significant. It is suspicious that the confidence bands increase a lot, (what's the explanation for that?)

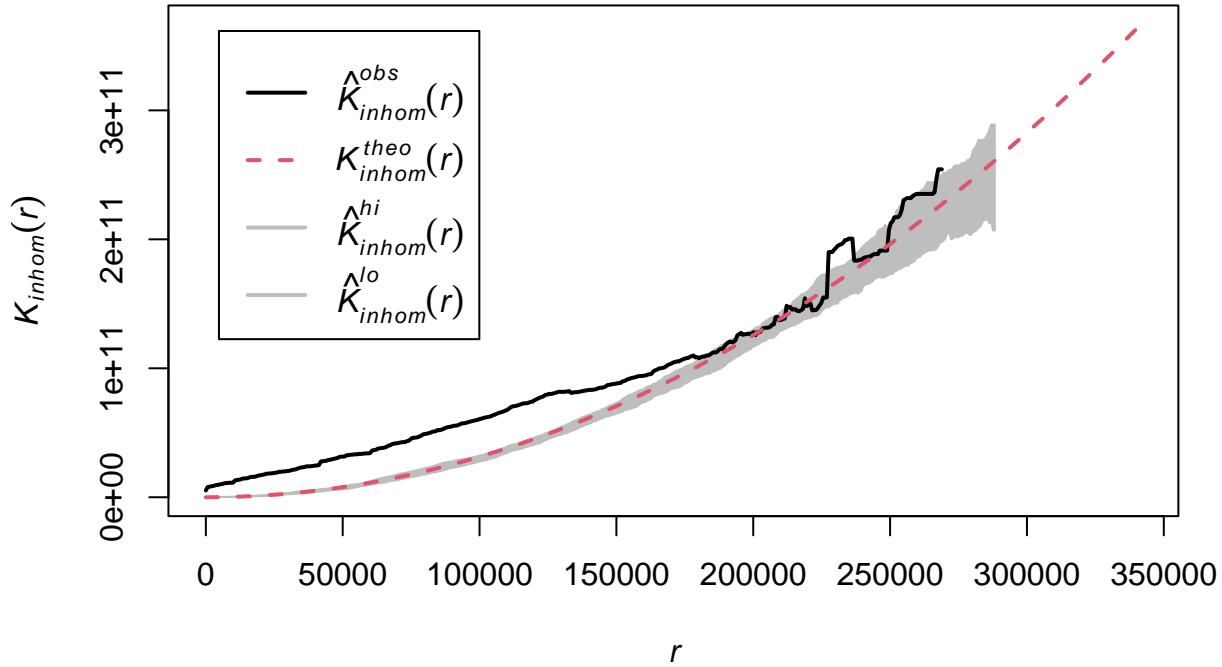
However, we know from first moment analysis that the intensity does not seem homogenous (right? Might need to double check). Trying inhomogenous:

```
E_fox <- envelope(parks_ppp,
                    Kinhom,
                    correction="border",
                    rank = 1,
                    nsim = 19, # aka alpha of 0.05
                    fix.n = T)

## Generating 19 simulations of CSR with fixed number of points ...
## 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19.
##
## Done.

plot(E_fox, lwd = 2)
```

E_fox



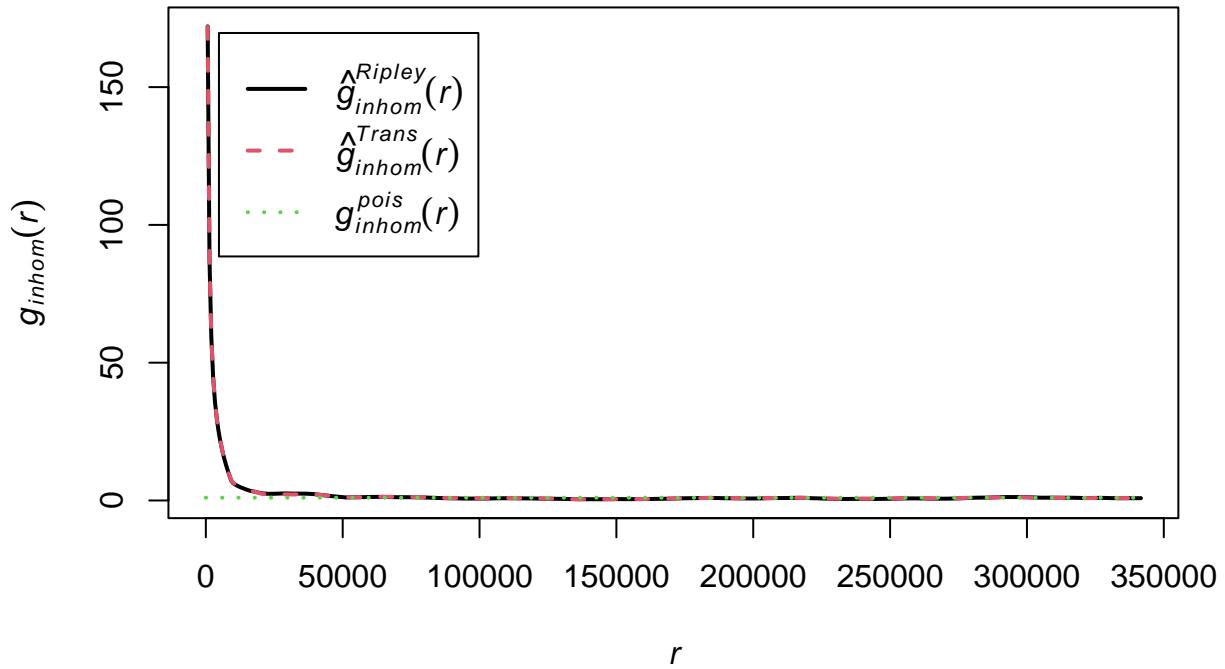
Using the Kinhom function ensures that we are not assuming the intensity is homogenous. It seems like from the smaller numbers, ie smaller distances between points, there is ‘evidence’ of clustering, whereas there are funny things going on as distances increase. The deviations are still meaningful in the ‘smaller’ distances, suggesting that the relationship between points may be due to effects between points rather than relationship with covariates.

