

589project

Rui Mao, Zheng Zhang, Zerui Zhang

2024-04-19

Load Package

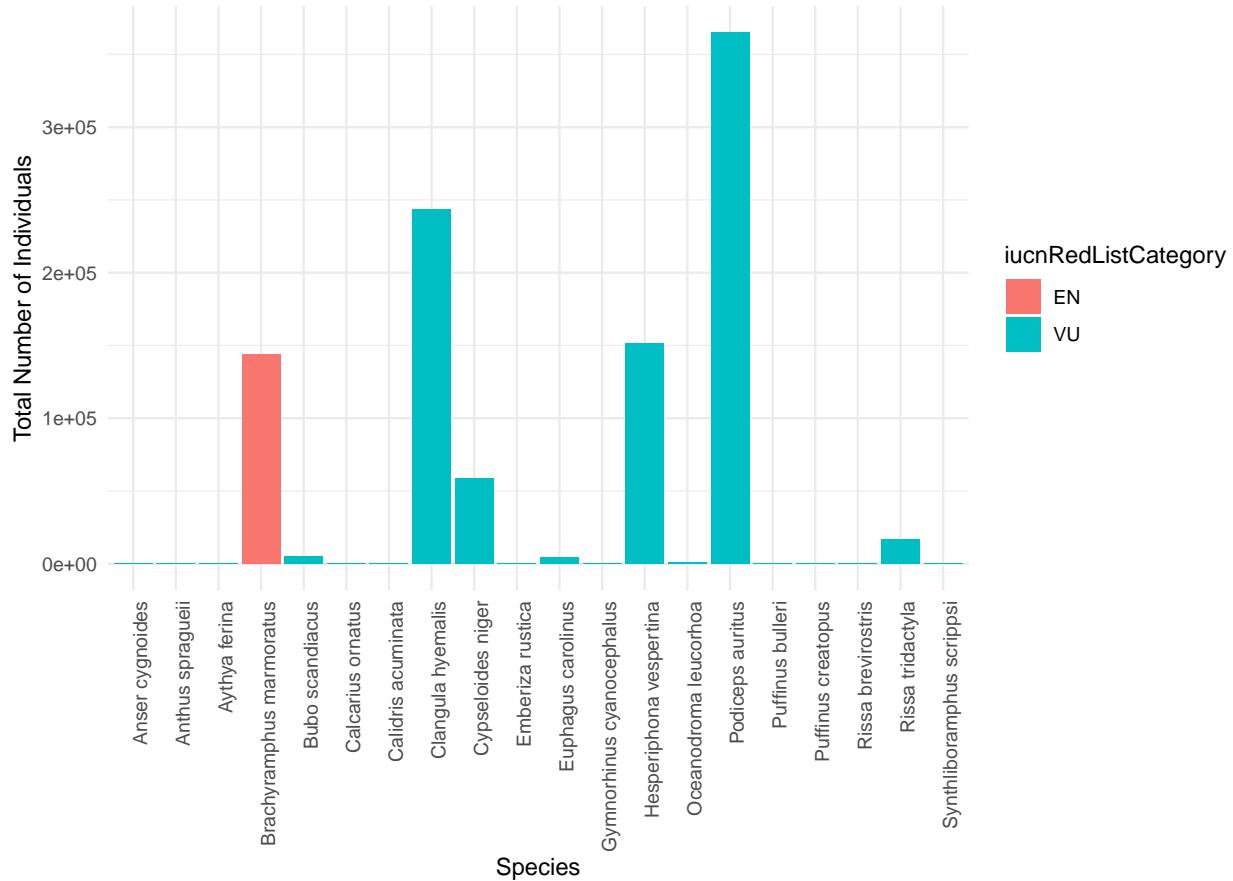
```
library(sf)
library(spatstat)
library(sp)
library(mapproj)
library(dplyr)
library(readr)
library(ggplot2)
library(lubridate)
```

