DATA 589 Project

Andrew Sarracini, Eden Chan, Nayeli Montiel

28 April, 2024 19:09:13

Data Loading

```r
# Libraries
library(sf)
```

```
## Linking to GEOS 3.11.0, GDAL 3.5.3, PROJ 9.1.0; sf_use_s2() is TRUE
```

```r
library(sp)
library(spatstat)
```

```
## Loading required package: spatstat.data

## Loading required package: spatstat.geom

## spatstat.geom 3.2-9

## Loading required package: spatstat.random

## spatstat.random 3.2-3

## Loading required package: spatstat.explore

## Loading required package: nlme

## spatstat.explore 3.2-7

## Loading required package: spatstat.model

## Loading required package: rpart

## spatstat.model 3.2-11

## Loading required package: spatstat.linnet

## spatstat.linnet 3.1-5

##
## spatstat 3.0-8
## For an introduction to spatstat, type 'beginner'
```

```r
# Load the spatial covariates
load("BC_Covariates.Rda")

# --- Species Data ---
# Set the file path to read occurrences from Anna's Hummingbirds
file_path <- "0158178-240321170329656/occurrence.txt"

# Read the entire dataset into a data frame
data <- read.delim(file_path, stringsAsFactors = FALSE)
```

```
## Warning in scan(file = file, what = what, sep = sep, quote = quote, dec = dec,
## : EOF within quoted string
```

```r
# Print the dimension of the data frame
dim(data)
```

```
## [1] 291431    223
```

Data pre-processing

```r
# Filtering the data by province
data_bc<- subset(data, stateProvince == "British Columbia")

#Filtering the data by year
data_bc<-data_bc[data_bc$year == 2024,]

# Selecting the latitude and longitude
clean_data <- data_bc[complete.cases(data_bc$decimalLatitude, data_bc$decimalLongitude), ]

# BC Albers projection
projection_string <- "+proj=aea +lat_0=45 +lon_0=-126 +lat_1=50 +lat_2=58.5 +x_0=1000000 +y_0=0 +datum=

# Create an sf object with your latitude and longitude coordinates
sf_data <- st_as_sf(clean_data, coords = c("decimalLongitude", "decimalLatitude"), crs = 4326)

# Transform the coordinates to BC Albers projection
sf_data_bc_albers <- st_transform(sf_data, projection_string)

# Extract the transformed coordinates
transformed_latitude <- st_coordinates(sf_data_bc_albers)[, 2]
transformed_longitude <- st_coordinates(sf_data_bc_albers)[, 1]

# Print the transformed coordinates
head(transformed_latitude)
```

```
## [1] 468171.3 462066.0 381652.0 484066.0 386189.2 385780.2
```

```r
head(transformed_longitude)
```

```
## [1] 1148759 1149840 1196961 1110793 1198486 1195120
```

Making a ppp object

```r
humm_ppp <- ppp(x = transformed_longitude,
                y = transformed_latitude,
                window = as.owin(st_as_sf(DATA$Window)))
```

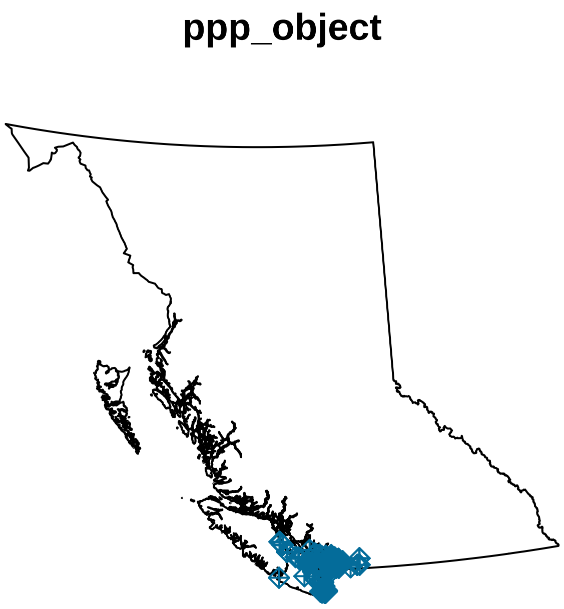
```
## Warning: 26 points were rejected as lying outside the specified window
```

```
## Warning: data contain duplicated points
```

```r
# Removing duplicates
duplicated_points <- duplicated(humm_ppp)

humm_ppp <- humm_ppp[!duplicated_points]
humm_ppp
```

```
## Planar point pattern: 260 points
## window: polygonal boundary
## enclosing rectangle: [273874.9, 1870573.4] x [369042.8, 1735666.4] units
```

```r
# Visualising ppp object
plot(humm_ppp,
     pch = 9,
     cols = "#046C9A",
     main = "ppp_object")
```



Quadrat count and homogeneity test

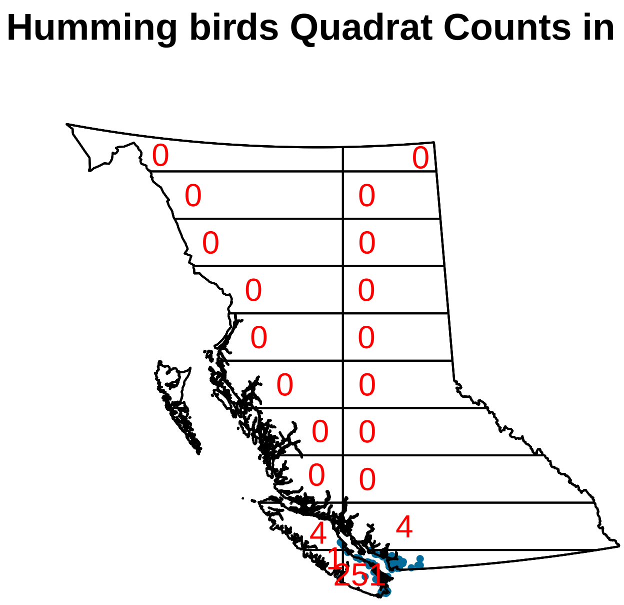
```r
#Split into a 2 by 10 quadrat and count points
Q <- quadratcount(humm_ppp,
                  nx=2,
                  ny=10)
sum(Q)
```

```
## [1] 260
```

```r
# Homogeneity test
quadrat.test(Q)
```

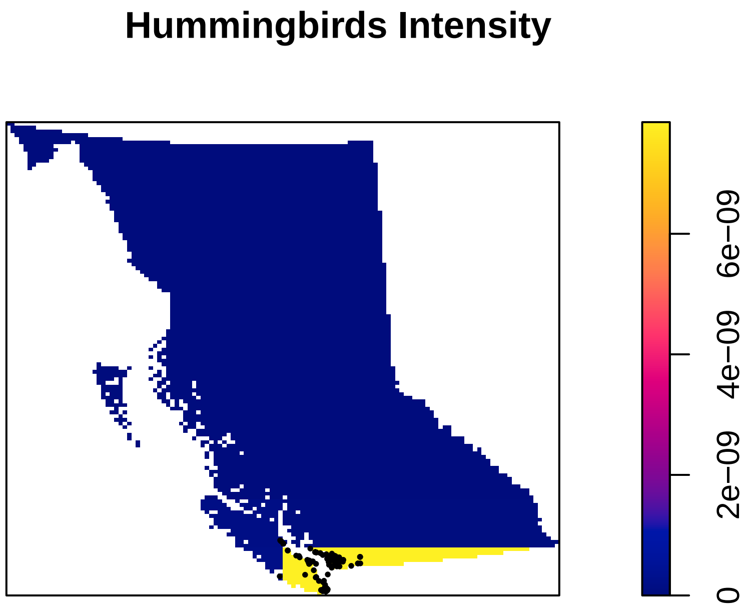
```
## Warning: Some expected counts are small; chi^2 approximation may be inaccurate

##
##  Chi-squared test of CSR using quadrat counts
##
## data:
## X2 = 6931.7, df = 19, p-value < 2.2e-16
## alternative hypothesis: two.sided
##
## Quadrats: 20 tiles (irregular windows)
```

```r
# Plot the output
plot(humm_ppp,
     pch = 16,
     cex = 0.5,
     cols = "#046C9A",
     main = "Humming birds Quadrat Counts in BC")

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



First Moment

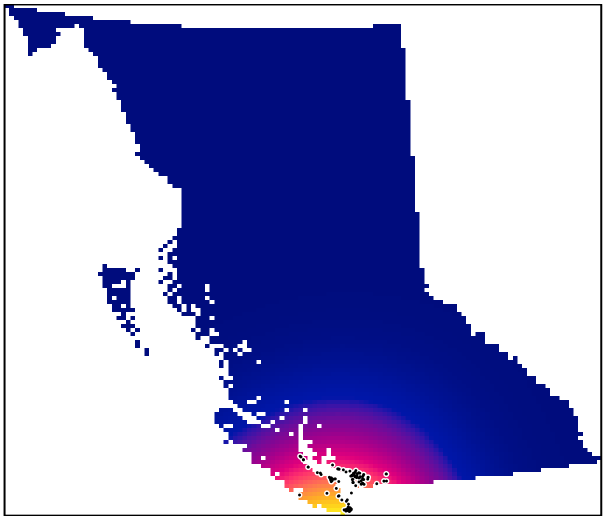
```
# Visualize intensity
plot(intensity(Q, image = T),
     main = "Hummingbirds Intensity")

# Add points
plot(humm_ppp,
     pch = 16,
     cex = 0.4,
     cols = "black",
     add = T)
```



Kernel-estimate

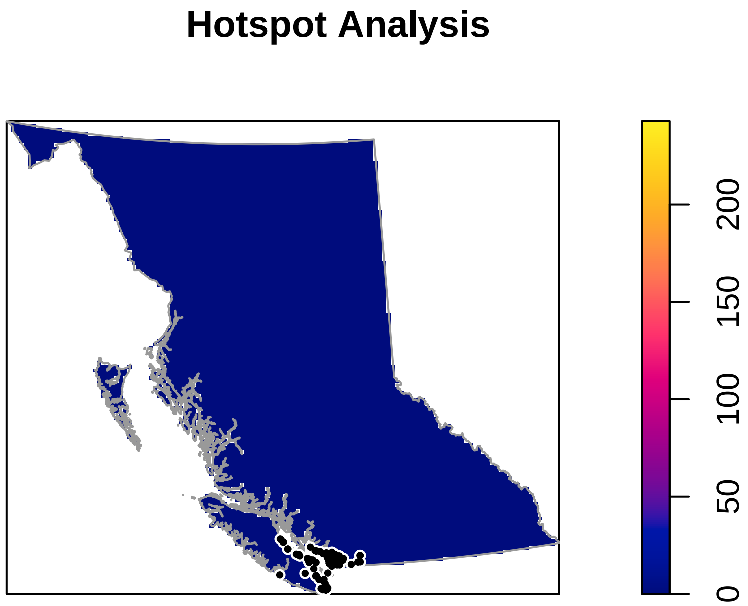
```
# Density estimation of lambda(u)
plot(density(humm_ppp),
     ribbon = F,
     main = "Kernel Esimate with Likelihood Cross Validation Bandwidth")

plot(humm_ppp,
     pch = 16,
     cex = 0.4,
     cols = "white",
     add = T)

plot(humm_ppp,
     pch = 16,
     cex = 0.2,
     cols = "black",
     add = T)
```

Kernel Esimate with Likelihood Cross Validation Bandwidth

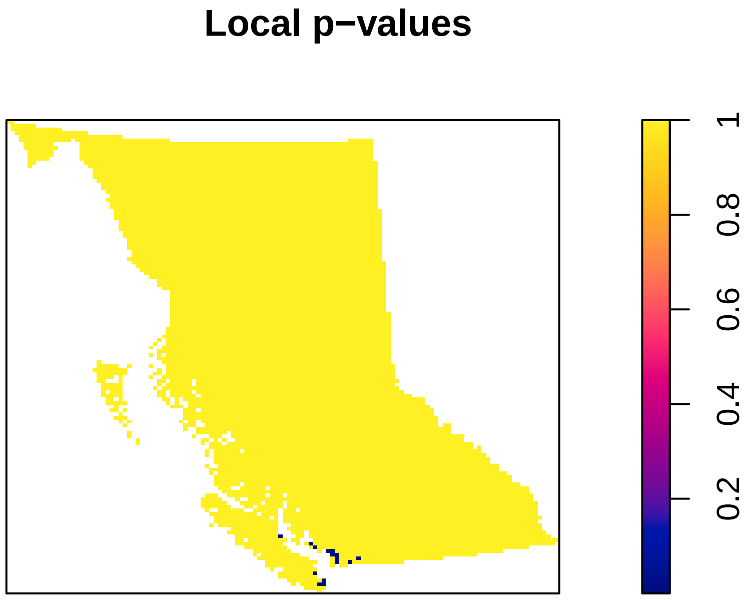


Hot-spot analisys

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

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

# Plot the output
plot(LR, main="Hotspot Analysis")

plot(DATA$Window,
     border = "gray60",
     add=T)

#Add the points
plot(humm_ppp,
     pch = 16,
     cex = 0.8,
     cols = "white",
     add = T)

plot(humm_ppp,
     pch = 16,
     cex = 0.5,
     cols = "black",
     add = T)
```



```
#Compute local p-values
pvals <- eval.im(pchisq(LR,
                        df = 1,
                        lower.tail = FALSE))


#Plot the output
plot(pvals, main = "Local p-values")
```