Pair correlation function

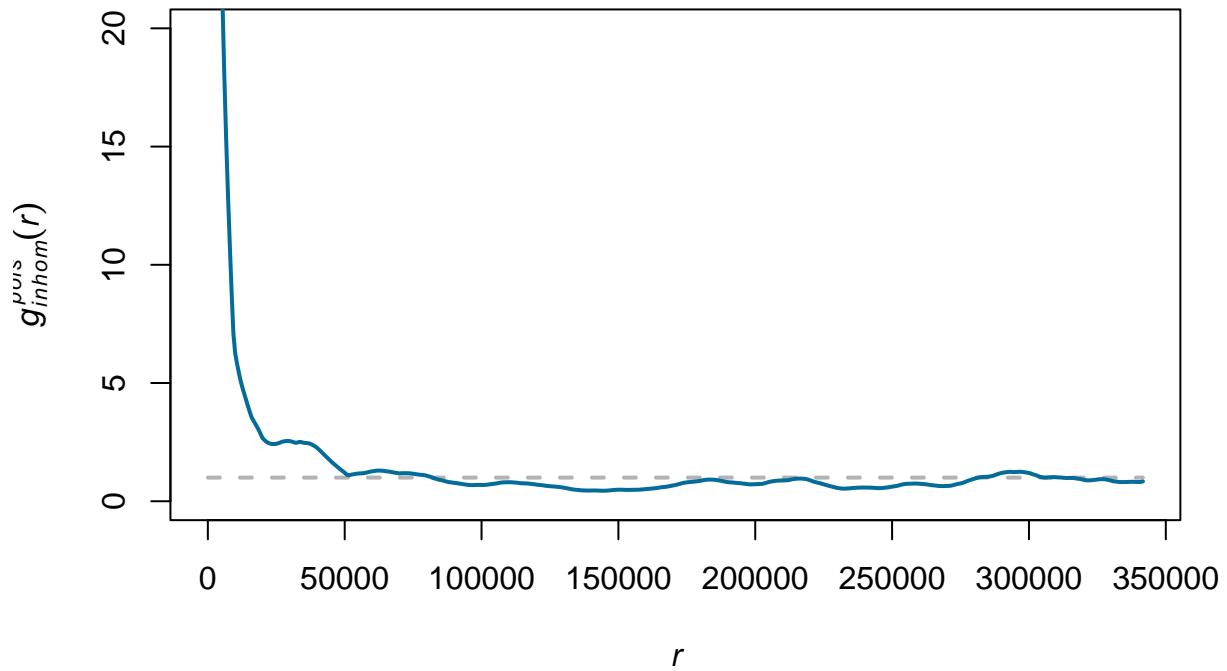
```
# Estimate the g function
pcf_fox <- pcfinhom(parks_ppp) # assumes inhomogeneity

# Default plot method
plot(pcf_fox, lwd = 2)
```

pcf_fox



```
# visualise the results
plot(pcf_fox,
      theo ~ r,
      ylim = c(0,20),
      main = "",
      col = "grey70",
      lwd = 2,
      lty = "dashed")
plot(pcf_fox,
      iso ~ r,
      col = c("#046C9A"),
      lwd = 2,
      add = T)
```



We observe that there seem to be evidence for clustering at smaller than 50 000, but after that it rides the $y = 1$ line slightly under.