Load Full Data

Species Distribution

```
ggplot(data = bird_bc, aes(x = species, weight = individualCount, fill = iucnRedListCategory)) +
  geom_bar() +
  labs(title = "Total Number of Individual Birds Observed by Species",
       x = "Species",
       y = "Total Number of Individuals") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

Total Number of Individual Birds Observed by Species



```
#rm(bird_bc)
```

Filter data

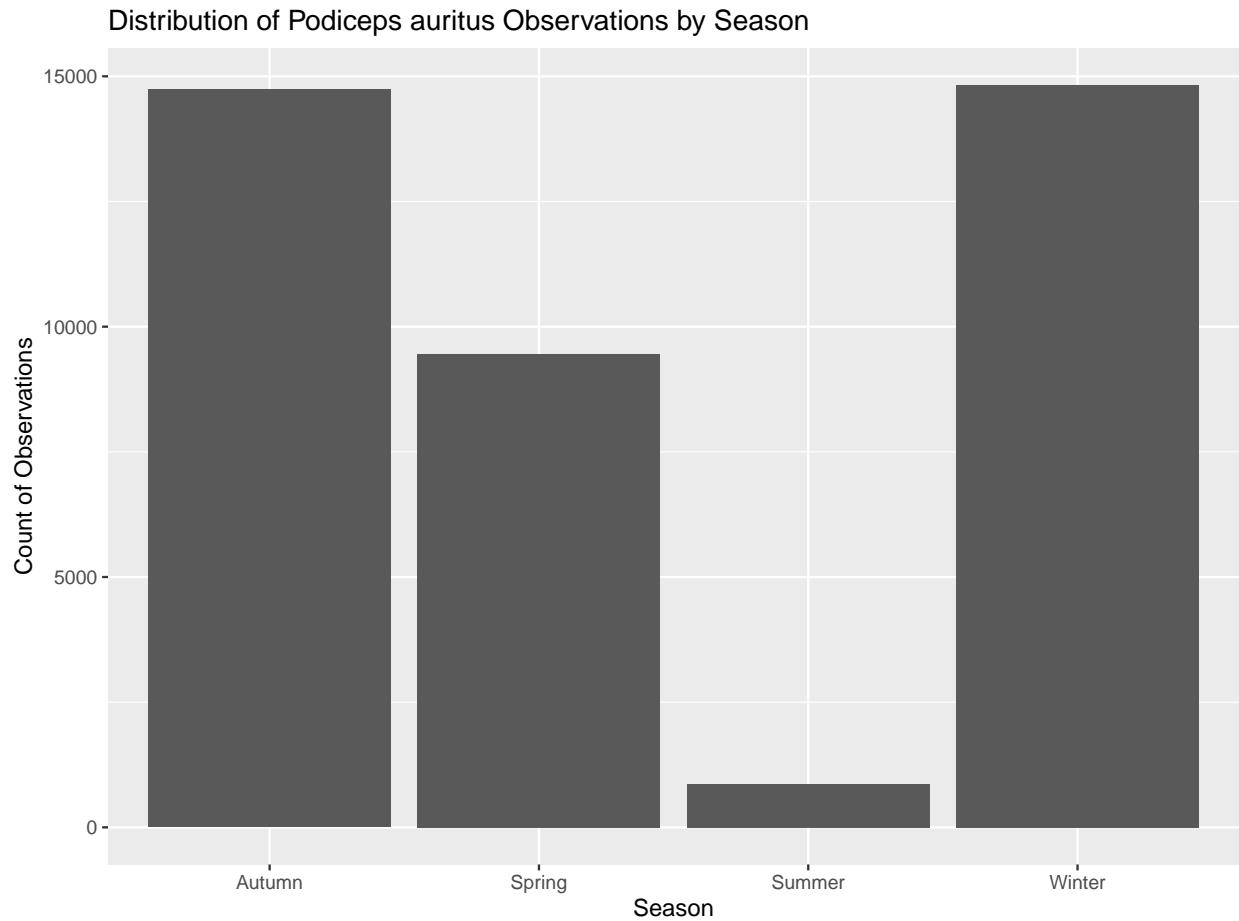
Load Data