Second Moment

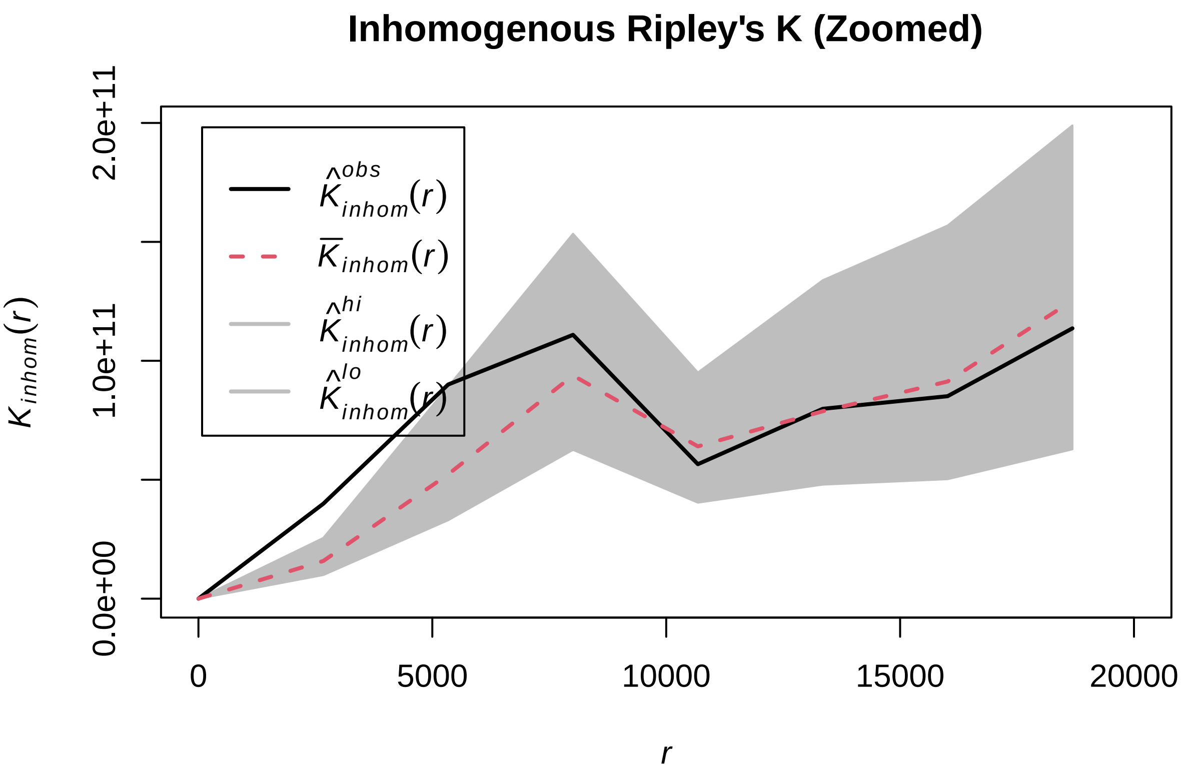
Inhomogenous K function

```r
#Estimate a strictly positive density
lambda_humm_pos <- density(humm_ppp,
                           sigma=bw.ppl,
                           positive=TRUE)

#Simulation envelope (with points drawn from the estimated intensity)
E_humm_inhom <- envelope(humm_ppp,
                         Kinhom,
                         simulate = expression(rpoispp(lambda_humm_pos)),
                         correction="border",
                         rank = 1,
                         nsim = 99,
                         fix.n = TRUE)
```

```
## Warning in envelope.ppp(humm_ppp, Kinhom, simulate =
## expression(rpoispp(lambda_humm_pos)), : fix.n and fix.marks were ignored,
## because 'simulate' was given
```

```
## Generating 99 simulations by evaluating expression  ...
## 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20,
## 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40,
## 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60,
## 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80,
## 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98,
## 99.
##
## Done.
```

```r
# Zoom-in Ripley's K
plot(E_humm_inhom,
     xlim = c(0,20000),
     main = "Inhomogenous Ripley's K (Zoomed)",
     lwd = 2)
```



When accounting for inhomogeneity, it's clear that there is correlation between the points in the 0-6000m range. This is fairly concrete proof that the points representing the hummingbirds are clustered together in this range.

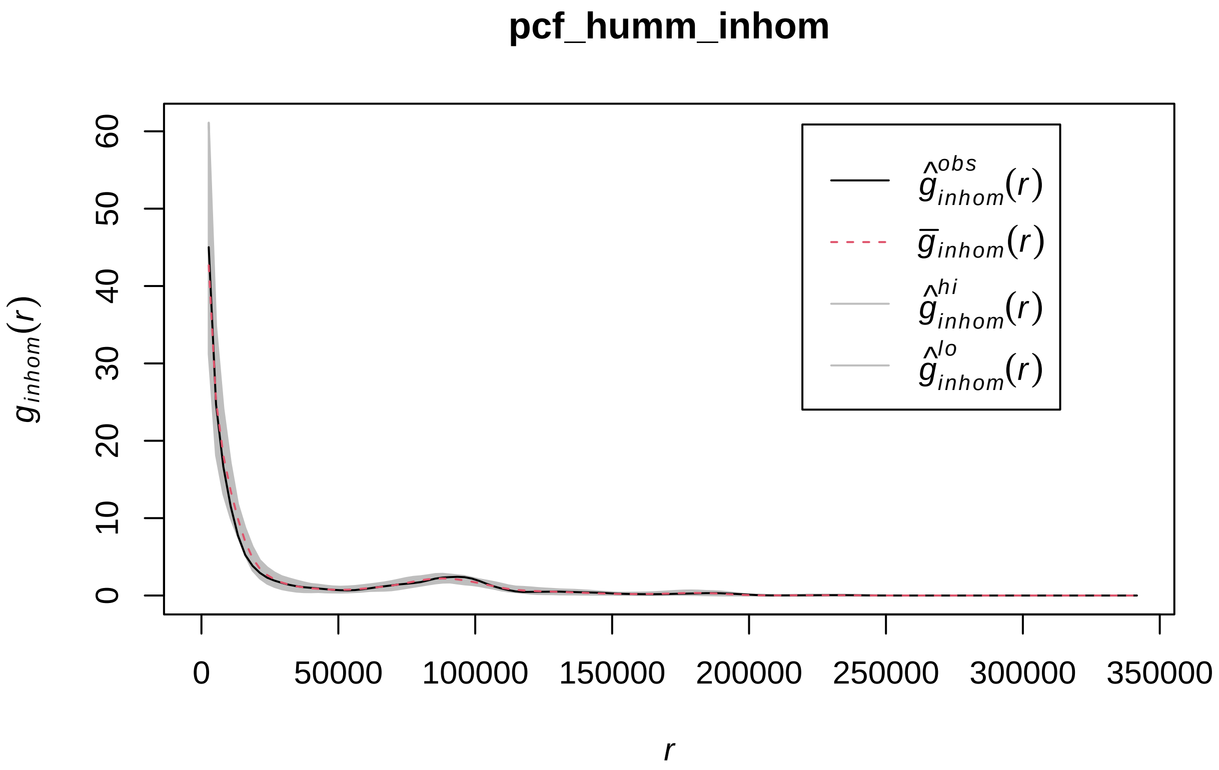
Inhomogenous pair correlation function

```
# Inhomogenous pcf
pcf_humm_inhom <- envelope(humm_ppp,
                          pcfinhom,
                          simulate = expression(rpoispp(lambda_humm_pos)),
                          rank = 1,
                          nsim = 99,
                          fix.n = TRUE)
```

```
## Warning in envelope.ppp(humm_ppp, pcfinhom, simulate =
## expression(rpoispp(lambda_humm_pos)), : fix.n and fix.marks were ignored,
## because 'simulate' was given
```

```
## Generating 99 simulations by evaluating expression  ...
## 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20,
## 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40,
## 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60,
## 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80,
## 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98,
## 99.
##
## Done.
```

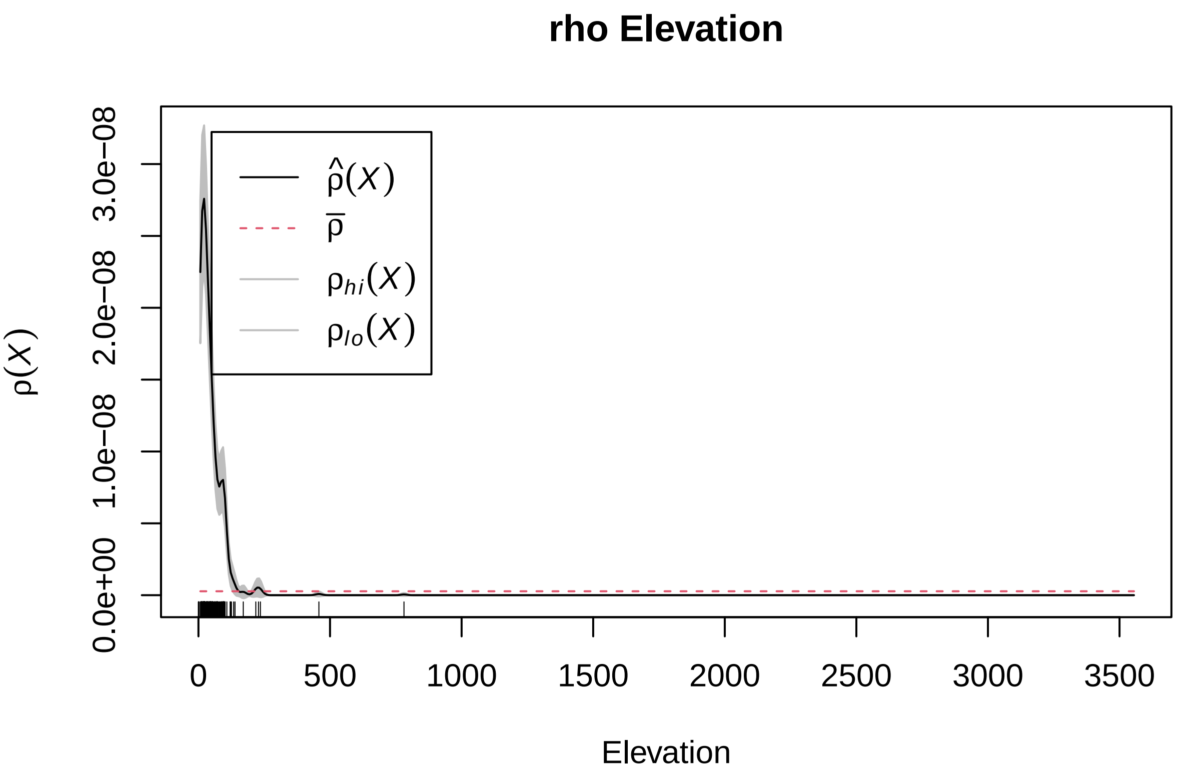
```r
plot(pcf_humm_inhom)
```



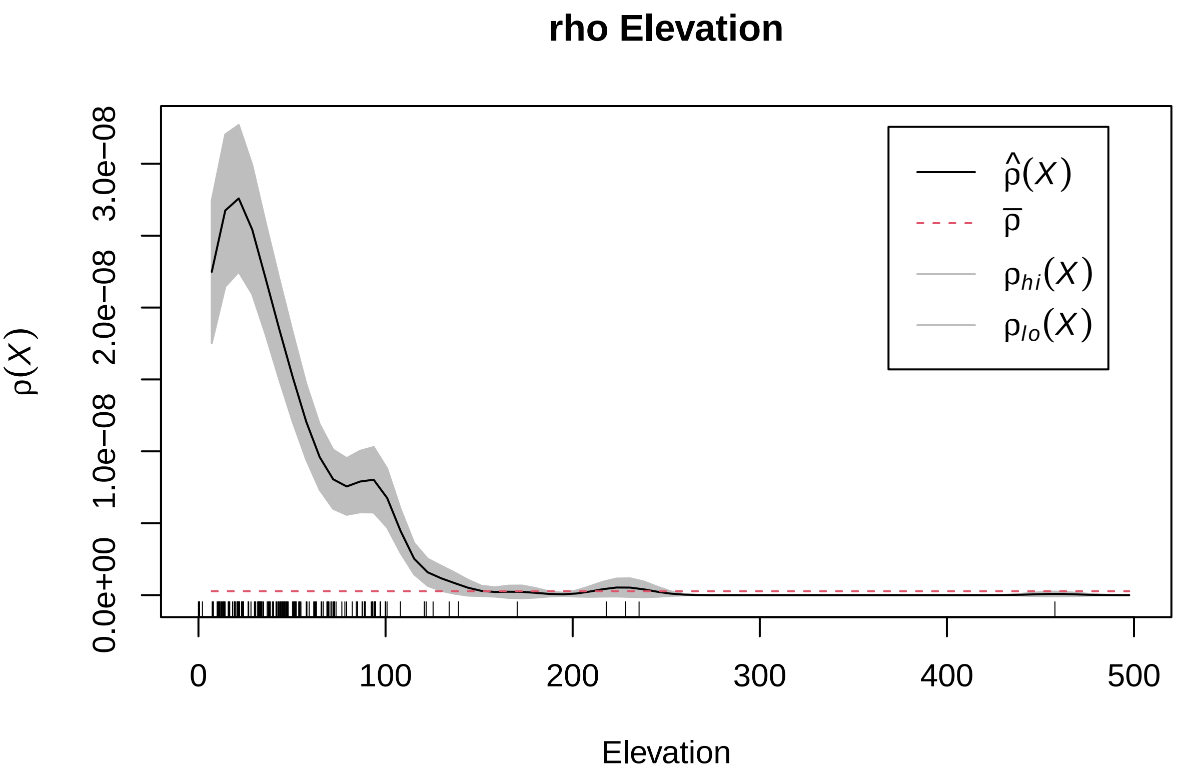
Model

Rho – Elevation

```r
#Estimate Rho
rho_elev <- rhohat(humm_ppp, DATA$Elevation)
plot(rho_elev, main="rho Elevation", xlab = "Elevation", xlim =c(0, max(DATA$Elevation)))
```