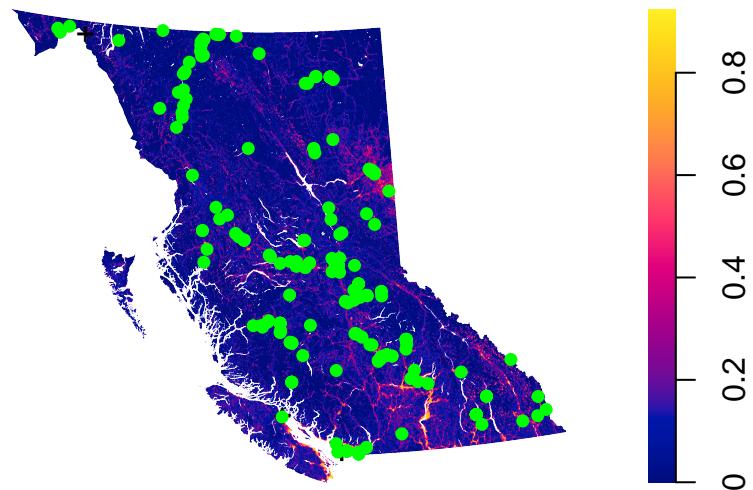
We need to validate clustering effects after modeling the data with covariates.

Relationship with Covariates

HFI

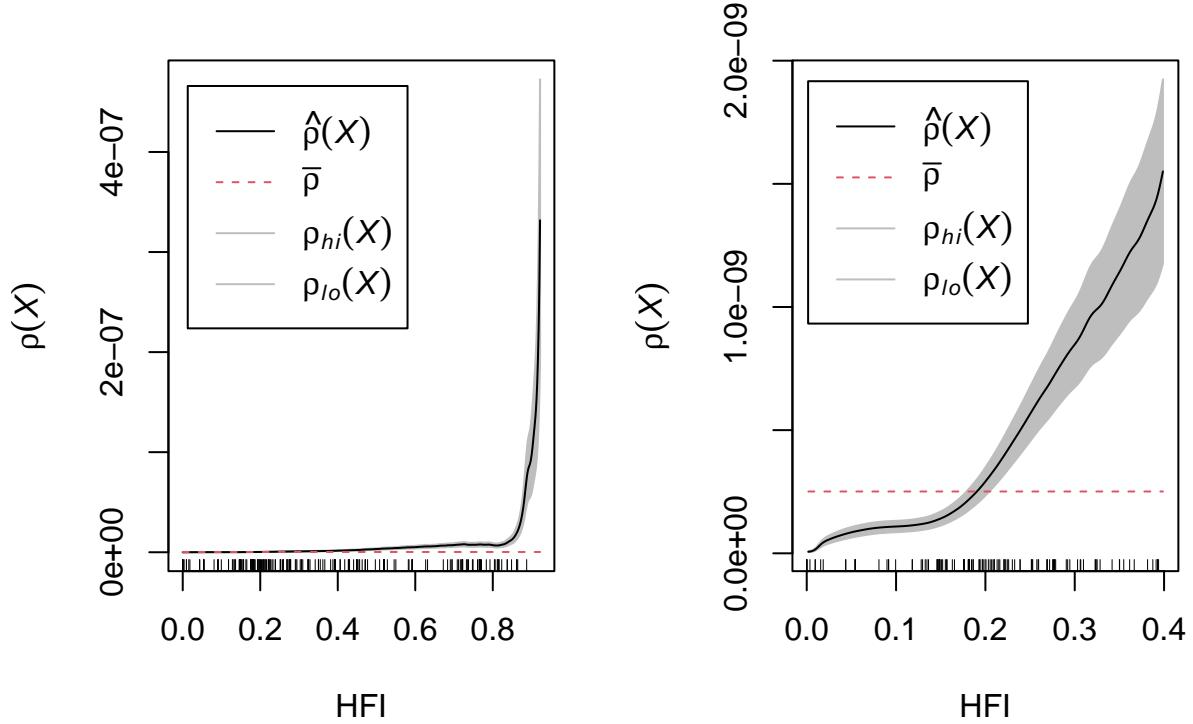
```
plot(DATA$HFI, box = F, par(cex.main = 2), main = "Forest")
plot(parks_ppp, pch = 16, cex = 0.9, col = "green", use.marks = F, add = T)
```