```
bird_data <- read_csv('bird_data.csv')
load('BC_Covariates.Rda')
window = DATA$Window
```

Podiceps auritus Observations by Season

```
bird_data$eventDate <- as.Date(bird_data$eventDate, format = "%Y-%m-%d")
bird_data$season <- case_when(
  month(bird_data$eventDate) %in% 3:5 ~ "Spring",
  month(bird_data$eventDate) %in% 6:8 ~ "Summer",
  month(bird_data$eventDate) %in% 9:11 ~ "Autumn",
  month(bird_data$eventDate) %in% c(12, 1:2) ~ "Winter"
)
```

```
ggplot(bird_data, aes(x = season)) +
  geom_bar() +
  labs(title = "Distribution of Podiceps auritus Observations by Season",
       x = "Season",
       y = "Count of Observations")
```

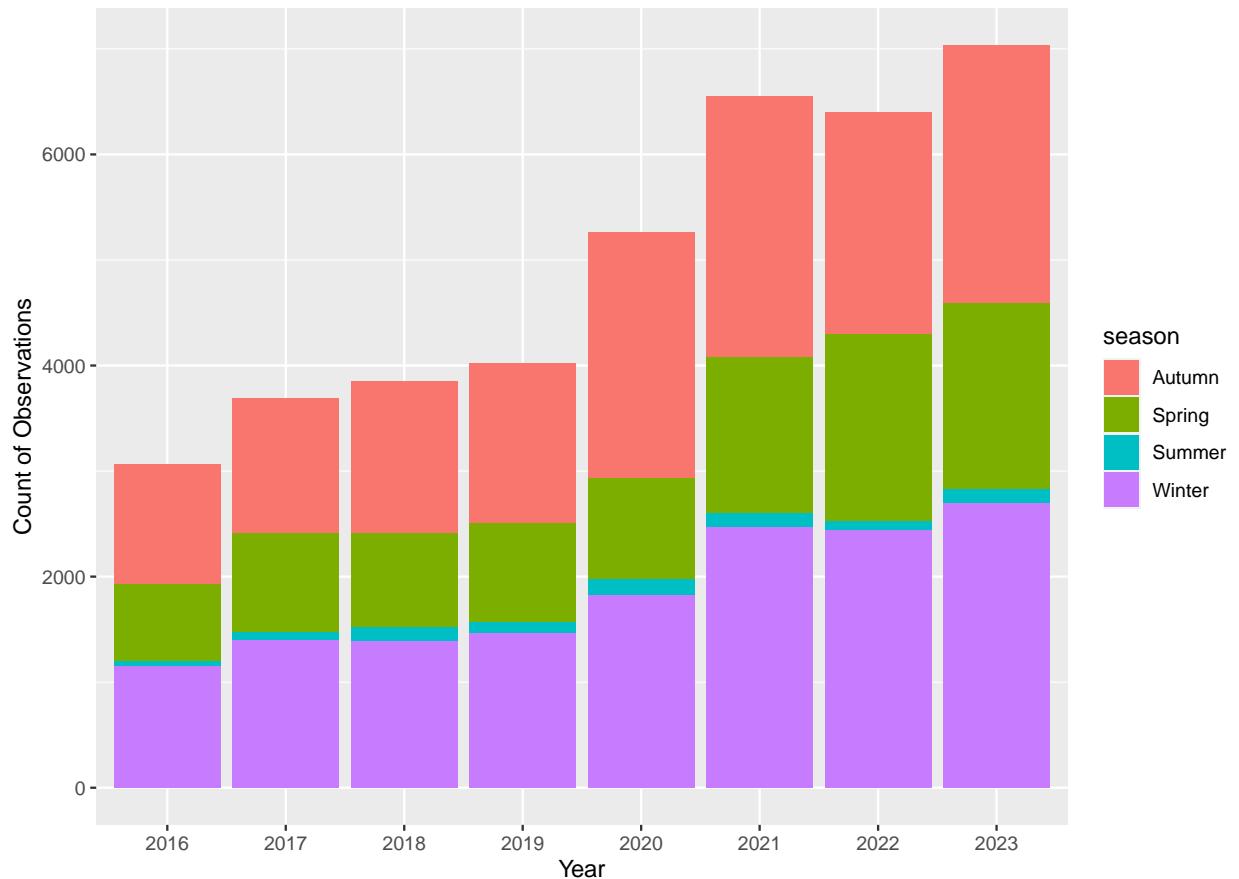


Podiceps auritus Observations by Year