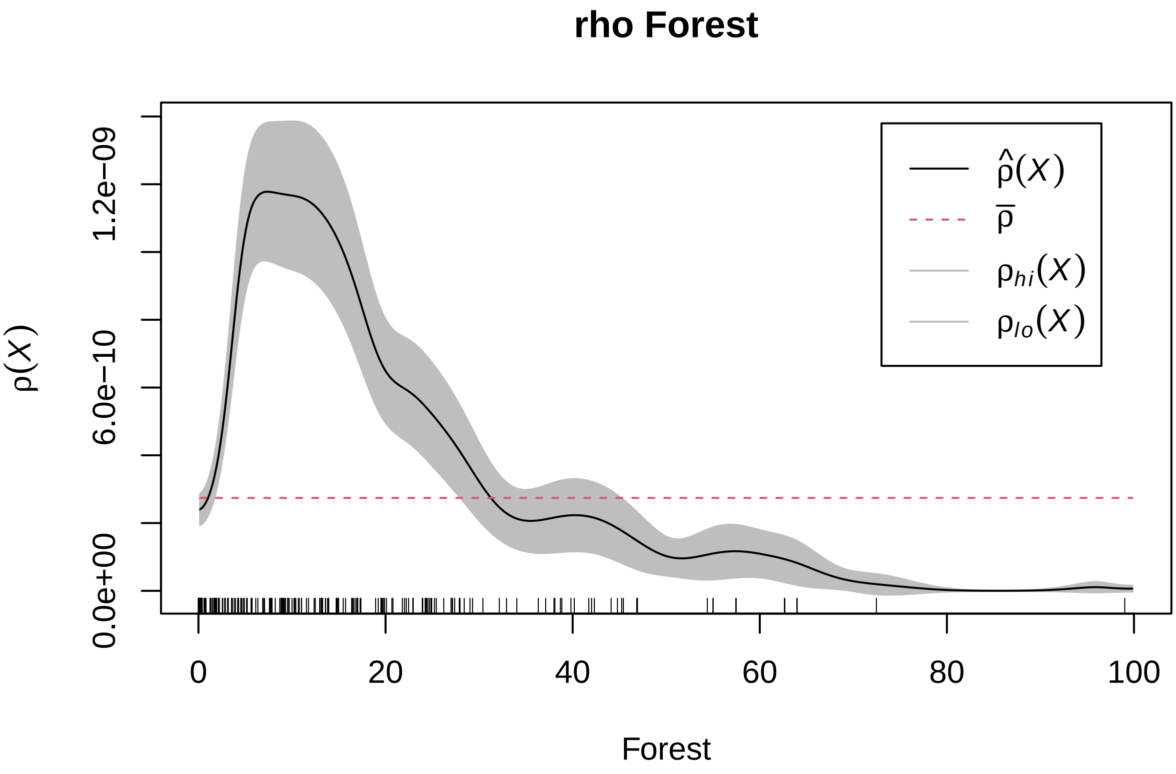


```
# Zoom rho elevation
plot(rho_elev, main="rho Elevation", xlab = "Elevation", xlim =c(0, 500))
```



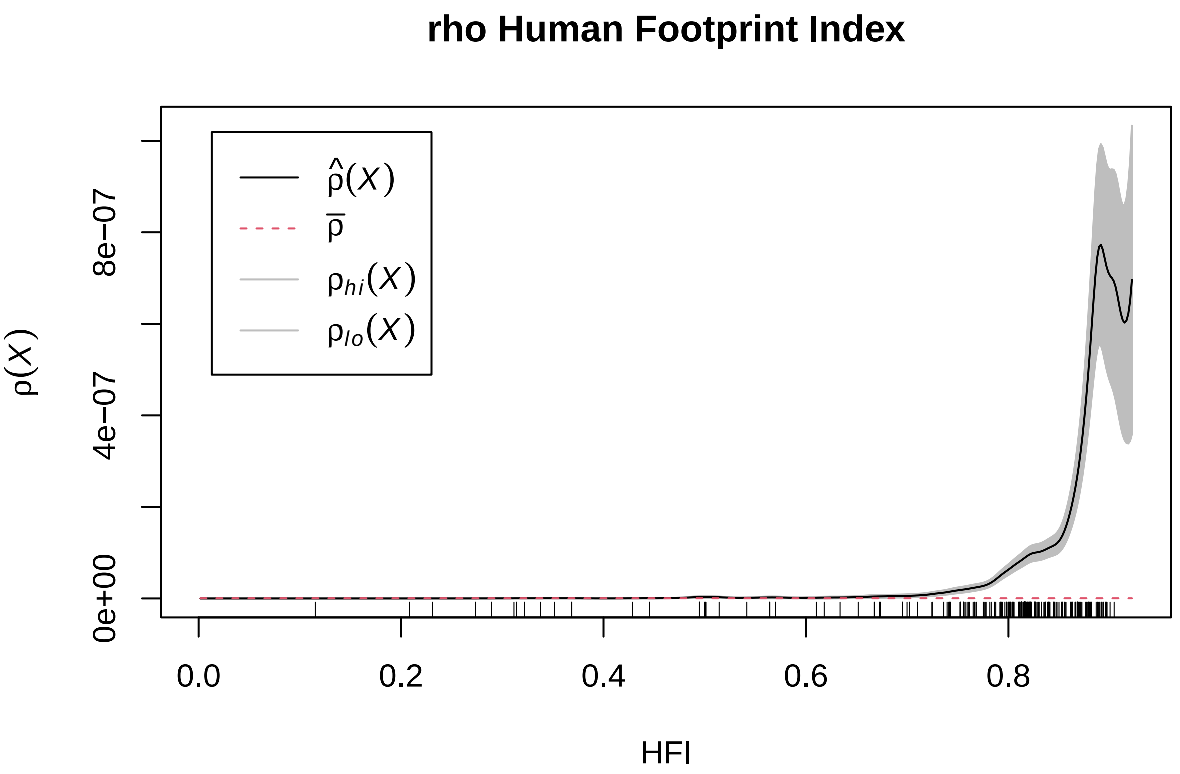
Rho – Forest

```
#Estimate Rho
rho_forest <- rhohat(humm_ppp, DATA$Forest)

plot(rho_forest,
     main="rho Forest",
     xlab = "Forest",
     xlim = c(0, max(as.vector(DATA$Forest))))
```



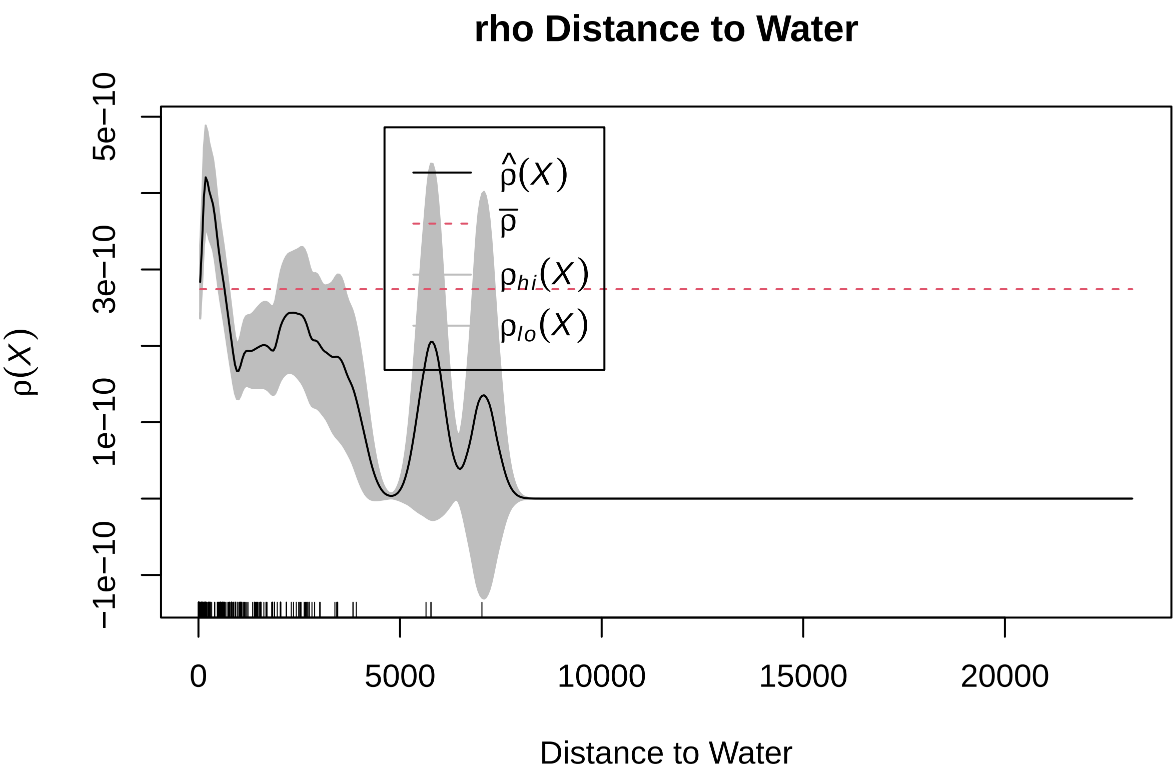
Rho – HFI

```
#Estimate Rho Human Footprint Index (HFI)
rho_hfi <- rhohat(humm_ppp, DATA$HFI)

plot(rho_hfi,
     main="rho Human Footprint Index",
     xlab = "HFI",
     xlim = c(0, max(as.vector(DATA$HFI))))
```



Rho – Distance to Water

```
#Estimate Rho Distance to Water
rho <- rhohat(humm_ppp, DATA$Dist_Water)

plot(rho,
     main="rho Distance to Water",
     xlab = "Distance to Water",
     xlim = c(0, max(as.vector(DATA$Dist_Water))))
```



Hummingbird vs BC Elevation – Median values

```
# Finding median elevation
BC_elev <- median(DATA$Elevation)
cat("BC Elevation:", BC_elev, "\n")
```

```
## BC Elevation: 1096.857
```

```
humm_elev <- median(DATA$Elevation[humm_ppp])
cat("Hummingbird Elevation:", humm_elev)
```

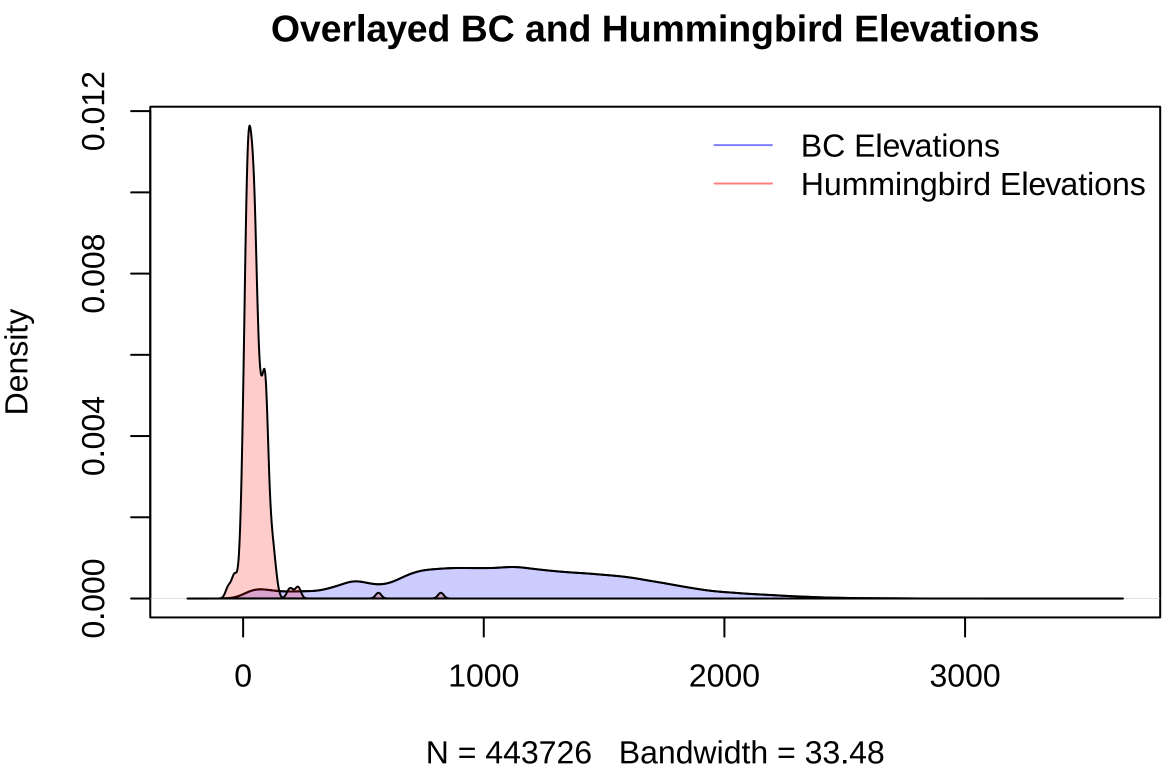
```
## Hummingbird Elevation: 45.47664
```

```
# KDE of elevation values within the province
bc_elev_kde <- density(as.numeric(DATA$Elevation))

# KDE of elevation values for hummingbirds
humm_elev_kde <- density(as.numeric(DATA$Elevation[humm_ppp]))
```