Forest



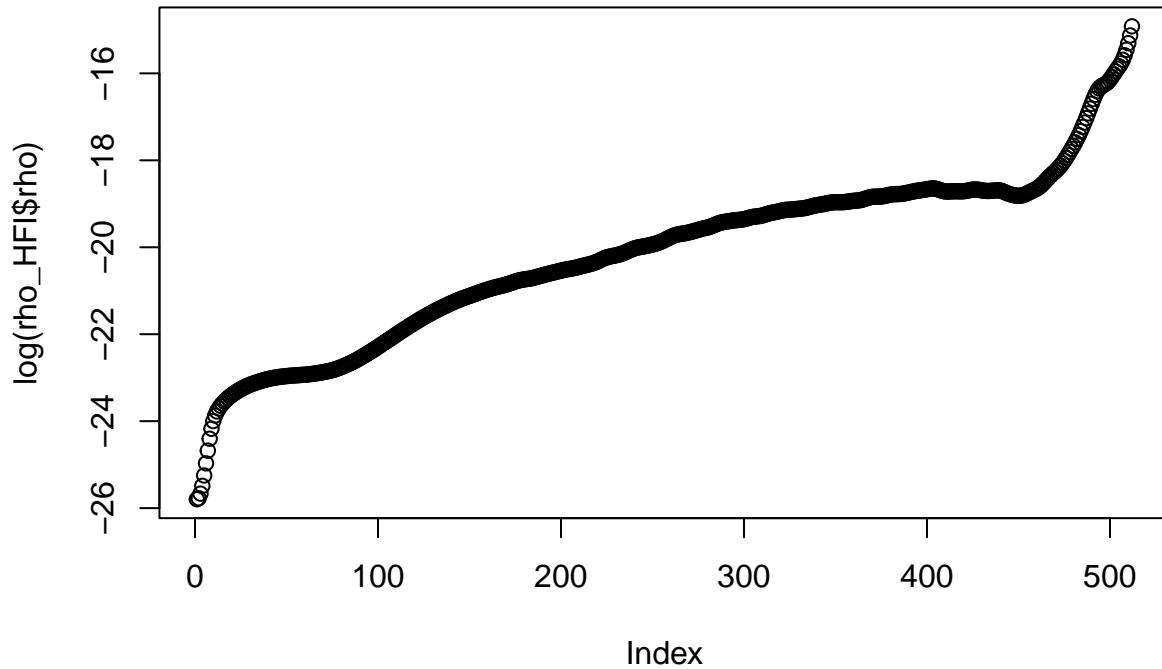
```
rho_HFI <- rhohat(parks_ppp, DATA$HFI)

par(mfrow = c(1,2))
plot(rho_HFI,
      main = "",
      xlab = "HFI")
plot(rho_HFI,
      main = "",
      xlab = "HFI", xlim = c(0,0.4))
```



In the first figure, we could be fooled into thinking that there is no relationship up to around $HFI = 0.4$, until which it seems like an exponential relationship. However, zooming in from $HFI \in [0, 1]$ to $HFI \in [0, 0.4]$, we see that the confidence bands don't intersect at all with the red line, which is the expected value given no relationship. This relationship appears non-linear and possibly exponential, where the greatest intensity of observed red foxes occurs at high HFIs. This relationship was expected, as our dataset is not exhaustive but rather is crowdsourced, and naturally foxes are more likely to be noticed by humans in spaces with higher HFIs.

```
plot(log(rho_HFI$rho))
```



If we plot the log of the rho, we get a line that could be reasonably interpreted as linear.

simple model

```

fitHFI <- ppm(parks_ppp ~ HFI, data=DATA)
fitHFI

## Nonstationary Poisson process
## Fitted to point pattern dataset 'park_ppp'
##
## Log intensity: ~HFI
##
## Fitted trend coefficients:
## (Intercept)      HFI
## -23.31421    5.98177
##
##             Estimate      S.E.    CI95.lo    CI95.hi Ztest      Zval
## (Intercept) -23.31421 0.1058336 -23.521645 -23.106785 *** -220.29132
## HFI         5.98177 0.2148471  5.560677  6.402862 ***   27.84199
## Problem:
## Values of the covariate 'HFI' were NA or undefined at 0.56% (12 out of 2137)
## of the quadrature points

```

```

fitHFIexp <- ppm(parks_ppp~HFI + exp(HFI), data=DATA)

## Warning: Values of the covariate 'HFI' were NA or undefined at 0.56% (12 out of
## 2137) of the quadrature points. Occurred while executing: ppm.ppp(Q = parks_ppp,
## trend = ~HFI + exp(HFI), data = list(new("SpatialPolygons",

fitHFIexp

## Nonstationary Poisson process
## Fitted to point pattern dataset 'parks_ppp'
##
## Log intensity: ~HFI + exp(HFI)
##
## Fitted trend coefficients:
## (Intercept)      HFI      exp(HFI)
## -13.71641    22.17067   -10.35893
##
##           Estimate     S.E.    CI95.lo    CI95.hi Ztest      Zval
## (Intercept) -13.71641 1.404309 -16.46881 -10.964015 *** -9.767372
## HFI         22.17067 2.382441  17.50117  26.840165 ***  9.305861
## exp(HFI)   -10.35893 1.522442 -13.34286  -7.374997 *** -6.804153
## Problem:
## Values of the covariate 'HFI' were NA or undefined at 0.56% (12 out of 2137)
## of the quadrature points

AIC(fitHFI); AIC(fitHFIexp)

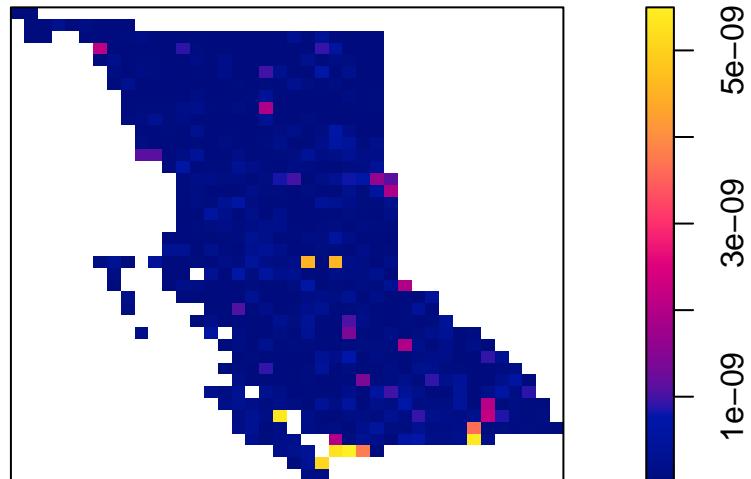
## [1] 10468.83

## [1] 10420.78

plot(fitHFIexp, se = F, superimpose = F)