```
bird_data$year <- as.integer(format(as.Date(bird_data$eventDate, "%Y-%m-%d"), "%Y"))

ggplot(bird_data, aes(x = factor(year))) +
  geom_bar(aes(fill = season)) +
  labs(title = "Distribution of Observations by Season and Year",
       x = "Year",
       y = "Count of Observations")
```

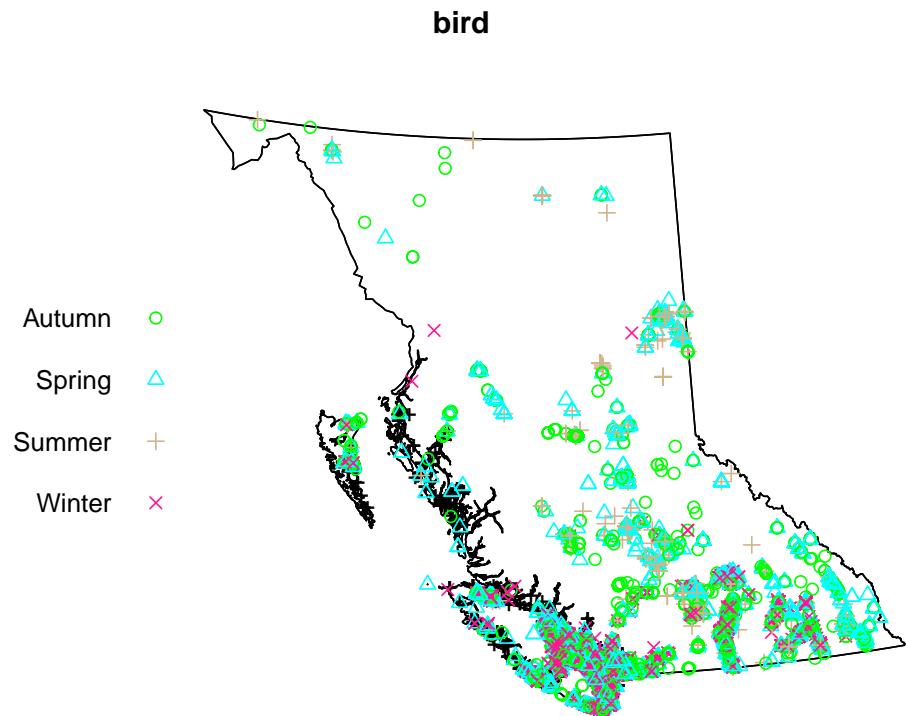
Distribution of Observations by Season and Year



ppp Object

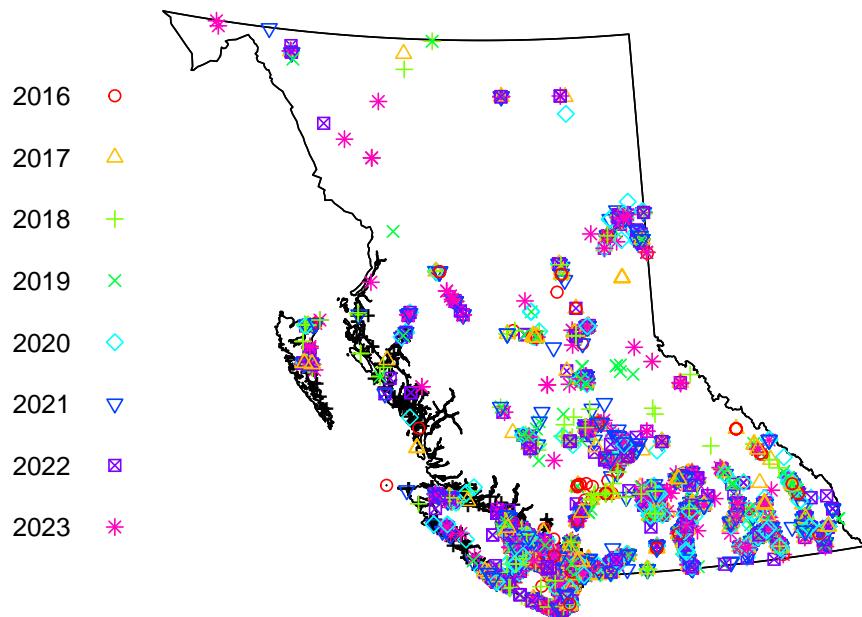
```
bird_sf <- st_as_sf(bird_data, coords = c("decimalLongitude", "decimalLatitude"), crs = 4326)
bc_albers_crs_string <- "+proj=aea +lat_0=45 +lon_0=-126 +lat_1=50 +lat_2=58.5 +x_0=1000000 +y_0=0 +dat"
bird_albers <- st_transform(bird_sf, crs = bc_albers_crs_string)