Hummingbird vs BC Elevation – KDE

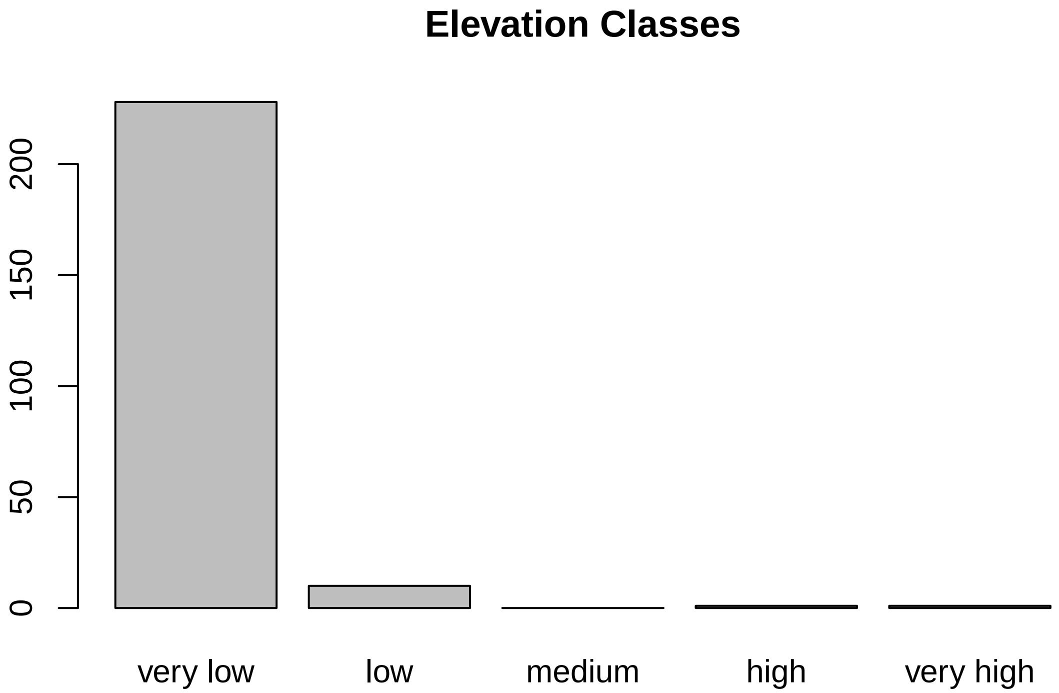
```
# Calculate the maximum value of KDEs
max_elev <- max(max(bc_elev_kde$y), max(humm_elev_kde$y))
```

```r
# Plot with adjusted y-axis limits
plot(bc_elev_kde, main="Overlayed BC and Hummingbird Elevations", ylim = c(0, max_elev))

# lines(parks_elev_kde, col=c("red", "blue"))

polygon(bc_elev_kde, col = rgb(0, 0, 1, alpha = 0.2))
polygon(humm_elev_kde, col = rgb(1, 0, 0, alpha = 0.2))

legend("topright", legend=c("BC Elevations", "Hummingbird Elevations"), col= c(rgb(0, 0, 1, alpha = 0.5
```

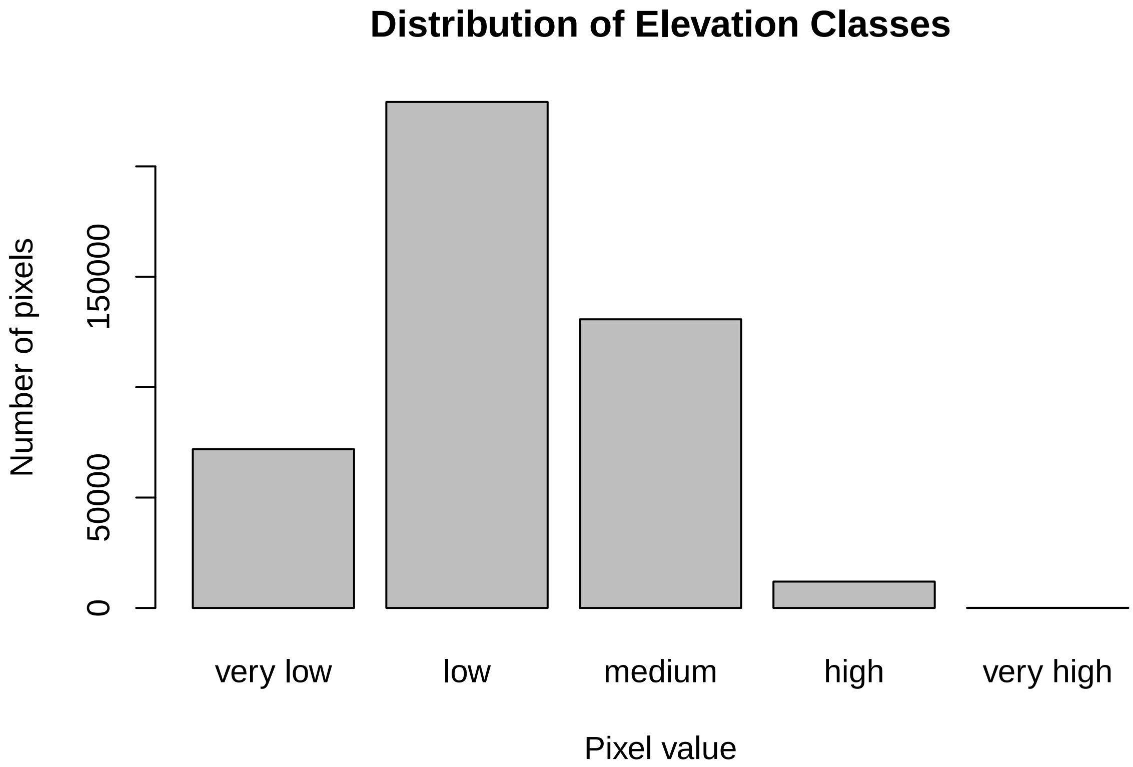


Hummingbird vs BC Elevation – Barplot counts

```r
plot(cut(as.numeric(DATA$Elevation[humm_ppp]),
         5,
         labels = c("very low", "low", "medium", "high", "very high")),
     main = "Elevation Classes")

points(transformed_longitude, transformed_latitude, pch=20, col='white' )
points(transformed_longitude, transformed_latitude, pch=20, col='black')
```



```r
hist(cut(DATA$Elevation,
         5,
         labels = c("very low", "low", "medium", "high", "very high")),
     main = "Distribution of Elevation Classes")
```



Hummingbird vs BC Forest – Median values

```
# Finding median elevation
BC_forest <- median(DATA$Forest)
cat("BC Forest median value:", BC_forest, "\n")
```

```
## BC Forest median value: 50.15247
```

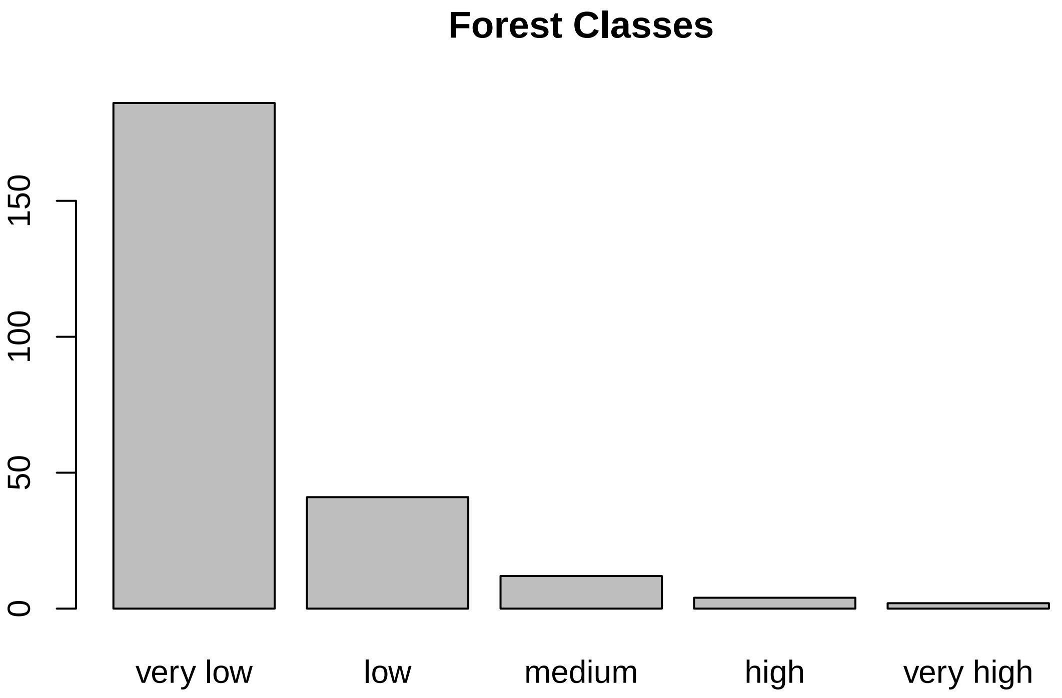
```
humm_forest <- median(DATA$Forest[humm_ppp])
cat("Hummingbird Forest median value:", humm_forest)
```

```
## Hummingbird Forest median value: 9.016245
```

```
# KDE of forest values within the province
bc_forest_kde <- density(as.numeric(DATA$Forest))

# KDE of elevation values for humming birds
humm_forest_kde <- density(as.numeric(DATA$Forest[humm_ppp]))
```

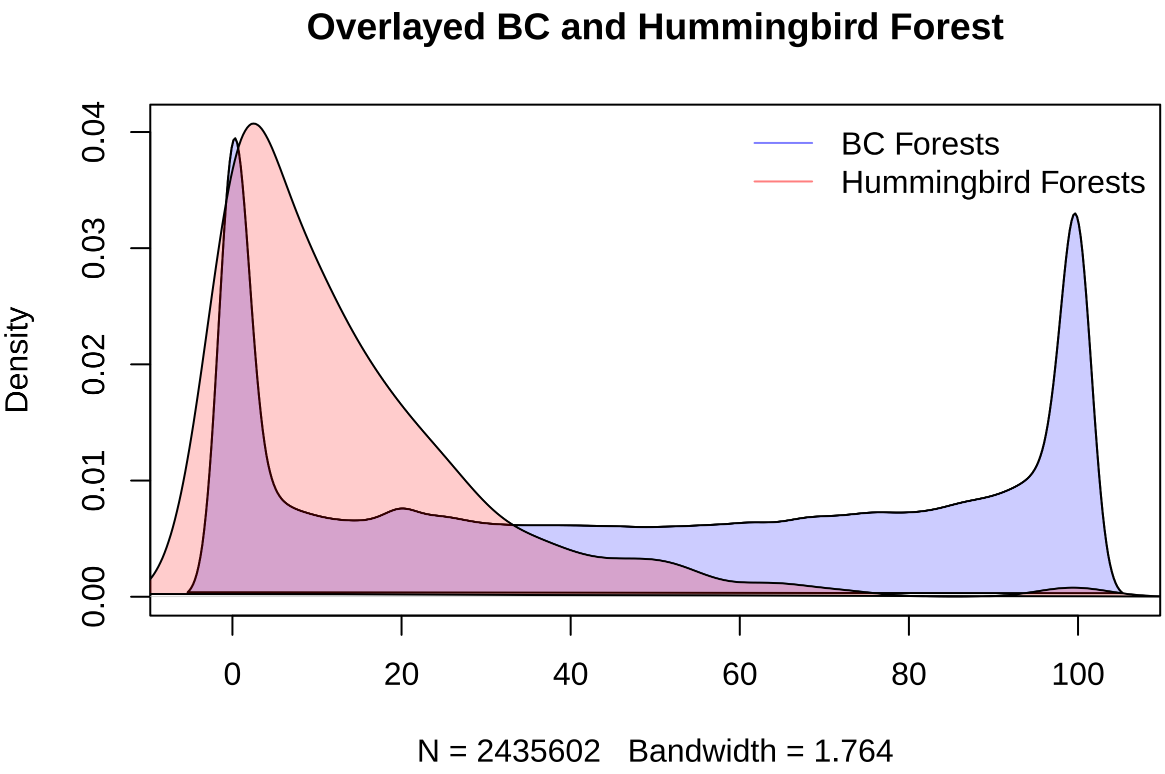
Hummingbird vs BC Forest Cover – Barplot counts

```
plot(cut(as.numeric(DATA$Forest[humm_ppp]),
         5,
         labels = c("very low", "low", "medium", "high", "very high")),
     main = "Forest Classes")
```



Hummingbird vs BC Forest Cover – KDE

```
# Calculate the maximum value of KDEs
max_forest <- max(max(bc_forest_kde$y), max(humm_forest_kde$y))

# Plot with adjusted y-axis limits
plot(bc_forest_kde, main="Overlayed BC and Hummingbird Forest", ylim = c(0, max_forest))


polygon(bc_forest_kde, col = rgb(0, 0, 1, alpha = 0.2))
polygon(humm_forest_kde, col = rgb(1, 0, 0, alpha = 0.2))

legend("topright", legend=c("BC Forests", "Hummingbird Forests"), col= c(rgb(0, 0, 1, alpha = 0.5), rgb
```



Hummingbird vs BC HFI – Median values

```
# Finding median elevation
BC_HFI <- median(DATA$HFI)
cat("BC HFI median value:", BC_HFI, "\n")
```

```
## BC HFI median value: 0.06296507
```

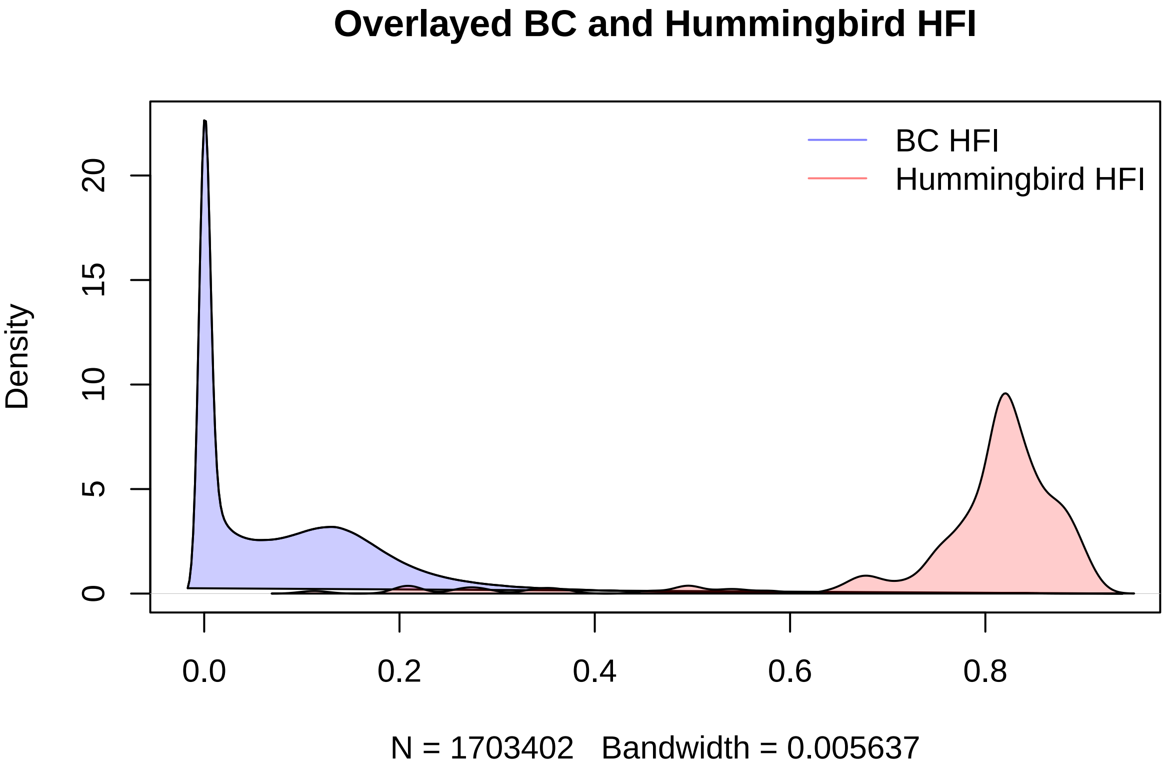
```
humm_HFI <- median(DATA$HFI[humm_ppp])
cat("Hummingbird HFI median value:", humm_HFI)
```

```
## Hummingbird HFI median value: 0.8170117
```

```
# KDE of forest values within the province
bc_HFI_kde <- density(as.numeric(DATA$HFI))
# plot(bc_elev_kde, main="BC Elevation KDE")


# KDE of elevation values for humming birds
humm_HFI_kde <- density(as.numeric(DATA$HFI[humm_ppp]))
# plot(humm_elev_kde, main="Hummingbird Elevation KDE")
```