```

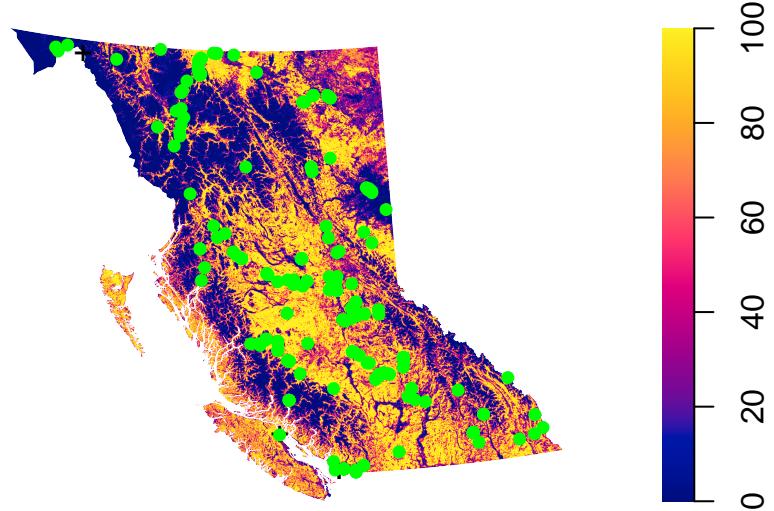
Fitted trend



Forest Cover

```
plot(DATA$Forest, box = F, par(cex.main = 2), main = "Forest")
plot(parks_ppp, pch = 16, cex = 0.9, col = "green", use.marks = F, add = T)
```

Forest



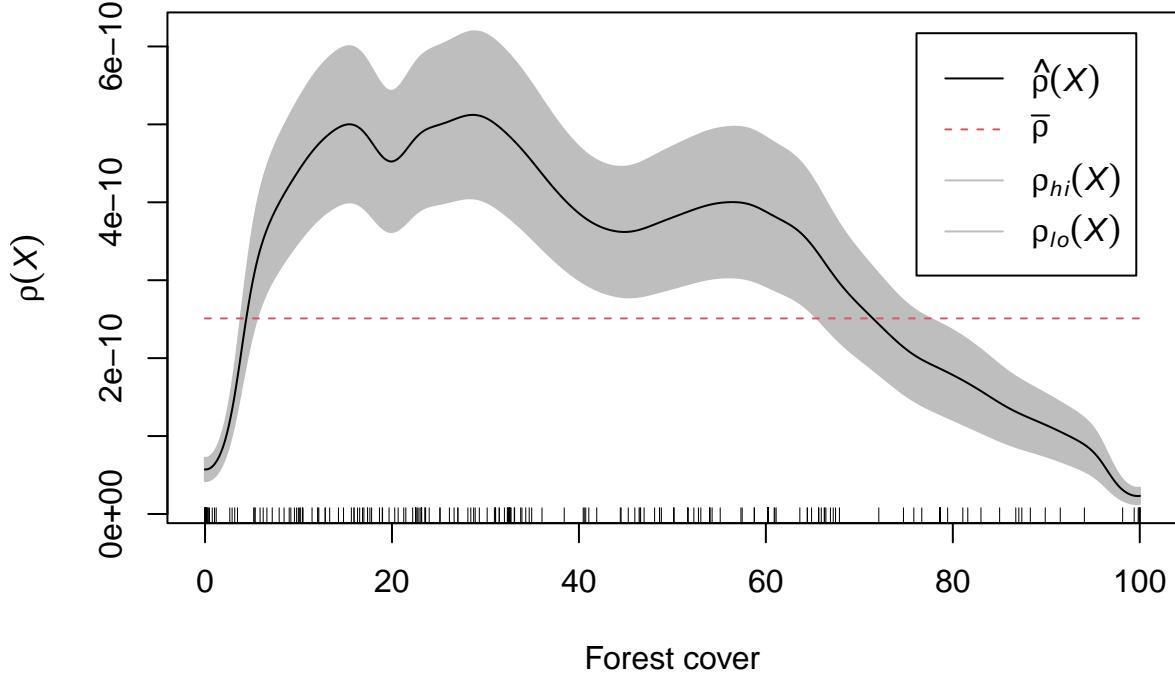
1. To add : Add other covariates together - side by side

```
#load the spatstat package
library(spatstat)

#Estimate Rho for Forest Cover
rho_elev <- rhohat(parks_ppp, DATA$Forest)

#Estimate Rho
#rho_grad <- rhohat(b ei, bei.extra$grad)

#par(mfrow = c(1,2))
plot(rho_elev,
      main = "",
      xlab = "Forest cover")
```



```
#plot(rho_grad,
#      main = "",
#      xlab = "Gradient (degrees)")
```