bird <- ppp(x = st_coordinates(bird_albers)[, 1],
            y = st_coordinates(bird_albers)[, 2],
            window = as.owin(window),
            marks = as.factor(bird_albers$season))
plot(bird, cols = c("green", "cyan", "tan", "deeppink"))
```



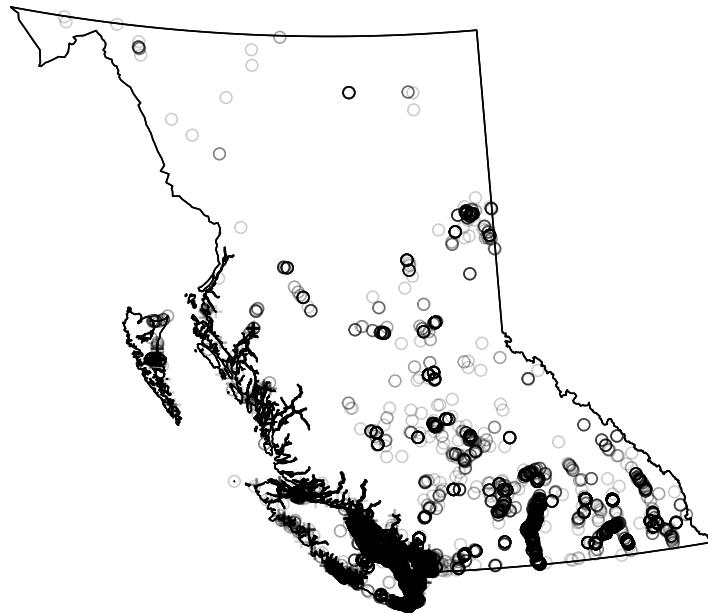
```
bird_year <- ppp(x = st_coordinates(bird_albers)[, 1],  
                  y = st_coordinates(bird_albers)[, 2],  
                  window = as.owin(window),  
                  marks = as.factor(bird_albers$year))  
plot(bird_year, cols = rainbow(length(unique(bird_albers$year))))
```

bird_year