Hummingbird vs BC HFI – KDE

```
# Calculate the maximum value of KDEs
max_HFI <- max(max(bc_HFI_kde$y), max(humm_HFI_kde$y))

# Plot with adjusted y-axis limits
plot(bc_HFI_kde, main="Overlayed BC and Hummingbird HFI", ylim = c(0, max_HFI))

# lines(parks_elev_kde, col=c("red", "blue"))

polygon(bc_HFI_kde, col = rgb(0, 0, 1, alpha = 0.2))
polygon(humm_HFI_kde, col = rgb(1, 0, 0, alpha = 0.2))

legend("topright", legend=c("BC HFI", "Hummingbird HFI"), col= c(rgb(0, 0, 1, alpha = 0.5), rgb(1, 0, 0
```



Hummingbird vs BC water – Median values

```
# Finding median elevation
BC_water <- median(DATA$Dist_Water)
cat("BC Distance to water median value:", BC_water, "\n")
```

```
## BC Distance to water median value: 1095.056
```

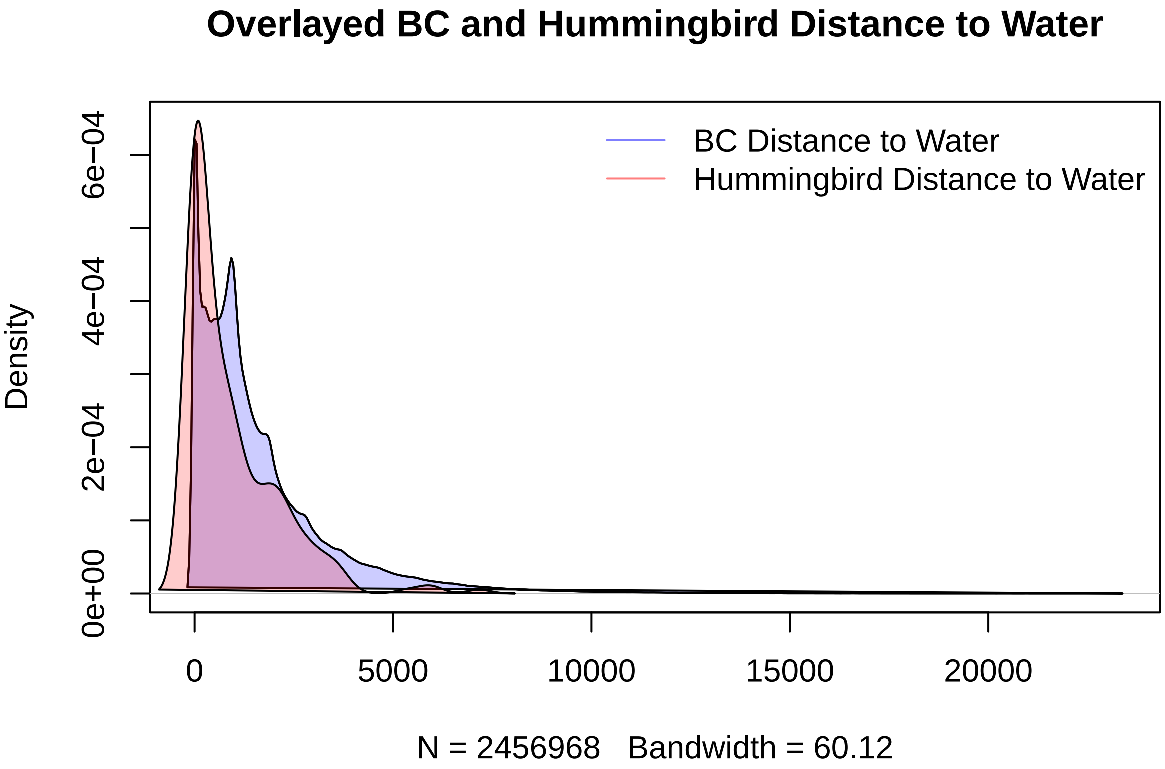
```
humm_water <- median(DATA$Dist_Water[humm_ppp])
cat("Hummingbird Distance to Water median value:", humm_water)
```

```
## Hummingbird Distance to Water median value: 536.5842
```

```
# KDE of forest values within the province
bc_water_kde <- density(as.numeric(DATA$Dist_Water))
# plot(bc_elev_kde, main="BC Elevation KDE")


# KDE of elevation values for humming birds
humm_water_kde <- density(as.numeric(DATA$Dist_Water[humm_ppp]))
# plot(humm_elev_kde, main="Hummingbird Elevation KDE")
```

Hummingbird vs BC water – KDE

```
# Calculate the maximum value of KDEs
max_water <- max(max(bc_water_kde$y), max(humm_water_kde$y))

# Plot with adjusted y-axis limits
plot(bc_water_kde, main="Overlayed BC and Hummingbird Distance to Water", ylim = c(0, max_water))

# lines(parks_elev_kde, col=c("red", "blue"))

polygon(bc_water_kde, col = rgb(0, 0, 1, alpha = 0.2))
polygon(humm_water_kde, col = rgb(1, 0, 0, alpha = 0.2))

legend("topright", legend=c("BC Distance to Water", "Hummingbird Distance to Water"), col= c(rgb(0, 0, 
```



Analysis rho function:

The figures of the rho function vs our covariates (elevation, forest cover, human foot print index and distance to water) suggest that there is likely to be a relationship with our response variable the number of hummingbirds in BC.

- Elevation exhibits a negative relationship with intensity, particularly noticeable up to elevations of 25m, indicating a decrease in the number of hummingbirds as elevation increases.
- Forest cover displays a right-skewed distribution, with intensity increasing up to 10% forest cover and decreasing thereafter.
- Distance to water demonstrates a downward trend, with the highest concentration of birds observed within 0 to 5000m of water.
- HFI exhibits a positive relationship with intensity, most clearly observed at extreme HFI values. However, with a higher HFI, there are fewer samples available to support accurate estimates.

Check for collinearity between covariates

```
# Correlation between variables
cor.im(DATA$Elevation, DATA$Forest, DATA$HFI, DATA$Dist_Water, use="pairwise.complete.obs")
```

```
##             ..1         ..2         ..3         ..4
## ..1  1.00000000 -0.26204718 -0.26625626 -0.03426387
## ..2 -0.26204718  1.00000000  0.06615541  0.04825439
## ..3 -0.26625626  0.06615541  1.00000000  0.13229408
## ..4 -0.03426387  0.04825439  0.13229408  1.00000000
```

Baseline - null model

```r
fit_null <- ppm(humm_ppp ~ 1, method="VBlogi")
fit_null
```

```
## Stationary Poisson process
## Fitted to point pattern dataset 'humm_ppp'
## Intensity: 2.714389e-10
## For standard errors, type coef(summary(x))
```

Linear Model

As equation shows, we use the log-linear to mimic our data:

$$\lambda_{parks}(u) = e^{\alpha+\beta_{elevation}Elevation(u)+\beta_{forest}Forest(u)+\beta_{HFI}HFI(u)+\beta_{dist.water}DistWater(u)}$$

Due to the normal distribution, the model can not converge so we use the `method="VBlogi"`

```r
DATA$HFI_change <- as.owin(DATA$HFI,na.replace=0)
Dist_Water <- as.owin(DATA$Dist_Water,na.replace=0)
fit_linear <- ppm(humm_ppp ~ Elevation + Forest + HFI_change + Dist_Water, data = DATA, method="VBlogi"
fit_linear
```

```
## Nonstationary Poisson process
## Fitted to point pattern dataset 'humm_ppp'
##
## Log intensity:  ~Elevation + Forest + HFI_change + Dist_Water
##
## Fitted trend coefficients:
##    (Intercept)      Elevation         Forest HFI_changeTRUE     Dist_Water
##  -1.741954e+01  -1.179908e-02  -4.885297e-02   8.067290e-01  -1.612941e-04
##
## For standard errors, type coef(summary(x))
```

Model selection Because the BVlogit object used in linear models cannot be applied to ANOVA, we employ both AIC and loss ratio to select the model. The results indicate that the linear model outperforms the reduced model.

```r
# AIC values
cat("AIC for our linear model proposal: ", AIC(fit_linear))
```

```
## AIC for our linear model proposal:  11231.66
```

```r
cat("\nAIC for the intercept only model: ", AIC(fit_null), "\n")
```

```
##
## AIC for the intercept only model:  12079.07
```

```
# Delta AIC
cat("\nDelta AIC: ", AIC(fit_linear) - AIC(fit_null), "\n")
```

```
##
## Delta AIC:  -847.4162
```

```
## Use loss ratio
# install.packages("lmtest")
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: ’zoo’
```

```
## The following objects are masked from ’package:base’:
##
##     as.Date, as.Date.numeric
```

```
lrtest(fit_null, fit_linear)
```

```
## Likelihood ratio test
##
## Model 1: ~1
## Model 2: ~Elevation + Forest + HFI_change + Dist_Water
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   1 -6038.5
## 2   5 -5610.8  4 855.42  < 2.2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

Quadratic Model

There are clearly nonlinear relationships between hummingbirds and all four variables. Therefore, we will proceed by fitting a quadratic model first. This warning suggests that the algorithm used for fitting a model did not converge, meaning it did not reach a stable solution. This could happen for various reasons, such as starting with poor initial parameter values, numerical instability, or the model being too complex for the data.

```
fit_q1 <- ppm(humm_ppp ~ Elevation + I(Elevation^2) + Forest + I(Forest^2) + HFI_change + I(HFI_change^
```

```
## Warning: glm.fit: algorithm did not converge
```

```
fit_q1
```

```
## Warning: Cannot compute variance; model is not valid
## Nonstationary Poisson process
## Fitted to point pattern dataset ’humm_ppp’
##
## Log intensity:  ~Elevation + I(Elevation^2) + Forest + I(Forest^2) + HFI_change
## + I(HFI_change^2) + Dist_Water + I(Dist_Water^2)
##
## Fitted trend coefficients:
##     (Intercept)       Elevation  I(Elevation^2)          Forest     I(Forest^2)
##   -1.658015e+01   -1.669589e-02    4.627629e-06   -3.396606e-02   -3.450510e-04
##  HFI_changeTRUE I(HFI_change^2)      Dist_Water I(Dist_Water^2)
##    6.944647e-01              NA    1.776642e-04   -6.054844e-08
##
## Standard errors unavailable; Fisher information matrix is singular
## *** Fitting algorithm for ’glm’ did not converge ***
##
## *** Model is not valid ***
## *** Some coefficients are NA or Inf ***
```

Due to the presence of NA and the unstable process, we think to reduce the parameters of our model.

```
fit_q2 <- ppm(humm_ppp ~ Elevation + I(Elevation^2),data=DATA,method="VBlogi")
fit_q2
```

```
## Nonstationary Poisson process
## Fitted to point pattern dataset ’humm_ppp’
##
## Log intensity:  ~Elevation + I(Elevation^2)
##
## Fitted trend coefficients:
##    (Intercept)      Elevation I(Elevation^2)
##  -1.735511e+01  -2.119553e-02   7.413727e-06
##
## For standard errors, type coef(summary(x))
```

```
fit_q3 <- ppm(humm_ppp ~ Forest + I(Forest^2),data=DATA,method="VBlogi")
fit_q3
```

```
## Nonstationary Poisson process
## Fitted to point pattern dataset ’humm_ppp’
##
## Log intensity:  ~Forest + I(Forest^2)
##
## Fitted trend coefficients:
##   (Intercept)        Forest   I(Forest^2)
## -21.233785705   0.008791772  -0.000680389
##
## For standard errors, type coef(summary(x))
```

```
fit_q4 <- ppm(humm_ppp ~  HFI_change + I(HFI_change^2),data=DATA,method="VBlogi")
fit_q4
```

```
## Nonstationary Poisson process
## Fitted to point pattern dataset 'humm_ppp'
##
## Log intensity:  ~HFI_change + I(HFI_change^2)
##
## Fitted trend coefficients:
##     (Intercept)  HFI_changeTRUE I(HFI_change^2)
##      -19.782628       -1.180374       -1.180374
##
## For standard errors, type coef(summary(x))
```

```
fit_q5 <- ppm(humm_ppp ~  Dist_Water + I(Dist_Water^2),data=DATA,method="VBlogi")
fit_q5
```

```
## Nonstationary Poisson process
## Fitted to point pattern dataset 'humm_ppp'
##
## Log intensity:  ~Dist_Water + I(Dist_Water^2)
##
## Fitted trend coefficients:
##     (Intercept)      Dist_Water I(Dist_Water^2)
##   -2.132543e+01   -6.756418e-04    3.990832e-08
##
## For standard errors, type coef(summary(x))
```

Model selection All quadratic elevation, HFI and distance from water models are better than linear one, while forest cover is worse.