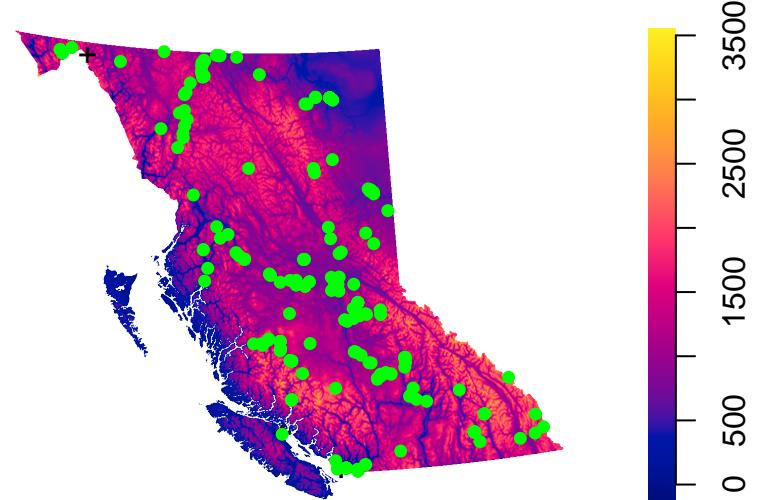
There seems to be non-linear relationship between forest cover and number of red foxs observed. The observance increase with increase in forest cover at intermediate coverage and then it decreases.

Next is to observe if there is any correlation between the covariates (collinearity in the covariates dataset). This is necessary to avoid any identifiability issues in modeling the data.

Elevation

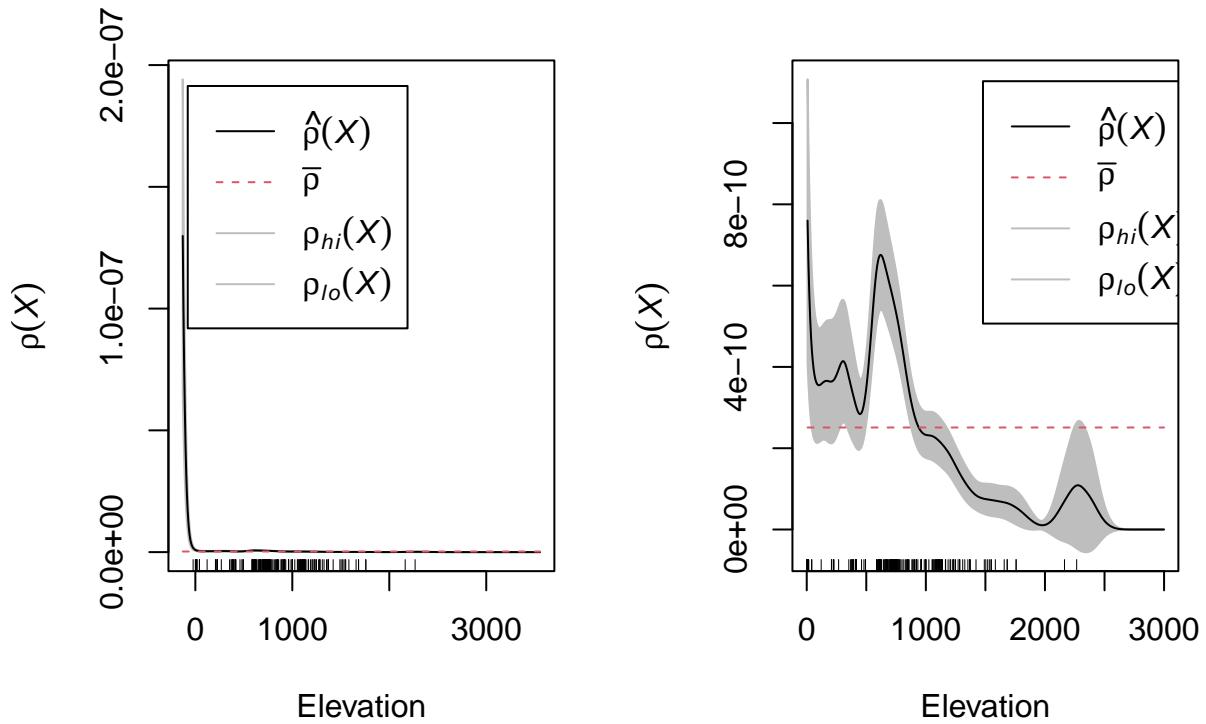
```
plot(DATA$Elevation, box = F, par(cex.main = 2), main = "Elevation")
plot(parks_ppp, pch = 16, cex = 0.9, col = "green", use.marks = F, add = T)
```