```
bird_ppp <- ppp(x = st_coordinates(bird_albers)[, 1],  
                  y = st_coordinates(bird_albers)[, 2],  
                  window = as.owin(window))  
plot(bird_ppp)
```

bird_ppp



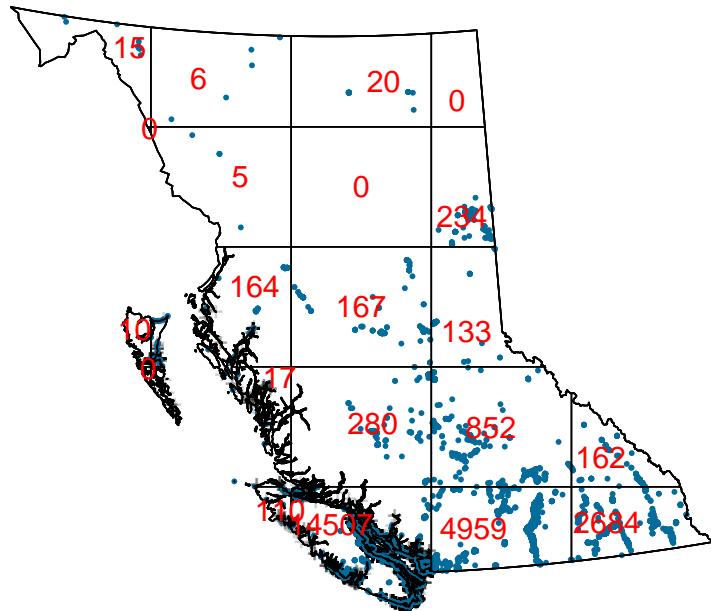
Spatially inhomogeneous λ

```
Q <- quadratcount(bird_ppp,
                     nx = 5,
                     ny = 5)

plot(bird_ppp,
      pch = 16,
      cex = 0.5,
      cols = "#046C9A")

plot(Q, cex = 1.2, col = "red", add = T)
```

bird_ppp



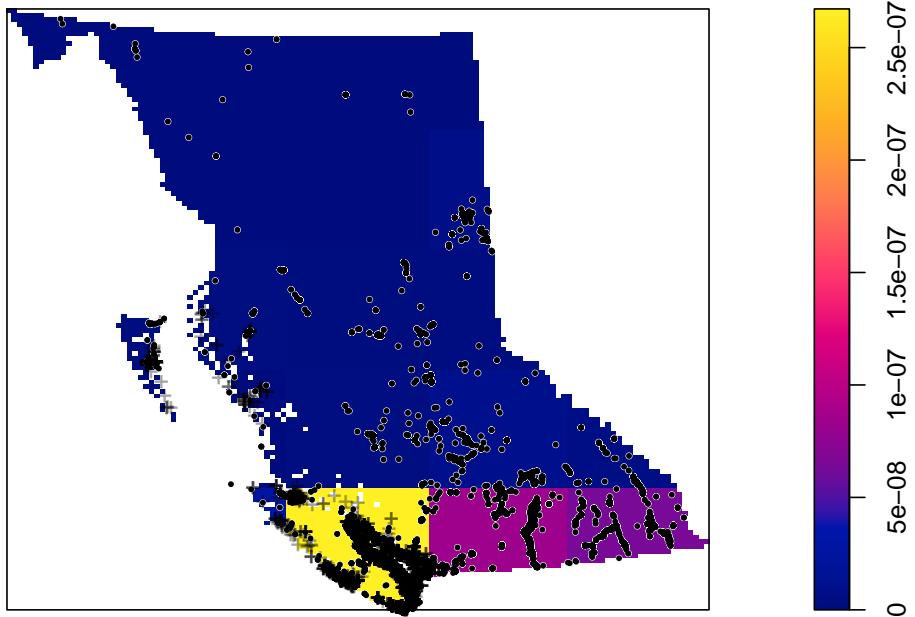
Intensity

```
plot(intensity(Q, image = T),
      main = "Birds intensity")

plot(bird_ppp,
      pch = 16,
      cex = 0.6,
      cols = "white",
      add = T)

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

Birds intensity



```
quadrat.test(Q)
```

```
##  
## Chi-squared test of CSR using quadrat counts  
##  
## data:  
## X2 = 150412, df = 20, p-value < 2.2e-16  
## alternative hypothesis: two.sided  
##  
## Quadrats: 21 tiles (irregular windows)
```

The small p-value suggests that there is a significant deviation from homogeneity.