```
# AIC values
cat("AIC for our linear model proposal: ", AIC(fit_linear))
```

```
## AIC for our linear model proposal:  11231.66
```

```
cat("\nAIC for quadratic elevation model: ", AIC(fit_q2), "\n")
```

```
##
## AIC for quadratic elevation model:  11204.59
```

```
# Delta AIC
cat("\nDelta AIC: ", AIC(fit_linear) - AIC(fit_q2), "\n")
```

```
##
## Delta AIC:  27.06284
```

```
## Use loss ratio
lrtest(fit_linear, fit_q2)
```

```
## Likelihood ratio test
##
## Model 1: ~Elevation + Forest + HFI_change + Dist_Water
## Model 2: ~Elevation + I(Elevation^2)
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   5 -5610.8
## 2   3 -5599.3 -2 23.063  9.817e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# AIC values
cat("AIC for our linear model proposal: ", AIC(fit_linear))
```

```
## AIC for our linear model proposal:  11231.66
```

```
cat("\nAIC for quadratic forest model: ", AIC(fit_q3), "\n")
```

```
##
## AIC for quadratic forest model:  11901.82
```

```
# Delta AIC
cat("\nDelta AIC: ", AIC(fit_linear) - AIC(fit_q3), "\n")
```

```
##
## Delta AIC:  -670.1669
```

```
## Use loss ratio
lrtest(fit_linear, fit_q3)
```

```
## Likelihood ratio test
##
## Model 1: ~Elevation + Forest + HFI_change + Dist_Water
## Model 2: ~Forest + I(Forest^2)
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   5 -5610.8
## 2   3 -5947.9 -2 674.17  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# AIC values
cat("AIC for our linear model proposal: ", AIC(fit_linear))
```

```
## AIC for our linear model proposal:  11231.66
```

```
cat("\nAIC for quadratic HFI model: ", AIC(fit_q4), "\n")
```

```
##
## AIC for quadratic HFI model:  12029.77
```

```
# Delta AIC
cat("\nDelta AIC: ", AIC(fit_linear) - AIC(fit_q4), "\n")
```

```
##
## Delta AIC:  -798.1126
```

```r
## Use loss ratio
lrtest(fit_linear, fit_q4)
```

```
## Likelihood ratio test
##
## Model 1: ~Elevation + Forest + HFI_change + Dist_Water
## Model 2: ~HFI_change + I(HFI_change^2)
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   5 -5610.8
## 2   3 -6011.9 -2 802.11  < 2.2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```r
# AIC values
cat("AIC for our linear model proposal: ", AIC(fit_linear))
```

```
## AIC for our linear model proposal:  11231.66
```

```r
cat("\nAIC for quadratic distance to water model: ", AIC(fit_q5), "\n")
```

```
##
## AIC for quadratic distance to water model:  12094.03
```

```r
# Delta AIC
cat("\nDelta AIC: ", AIC(fit_linear) - AIC(fit_q5), "\n")
```

```
##
## Delta AIC:  -862.3713
```

```r
## Use loss ratio
lrtest(fit_linear, fit_q5)
```

```
## Likelihood ratio test
##
## Model 1: ~Elevation + Forest + HFI_change + Dist_Water
## Model 2: ~Dist_Water + I(Dist_Water^2)
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   5 -5610.8
## 2   3 -6044.0 -2 866.37  < 2.2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

Due to the failure of convergence in the quadratic models with reduced variables, we compared them with the original unconverged model, which includes all quadratic covariates. And the results indicate that the original model performs better, even surpassing the linear model.

```r
fit_qfinal <- ppm(humm_ppp ~ Elevation + I(Elevation^2) + Forest + I(HFI_change^2) + Dist_Water + I(Dis
```

```
## Warning: glm.fit: algorithm did not converge
```

```
fit_qfinal
```

```
## Nonstationary Poisson process
## Fitted to point pattern dataset 'humm_ppp'
##
## Log intensity:  ~Elevation + I(Elevation^2) + Forest + I(HFI_change^2) +
## Dist_Water + I(Dist_Water^2)
##
## Fitted trend coefficients:
##     (Intercept)       Elevation  I(Elevation^2)          Forest I(HFI_change^2)
##   -1.650240e+01   -1.672429e-02    4.626687e-06   -5.443703e-02    7.509351e-01
##      Dist_Water I(Dist_Water^2)
##    1.547740e-04   -5.640326e-08
##
##                      Estimate         S.E.       CI95.lo       CI95.hi Ztest
## (Intercept)     -1.650240e+01 1.681080e-01 -1.683189e+01 -1.617291e+01   ***
## Elevation       -1.672429e-02 9.277712e-04 -1.854269e-02 -1.490589e-02   ***
## I(Elevation^2)   4.626687e-06 3.227563e-07  3.994096e-06  5.259277e-06   ***
## Forest          -5.443703e-02 3.349432e-03 -6.100180e-02 -4.787227e-02   ***
## I(HFI_change^2)  7.509351e-01 1.914980e-01  3.756060e-01  1.126264e+00   ***
## Dist_Water       1.547740e-04 1.131860e-04 -6.706646e-05  3.766145e-04
## I(Dist_Water^2) -5.640326e-08 2.291975e-08 -1.013251e-07 -1.148138e-08     *
##                        Zval
## (Intercept)     -98.165450
## Elevation       -18.026307
## I(Elevation^2)   14.334925
## Forest          -16.252615
## I(HFI_change^2)   3.921374
## Dist_Water        1.367431
## I(Dist_Water^2)  -2.460902
## *** Fitting algorithm for 'glm' did not converge ***
```

```
# AIC values
cat("AIC for final quadratic model : ", AIC(fit_qfinal))
```

```
## AIC for final quadratic model :  9631.574
```

```
cat("\nAIC for origin quadratic model: ", AIC(fit_q1), "\n")
```

```
##
## AIC for origin quadratic model:  9630.002
```

```
# Delta AIC
cat("\nDelta AIC: ", AIC(fit_qfinal) - AIC(fit_q1), "\n")
```

```
##
## Delta AIC:  1.57142
```

```
## Use loss ratio
lrtest(fit_qfinal, fit_q1)
```

```
## Likelihood ratio test
##
## Model 1: ~Elevation + I(Elevation^2) + Forest + I(HFI_change^2) + Dist_Water +
##     I(Dist_Water^2)
## Model 2: ~Elevation + I(Elevation^2) + Forest + I(Forest^2) + HFI_change +
##     I(HFI_change^2) + Dist_Water + I(Dist_Water^2)
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -4808.8
## 2   9 -4806.0  2 5.5714    0.06169 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# AIC values
cat("AIC for linear model : ", AIC(fit_linear))
```

```
## AIC for linear model :  11231.66
```

```
cat("\nAIC for origin quadratic model: ", AIC(fit_q1), "\n")
```

```
##
## AIC for origin quadratic model:  9630.002
```

```
# Delta AIC
cat("\nDelta AIC: ", AIC(fit_linear) - AIC(fit_q1), "\n")
```

```
##
## Delta AIC:  1601.653
```

```
## Use loss ratio
lrtest(fit_linear, fit_q1)
```

```
## Likelihood ratio test
##
## Model 1: ~Elevation + Forest + HFI_change + Dist_Water
## Model 2: ~Elevation + I(Elevation^2) + Forest + I(Forest^2) + HFI_change +
##     I(HFI_change^2) + Dist_Water + I(Dist_Water^2)
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   5 -5610.8
## 2   9 -4806.0  4 1609.7  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

GAM Framework: Model fit rho shape and additive modelling framework

```
library(splines)

fit_rhogam <-ppm(humm_ppp ~  Elevation + I(Elevation^2) + Forest +I(Forest^2)+HFI_change+I(HFI_change^2
fit_rhogam
```

```
## Nonstationary Poisson process
## Fitted to point pattern dataset ’humm_ppp’
##
## Log intensity:  ~Elevation + I(Elevation^2) + Forest + I(Forest^2) + HFI_change
## + I(HFI_change^2) + bs(Dist_Water, df = 8)
##
## Fitted trend coefficients:
##             (Intercept)               Elevation          I(Elevation^2)
##           -1.647311e+01           -1.853627e-02            4.782343e-06
##                  Forest             I(Forest^2)          HFI_changeTRUE
##           -3.306880e-02           -4.231002e-04            0.000000e+00
##         I(HFI_change^2) bs(Dist_Water, df = 8)1 bs(Dist_Water, df = 8)2
##            5.722317e-01           -1.334931e+00            2.138256e+00
## bs(Dist_Water, df = 8)3 bs(Dist_Water, df = 8)4 bs(Dist_Water, df = 8)5
##           -1.926349e-01            8.165015e-01            1.839645e-01
## bs(Dist_Water, df = 8)6 bs(Dist_Water, df = 8)7 bs(Dist_Water, df = 8)8
##           -3.096686e+00            4.474116e+01           -6.276349e+02
##
## For standard errors, type coef(summary(x))
```

Model selection

- “rhogam” is better than linear model and have similar resule of quadratic model.

```
# AIC values
cat("AIC for linear model : ", AIC(fit_linear))
```

```
## AIC for linear model :  11231.66
```

```
cat("\nAIC for rhogam model: ", AIC(fit_rhogam), "\n")
```

```
##
## AIC for rhogam model:  9620.369
```

```
# Delta AIC
cat("\nDelta AIC: ", AIC(fit_linear) - AIC(fit_rhogam), "\n")
```

```
##
## Delta AIC:  1611.287
```

```
## Use loss ratio
lrtest(fit_linear, fit_rhogam)
```

```
## Likelihood ratio test
##
## Model 1: ~Elevation + Forest + HFI_change + Dist_Water
## Model 2: ~Elevation + I(Elevation^2) + Forest + I(Forest^2) + HFI_change +
##     I(HFI_change^2) + bs(Dist_Water, df = 8)
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   5 -5610.8
## 2  15 -4795.2 10 1631.3  < 2.2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
# AIC values
cat("AIC for quadratic model : ", AIC(fit_q1))
```

```
## AIC for quadratic model :  9630.002
```

```
cat("\nAIC for rhogam model: ", AIC(fit_rhogam), "\n")
```

```
##
## AIC for rhogam model:  9620.369
```

```
# Delta AIC
cat("\nDelta AIC: ", AIC(fit_q1) - AIC(fit_rhogam), "\n")
```

```
##
## Delta AIC:  9.633651
```

```
## Use loss ratio
lrtest(fit_q1, fit_rhogam)
```

```
## Likelihood ratio test
##
## Model 1: ~Elevation + I(Elevation^2) + Forest + I(Forest^2) + HFI_change +
##     I(HFI_change^2) + Dist_Water + I(Dist_Water^2)
## Model 2: ~Elevation + I(Elevation^2) + Forest + I(Forest^2) + HFI_change +
##     I(HFI_change^2) + bs(Dist_Water, df = 8)
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   9 -4806.0
## 2  15 -4795.2  6 21.634   0.001411 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

GAM model

Finally, we aimed to explore a more complex model, namely GAM. The results demonstrate that it is the most superior model we have encountered thus far.

```
fit_gam <-ppm(humm_ppp ~  bs(Elevation,df=3)+ bs(Forest,df=3)+ bs(HFI_change,df=3) + bs(Dist_Water,df=8
fit_gam
```

```
## Nonstationary Poisson process
## Fitted to point pattern dataset ’humm_ppp’
##
## Log intensity:  ~bs(Elevation, df = 3) + bs(Forest, df = 3) + bs(HFI_change, df
## = 3) + bs(Dist_Water, df = 8)
##
## Fitted trend coefficients:
##             (Intercept)  bs(Elevation, df = 3)1  bs(Elevation, df = 3)2
##             -15.4418349             -21.7766176             -30.2661224
##  bs(Elevation, df = 3)3     bs(Forest, df = 3)1     bs(Forest, df = 3)2
##              -7.8824194               0.6514961              -7.2018404
##     bs(Forest, df = 3)3 bs(HFI_change, df = 3)1 bs(HFI_change, df = 3)2
##              -4.0651657               0.0000000               0.0000000
## bs(HFI_change, df = 3)3 bs(Dist_Water, df = 8)1 bs(Dist_Water, df = 8)2
##               0.5193305              -1.3088064               2.1582454
## bs(Dist_Water, df = 8)3 bs(Dist_Water, df = 8)4 bs(Dist_Water, df = 8)5
##              -0.2973903               0.8063324               0.2482539
## bs(Dist_Water, df = 8)6 bs(Dist_Water, df = 8)7 bs(Dist_Water, df = 8)8
##              -3.4528640              47.4756499            -661.3967573
##
## For standard errors, type coef(summary(x))
```

```r
# AIC values
cat("AIC for gam model : ", AIC(fit_gam))
```

Model selection

```
## AIC for gam model :  9612.959
```

```r
cat("\nAIC for rhogam model: ", AIC(fit_rhogam), "\n")
```

```
##
## AIC for rhogam model:  9620.369
```

```r
# Delta AIC
cat("\nDelta AIC: ", AIC(fit_gam) - AIC(fit_rhogam), "\n")
```

```
##
## Delta AIC:  -7.409346
```