Elevation



```
#Estimate Rho for elevation
rho_elev <- rhohat(parks_ppp, DATA$Elevation)

#Plot Rho
par(mfrow = c(1,2))
plot(rho_elev,
      main = "",
      xlab = "Elevation")
plot(rho_elev,
      main = "",
      xlab = "Elevation", xlim = c(0,3000))
```

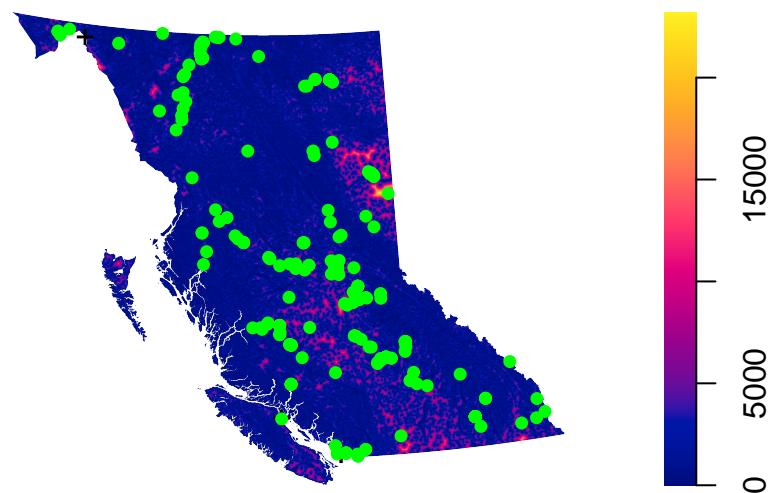


In the first figure, there seems to be no relationship but after taking a closer look we see that there is non-linear relationship. The relationship appears to be non-linear as the graph is showing different results for different elevation and we cannot see any type of specific pattern from the same.

Distance to Water

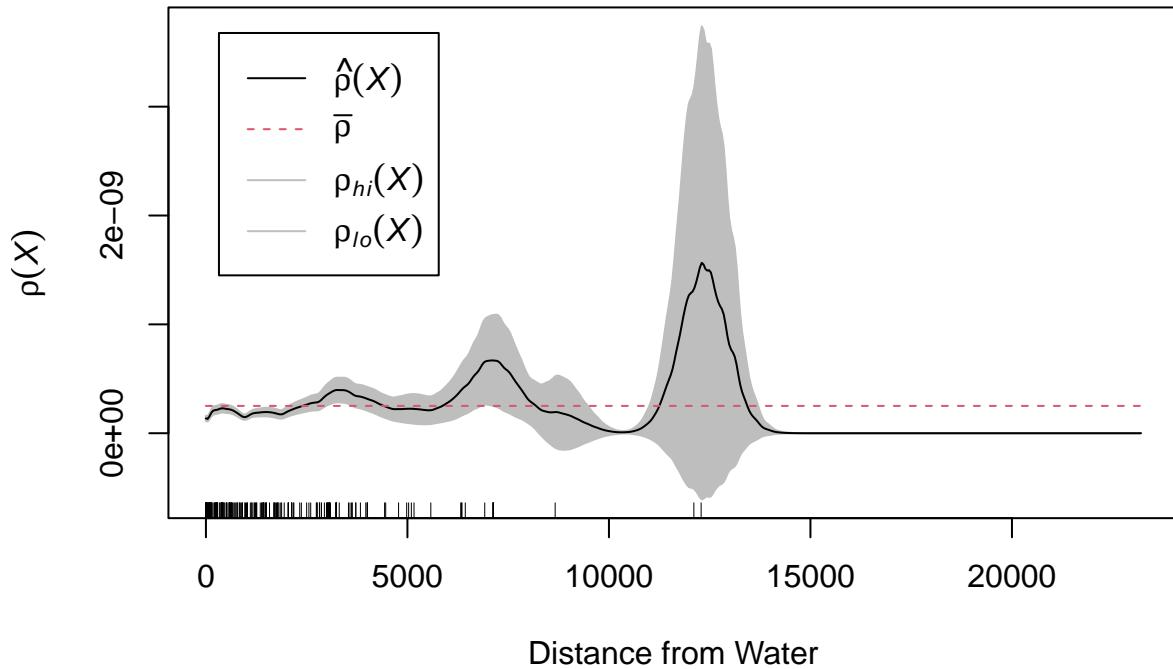
```
plot(DATA$Dist_Water, box = F, par(cex.main = 2), main = "Distance from Water")
plot(parks_ppp, pch = 16, cex = 0.9, col = "green", use.marks = F, add = T)
```

Distance from Water



```
#Estimate Rho for water distance
rho_water <- rhohat(parks_ppp, DATA$Dist_Water)

#Plot Rho
plot(rho_water,
      main = "",
      xlab = "Distance from Water")
```



In the above figure, we see there is non linear relationship.