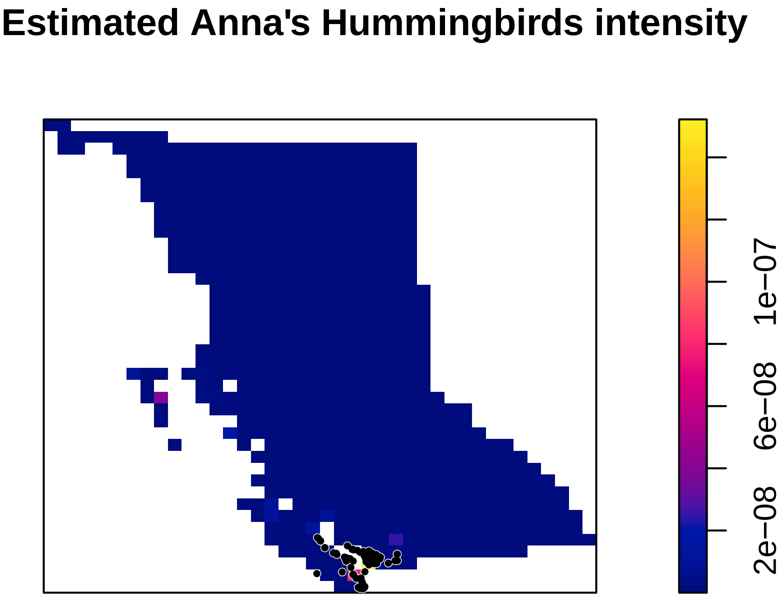
```r
## Use loss ratio
lrtest(fit_rhogam, fit_gam)
```

```
## Likelihood ratio test
##
## Model 1: ~Elevation + I(Elevation^2) + Forest + I(Forest^2) + HFI_change +
##     I(HFI_change^2) + bs(Dist_Water, df = 8)
## Model 2: ~bs(Elevation, df = 3) + bs(Forest, df = 3) + bs(HFI_change,
##     df = 3) + bs(Dist_Water, df = 8)
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  15 -4795.2
## 2  18 -4788.5  3 13.409    0.00383 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

Model evaluation

Predict

```
# Plot the model fitted
plot(fit_gam,
     se = FALSE,
     superimpose = FALSE,
     main = "Estimated Anna's Hummingbirds intensity")

# Overlay the Parks locations
points(humm_ppp, pch = 16, cex = 0.6, col = "white")
points(humm_ppp, pch = 16, cex = 0.5, col = "black")
```



Q-test

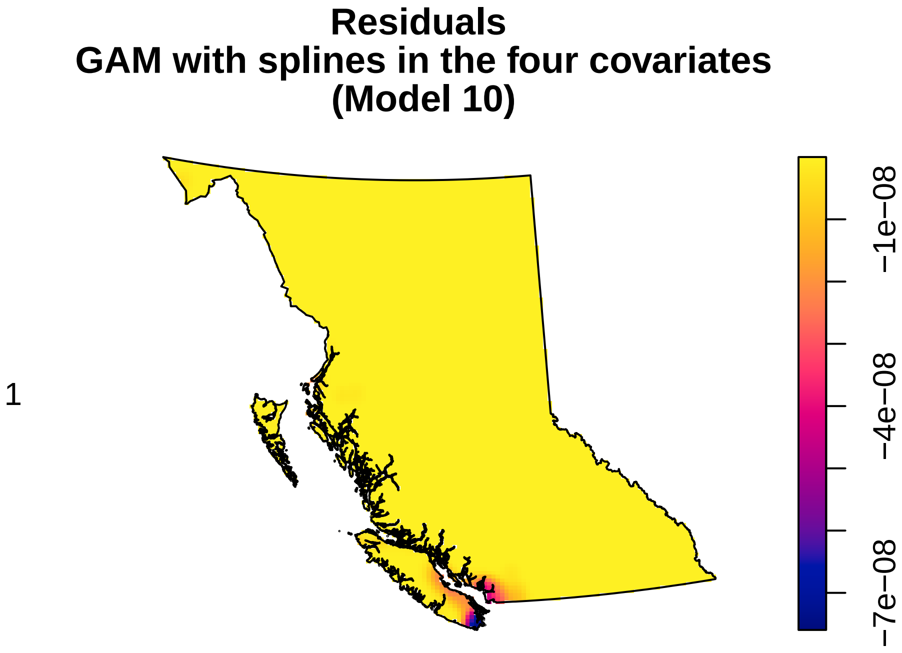
The small p value tells us that there's a significant deviation from our model's predictions. While this is useful for suggesting that our model has room for improvement, it provides us with no direction on how to do so (e.g., missing parameters, model mispecification (e.g., polynomial vs. linear), a lack of independence, non-stationarity, etc…).

```
# Quadrat counting for significant deviations from our intensity function using chi-squared test
# Performs a goodness-of-fit test of a fitted inhomogeneous Poisson model
quadrat.test(fit_gam, nx =2 , ny = 10)
```

```
## Warning in bs(Elevation, degree = 3L, knots = numeric(0), Boundary.knots =
## c(-61.2767014431655, : some 'x' values beyond boundary knots may cause
## ill-conditioned bases

## Warning: Some expected counts are small; chi^2 approximation may be inaccurate

##
##  Chi-squared test of fitted Poisson model 'fit_gam' using quadrat counts
##
## data:  data from fit_gam
## X2 = 213.86, df = 2, p-value < 2.2e-16
## alternative hypothesis: two.sided
##
## Quadrats: 20 tiles (irregular windows)
```

Residuals plot

```
#Calculate the residuals
res <- residuals(fit_gam)

#Visualise
plot(res,
     cols = "transparent",
     main = "Residuals \n GAM with splines in the four covariates \n(Model 10)",
     main.cex = 0.6)
```



```
#Calculate the partial residuals as a function of Elevation
par_res_elev <- parres(fit_gam, "Elevation")
```

Partial residuals

```
## Warning in bs(Dist_Water, df = 8): all interior knots match left boundary knot
```

```
## Warning in bs(Dist_Water, df = 8): all interior knots match right boundary knot
```

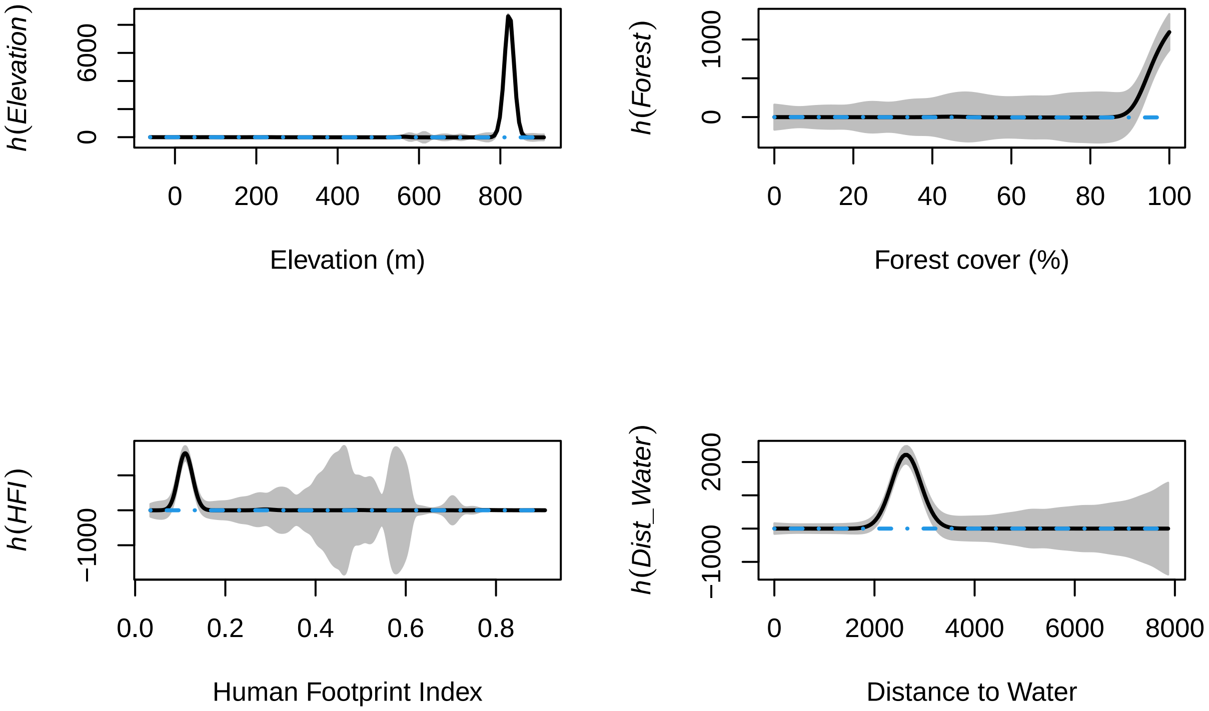
```
#Calculate the relative intensity as a function of Forest cover
par_res_fc <- parres(fit_gam, "Forest")
```

```
## Warning in bs(Dist_Water, df = 8): all interior knots match left boundary knot

## Warning in bs(Dist_Water, df = 8): all interior knots match right boundary knot
```

```
#Calculate the relative intensity as a function of Distance from water
par_res_water <- parres(fit_gam, "Dist_Water")

#Calculate the relative intensity as a function of HFI
par_res_HFI <- parres(fit_gam, "HFI")
```

```
## Warning: Values for 1 query point lying outside the pixel image domain were
## estimated by projection to the nearest pixel
```

```
#Side by side plotting
par(mfrow = c(2,2))
plot(par_res_elev,
     legend = FALSE,
     lwd = 2,
     main = "",
     xlab = "Elevation (m)")
plot(par_res_fc,
     legend = FALSE,
     lwd = 2,
     main = "",
     xlab = "Forest cover (%)")
plot(par_res_HFI,
     legend = FALSE,
     lwd = 2,
     main = "",
     xlab = "Human Footprint Index")
plot(par_res_water,
     legend = FALSE,
     lwd = 2,
     main = "",
     xlab = "Distance to Water")
```



Lurking plot

```
par(mfrow = c(2,2))
lurk_ele<-lurking (fit_gam, DATA$Elevation ,type = "raw",cumulative = F,envelope = T,xlab = "Gradient",
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 28 out of 2159 quadrature points discarded because they lie outside the domain
## of the covariate image

## Warning in bs(Elevation, degree = 3L, knots = numeric(0), Boundary.knots =
## c(-61.2767014431655, : some 'x' values beyond boundary knots may cause
## ill-conditioned bases

## Generating 39 simulated patterns ...1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 1
## 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38,
## 39.
## Processing..

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 8 out of 2127 quadrature points discarded because they lie outside the domain
## of the covariate image

## 1,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 8 out of 2167 quadrature points discarded because they lie outside the domain
## of the covariate image
## 2,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 9 out of 2181 quadrature points discarded because they lie outside the domain
## of the covariate image

## 3,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 8 out of 2186 quadrature points discarded because they lie outside the domain
## of the covariate image

## 4,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 8 out of 2168 quadrature points discarded because they lie outside the domain
## of the covariate image

## 5,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 10 out of 2165 quadrature points discarded because they lie outside the domain
## of the covariate image

## 6,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 11 out of 2178 quadrature points discarded because they lie outside the domain
## of the covariate image

## 7,

## Warning: Values of the covariate ’Elevation’ were NA or undefined at 0.05% (1
## out of 2157) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,

## Warning: Some infinite, NA or NaN increments were removed

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 13 out of 2157 quadrature points discarded because they lie outside the domain
## of the covariate image

## 8,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 10 out of 2173 quadrature points discarded because they lie outside the domain
## of the covariate image

## 9,
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 11 out of 2174 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 10,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 8 out of 2190 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 11,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 9 out of 2187 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 12,
```

```
## Warning: Values of the covariate ’Elevation’ were NA or undefined at 0.05% (1
## out of 2195) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,
```

```
## Warning: Some infinite, NA or NaN increments were removed
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 11 out of 2195 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 13,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 10 out of 2199 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 14,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 9 out of 2183 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 15,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 9 out of 2194 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 16,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 9 out of 2177 quadrature points discarded because they lie outside the domain
## of the covariate image
## 17,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 11 out of 2162 quadrature points discarded because they lie outside the domain
## of the covariate image

## 18,

## Warning: Values of the covariate 'Elevation' were NA or undefined at 0.05% (1
## out of 2184) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,

## Warning: Some infinite, NA or NaN increments were removed

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 8 out of 2184 quadrature points discarded because they lie outside the domain
## of the covariate image

## 19,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 10 out of 2170 quadrature points discarded because they lie outside the domain
## of the covariate image

## 20,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 9 out of 2197 quadrature points discarded because they lie outside the domain
## of the covariate image

## 21,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 9 out of 2155 quadrature points discarded because they lie outside the domain
## of the covariate image

## 22,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 10 out of 2162 quadrature points discarded because they lie outside the domain
## of the covariate image

## 23,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 10 out of 2162 quadrature points discarded because they lie outside the domain
## of the covariate image

## 24,
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 8 out of 2174 quadrature points discarded because they lie outside the domain
## of the covariate image

## 25,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 10 out of 2176 quadrature points discarded because they lie outside the domain
## of the covariate image

## 26,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 10 out of 2158 quadrature points discarded because they lie outside the domain
## of the covariate image

## 27,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 8 out of 2162 quadrature points discarded because they lie outside the domain
## of the covariate image

## 28,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 11 out of 2161 quadrature points discarded because they lie outside the domain
## of the covariate image

## 29,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 9 out of 2178 quadrature points discarded because they lie outside the domain
## of the covariate image

## 30,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 9 out of 2156 quadrature points discarded because they lie outside the domain
## of the covariate image

## 31,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 9 out of 2219 quadrature points discarded because they lie outside the domain
## of the covariate image

## 32,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 9 out of 2179 quadrature points discarded because they lie outside the domain
## of the covariate image
## 33,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 8 out of 2186 quadrature points discarded because they lie outside the domain
## of the covariate image

## 34,

## Warning: Values of the covariate 'Elevation' were NA or undefined at 0.05% (1
## out of 2166) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,

## Warning: Some infinite, NA or NaN increments were removed

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 8 out of 2166 quadrature points discarded because they lie outside the domain
## of the covariate image

## 35,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 10 out of 2156 quadrature points discarded because they lie outside the domain
## of the covariate image

## 36,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 8 out of 2168 quadrature points discarded because they lie outside the domain
## of the covariate image

## 37,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 11 out of 2188 quadrature points discarded because they lie outside the domain
## of the covariate image

## 38,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 10 out of 2193 quadrature points discarded because they lie outside the domain
## of the covariate image

##
## 39.
## Done.
```

```
lurk_forest<-lurking (fit_gam, DATA$Forest ,type = "raw",cumulative = F,envelope = T,xlab = "Gradient",
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 28 out of 2159 quadrature points discarded because they lie outside the domain
## of the covariate image
## Warning in bs(Elevation, degree = 3L, knots = numeric(0), Boundary.knots =
## c(-61.2767014431655, : some 'x' values beyond boundary knots may cause
## ill-conditioned bases

## Generating 39 simulated patterns ...1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19
## 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38,
## 39.
## Processing..

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 17 out of 2199 quadrature points discarded because they lie outside the domain
## of the covariate image

## 1,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 17 out of 2185 quadrature points discarded because they lie outside the domain
## of the covariate image

## 2,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 17 out of 2196 quadrature points discarded because they lie outside the domain
## of the covariate image

## 3,

## Warning: Values of the covariate 'Forest' were NA or undefined at 0.05% (1 out
## of 2179) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,

## Warning: Some infinite, NA or NaN increments were removed

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 16 out of 2179 quadrature points discarded because they lie outside the domain
## of the covariate image

## 4,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 17 out of 2177 quadrature points discarded because they lie outside the domain
## of the covariate image

## 5,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 15 out of 2176 quadrature points discarded because they lie outside the domain
## of the covariate image

## 6,
## Warning: Values of the covariate 'Elevation' were NA or undefined at 0.05% (1
## out of 2179) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,

## Warning: Some infinite, NA or NaN increments were removed

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 15 out of 2179 quadrature points discarded because they lie outside the domain
## of the covariate image

## 7,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 16 out of 2198 quadrature points discarded because they lie outside the domain
## of the covariate image

## 8,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 14 out of 2177 quadrature points discarded because they lie outside the domain
## of the covariate image

## 9,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 16 out of 2181 quadrature points discarded because they lie outside the domain
## of the covariate image

## 10,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 16 out of 2177 quadrature points discarded because they lie outside the domain
## of the covariate image

## 11,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 19 out of 2181 quadrature points discarded because they lie outside the domain
## of the covariate image

## 12,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 19 out of 2194 quadrature points discarded because they lie outside the domain
## of the covariate image

## 13,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 14 out of 2170 quadrature points discarded because they lie outside the domain
## of the covariate image
## 14,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 15 out of 2177 quadrature points discarded because they lie outside the domain
## of the covariate image

## 15,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 14 out of 2172 quadrature points discarded because they lie outside the domain
## of the covariate image

## 16,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 15 out of 2209 quadrature points discarded because they lie outside the domain
## of the covariate image

## 17,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 16 out of 2185 quadrature points discarded because they lie outside the domain
## of the covariate image

## 18,

## Warning: Values of the covariate 'Elevation' were NA or undefined at 0.05% (1
## out of 2190) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,

## Warning: Some infinite, NA or NaN increments were removed

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 16 out of 2190 quadrature points discarded because they lie outside the domain
## of the covariate image

## 19,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 15 out of 2175 quadrature points discarded because they lie outside the domain
## of the covariate image

## 20,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 17 out of 2186 quadrature points discarded because they lie outside the domain
## of the covariate image

## 21,
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 17 out of 2199 quadrature points discarded because they lie outside the domain
## of the covariate image

## 22,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 19 out of 2180 quadrature points discarded because they lie outside the domain
## of the covariate image

## 23,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 14 out of 2151 quadrature points discarded because they lie outside the domain
## of the covariate image

## 24,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 19 out of 2172 quadrature points discarded because they lie outside the domain
## of the covariate image

## 25,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 17 out of 2189 quadrature points discarded because they lie outside the domain
## of the covariate image

## 26,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 14 out of 2172 quadrature points discarded because they lie outside the domain
## of the covariate image

## 27,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 15 out of 2140 quadrature points discarded because they lie outside the domain
## of the covariate image

## 28,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 17 out of 2199 quadrature points discarded because they lie outside the domain
## of the covariate image

## 29,

## Warning: Values of the covariate 'Elevation' were NA or undefined at 0.05% (1
## out of 2158) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,
## Warning: Some infinite, NA or NaN increments were removed

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 13 out of 2158 quadrature points discarded because they lie outside the domain
## of the covariate image

## 30,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 13 out of 2169 quadrature points discarded because they lie outside the domain
## of the covariate image

## 31,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 15 out of 2189 quadrature points discarded because they lie outside the domain
## of the covariate image

## 32,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 20 out of 2174 quadrature points discarded because they lie outside the domain
## of the covariate image

## 33,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 17 out of 2185 quadrature points discarded because they lie outside the domain
## of the covariate image

## 34,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 17 out of 2205 quadrature points discarded because they lie outside the domain
## of the covariate image

## 35,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 16 out of 2139 quadrature points discarded because they lie outside the domain
## of the covariate image

## 36,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 15 out of 2182 quadrature points discarded because they lie outside the domain
## of the covariate image

## 37,
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 16 out of 2158 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 38,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 20 out of 2163 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
##
## 39.
## Done.
```

```r
lurk_HFI<-lurking (fit_gam, DATA$HFI ,type = "raw",cumulative = F,envelope = T,xlab = "Gradient", main=
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 77 out of 2159 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## Warning in bs(Elevation, degree = 3L, knots = numeric(0), Boundary.knots =
## c(-61.2767014431655, : some 'x' values beyond boundary knots may cause
## ill-conditioned bases
```

```
## Generating 39 simulated patterns ...1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 1
## 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38,
## 39.
## Processing..
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 77 out of 2217 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 1,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 78 out of 2176 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 2,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 76 out of 2172 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 3,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 80 out of 2152 quadrature points discarded because they lie outside the domain
## of the covariate image
## 4,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 80 out of 2181 quadrature points discarded because they lie outside the domain
## of the covariate image

## 5,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 76 out of 2194 quadrature points discarded because they lie outside the domain
## of the covariate image

## 6,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 75 out of 2170 quadrature points discarded because they lie outside the domain
## of the covariate image

## 7,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 79 out of 2165 quadrature points discarded because they lie outside the domain
## of the covariate image

## 8,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 77 out of 2162 quadrature points discarded because they lie outside the domain
## of the covariate image

## 9,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 94 out of 2202 quadrature points discarded because they lie outside the domain
## of the covariate image

## 10,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 70 out of 2164 quadrature points discarded because they lie outside the domain
## of the covariate image

## 11,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 76 out of 2191 quadrature points discarded because they lie outside the domain
## of the covariate image

## 12,
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 87 out of 2171 quadrature points discarded because they lie outside the domain
## of the covariate image

## 13,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 83 out of 2185 quadrature points discarded because they lie outside the domain
## of the covariate image

## 14,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 75 out of 2194 quadrature points discarded because they lie outside the domain
## of the covariate image

## 15,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 81 out of 2200 quadrature points discarded because they lie outside the domain
## of the covariate image

## 16,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 69 out of 2160 quadrature points discarded because they lie outside the domain
## of the covariate image

## 17,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 89 out of 2172 quadrature points discarded because they lie outside the domain
## of the covariate image

## 18,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 74 out of 2192 quadrature points discarded because they lie outside the domain
## of the covariate image

## 19,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 76 out of 2204 quadrature points discarded because they lie outside the domain
## of the covariate image

## 20,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 72 out of 2158 quadrature points discarded because they lie outside the domain
## of the covariate image
## 21,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 70 out of 2175 quadrature points discarded because they lie outside the domain
## of the covariate image

## 22,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 72 out of 2182 quadrature points discarded because they lie outside the domain
## of the covariate image

## 23,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 81 out of 2207 quadrature points discarded because they lie outside the domain
## of the covariate image

## 24,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 83 out of 2198 quadrature points discarded because they lie outside the domain
## of the covariate image

## 25,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 72 out of 2166 quadrature points discarded because they lie outside the domain
## of the covariate image

## 26,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 74 out of 2185 quadrature points discarded because they lie outside the domain
## of the covariate image

## 27,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 62 out of 2147 quadrature points discarded because they lie outside the domain
## of the covariate image

## 28,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 75 out of 2192 quadrature points discarded because they lie outside the domain
## of the covariate image

## 29,
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 73 out of 2164 quadrature points discarded because they lie outside the domain
## of the covariate image

## 30,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 67 out of 2166 quadrature points discarded because they lie outside the domain
## of the covariate image

## 31,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 69 out of 2194 quadrature points discarded because they lie outside the domain
## of the covariate image

## 32,

## Warning: Values of the covariate 'Elevation' were NA or undefined at 0.05% (1
## out of 2171) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,

## Warning: Some infinite, NA or NaN increments were removed

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 69 out of 2171 quadrature points discarded because they lie outside the domain
## of the covariate image

## 33,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 74 out of 2157 quadrature points discarded because they lie outside the domain
## of the covariate image

## 34,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 73 out of 2186 quadrature points discarded because they lie outside the domain
## of the covariate image

## 35,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 75 out of 2169 quadrature points discarded because they lie outside the domain
## of the covariate image

## 36,

## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 91 out of 2173 quadrature points discarded because they lie outside the domain
## of the covariate image
## 37,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 74 out of 2168 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
## 38,
```

```
## Warning in LurkEngine(object = object, type = type, cumulative = cumulative, :
## 74 out of 2165 quadrature points discarded because they lie outside the domain
## of the covariate image
```

```
##
## 39.
## Done.
```

```
lurk_water<-lurking (fit_gam, DATA$Dist_Water ,type = "raw",cumulative = F,envelope = T,xlab = "Gradien
```

```
## Warning in bs(Elevation, degree = 3L, knots = numeric(0), Boundary.knots =
## c(-61.2767014431655, : some ’x’ values beyond boundary knots may cause
## ill-conditioned bases
```

```
## Generating 39 simulated patterns ...1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 1
## 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38,
## 39.
## Processing.. 1,
```

```
## Warning: Values of the covariate ’Elevation’ were NA or undefined at 0.05% (1
## out of 2190) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,
```

```
## Warning: Some infinite, NA or NaN increments were removed
```

```
## 2, 3, 4, 5, 6, 7, 8, 9,
```

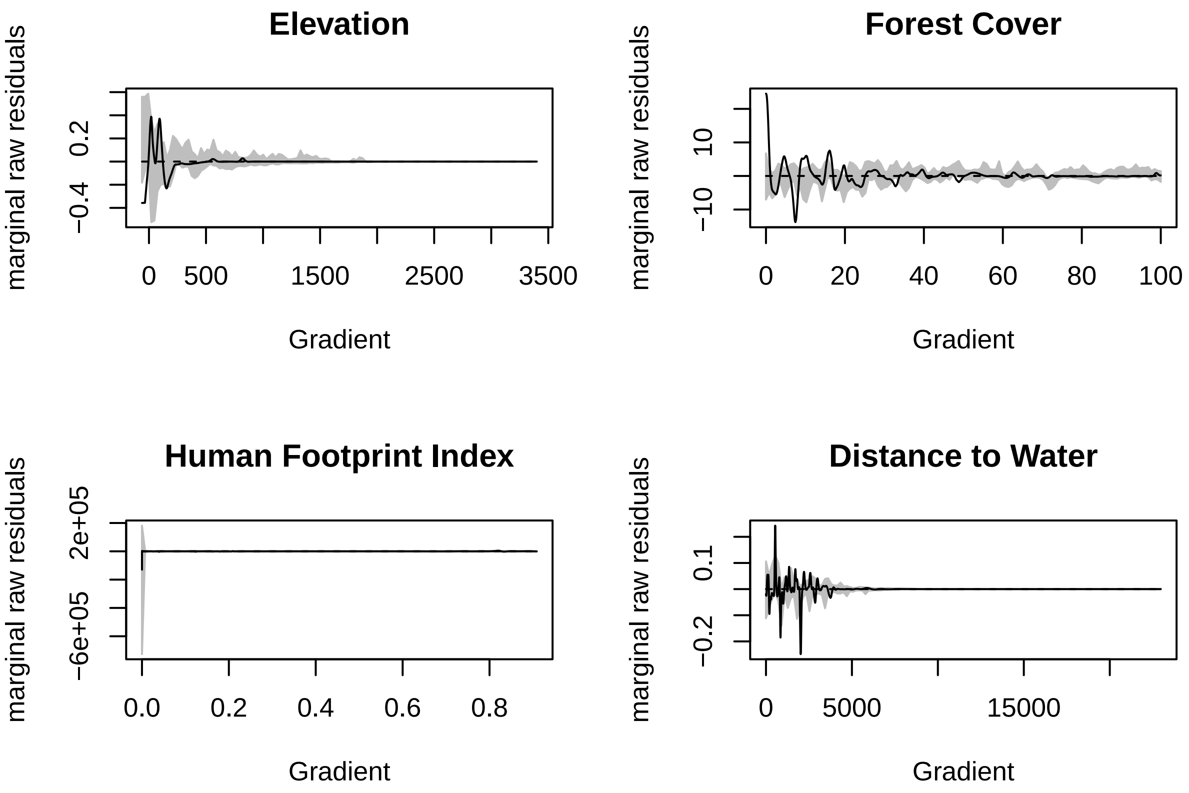
```
## Warning: Values of the covariate ’Elevation’ were NA or undefined at 0.05% (1
## out of 2175) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,
```

```
## Warning: Some infinite, NA or NaN increments were removed
```

```
## 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20,
## 21, 22, 23, 24, 25, 26, 27,
```

```
## Warning: Values of the covariate ’Elevation’ were NA or undefined at 0.05% (1
## out of 2160) of the quadrature points. Occurred while executing: ppm.quad(Q =
## list(list(list("polygonal", c(273874.897964858,
```

```
## Warning: Some infinite, NA or NaN increments were removed
## 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38,
## 39.
## Done.
```


Elevation
marginal raw residuals
Gradient
Forest Cover
marginal raw residuals
Gradient
Human Footprint Index
marginal raw residuals
2e+05
−6e+05
Gradient
Distance to Water
marginal raw residuals
Gradient