2. Fix issues with collinearity

```
#Check for collinearity
x <- DATA$HFI
y <- DATA$Forest
#Y<- scaletointerval.im
#im.apply(Y, cor.im, na.rm=TRUE, fun.handles.na=TRUE, check=TRUE)

#FUNC <- function(x,y){
#  cor.im(x,y)
#}

#class(y)

#plot(DATA$HFI$v ~ DATA$Forest$v,
#      xlab = "HFI",
#      ylab = "Forest",
#      pch = 16,
#      cex = 0.3,
#      col = "#046C9A")
```

3. Finalize and summarize about other two variables - elevation and water.

Model Fitting

Try different models - linear and quadratic, do necessary anova and other tests for model selection and validation

Model with Forest

```
#Fit the PPP model
fit <- ppm(parks_ppp ~ Forest + I(Forest^2), data=DATA)

fit

## Nonstationary Poisson process
## Fitted to point pattern dataset 'parks_ppp'
##
## Log intensity: ~Forest + I(Forest^2)
##
## Fitted trend coefficients:
##   (Intercept)      Forest    I(Forest^2)
## -2.225373e+01  3.996522e-02 -5.280288e-04
##
##           Estimate       S.E.     CI95.lo     CI95.hi Ztest
## (Intercept) -2.225373e+01 1.330469e-01 -2.251450e+01 -2.199297e+01 *** 
## Forest      3.996522e-02 7.091557e-03  2.606603e-02  5.386442e-02 *** 
## I(Forest^2) -5.280288e-04 7.699618e-05 -6.789385e-04 -3.771191e-04 *** 
##
##           Zval
## (Intercept) -167.262353
## Forest       5.635606
## I(Forest^2)  -6.857857
```

All variables look significant.

```
#Plot the model predictions
plot(fit,
      se = FALSE,
      superimpose = FALSE)

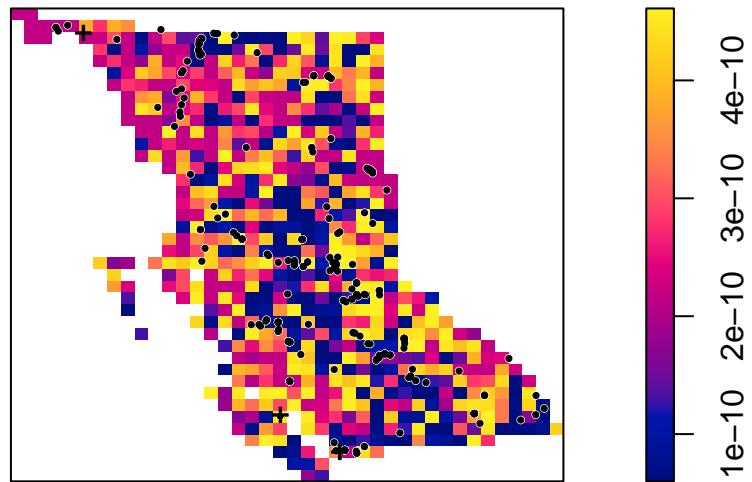
#Overlay the red fox locations
plot(parks_ppp,
      pch = 16,
      cex = 0.6,
      cols = "white",
      add = TRUE)

## Warning in plot.ppp(parks_ppp, pch = 16, cex = 0.6, cols = "white", add = TRUE):
## 4 illegal points also plotted
```

```
plot(parks_ppp,
      pch = 16,
      cex = 0.5,
      cols = "black",
      add = TRUE)

## Warning in plot.ppp(parks_ppp, pch = 16, cex = 0.5, cols = "black", add = TRUE):
## 4 illegal points also plotted
```

Fitted trend



Does not look like a good fit for all points and also not easy to interpret.

Analyze - linear+quad, AIC

Validate if we need GAM - 5. linear vs gam - table: evaluate, AIC, visualize

Results: Length: Describe your statistical findings. Tables and figures should be used throughout. Length: As long as necessary.

Findings from EDA

model: Tablulate observations -4. table to summarize models - linear+quad, AIC

model: GAM - Table if necessary

Discussion: Provide a brief summary of your findings. Length: ca. 1 page.

References: Include references to all necessary literature.

1. Data: GBIF.org (09 April 2023) GBIF Occurrence Download <https://doi.org/10.15468/dl.p6tsaa>
2. Research topics: <https://cwf-fcf.org/en/resources/encyclopedias/fauna/mammals/red-fox.html>