Kernel estimation

```
#Density estimation of lambda(u)  
lambda_u_hat <- density(bird_ppp)  
  
#Plot the output Note the use of image = TRUE  
plot(lambda_u_hat,
```

```

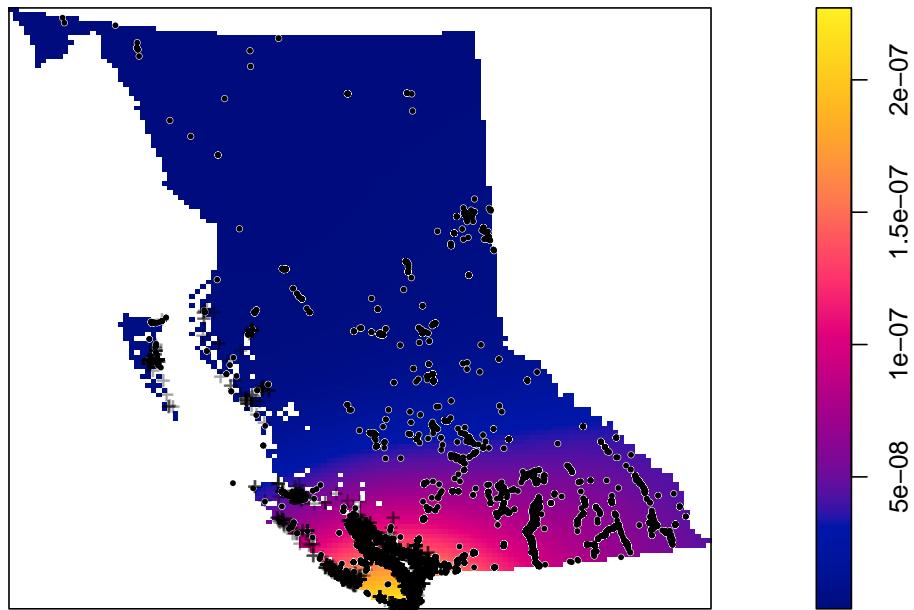
main = "Kernel estimate of Birds intensity")

plot(bird_ppp,
      pch = 16,
      cex = 0.6,
      cols = "white",
      add = T)

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

```

Kernel estimate of Birds intensity



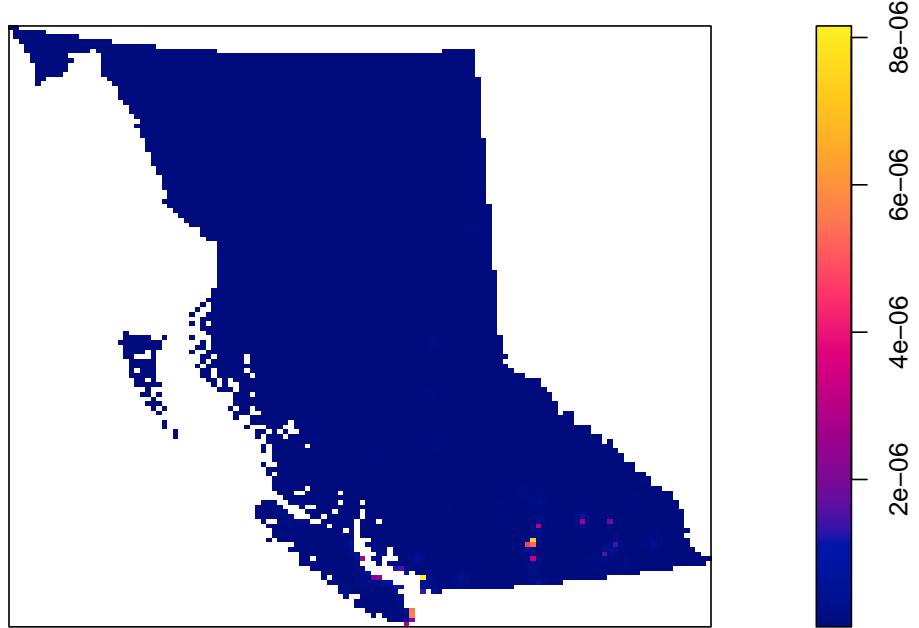
```

#Density estimation of lambda(u)
lambda_u_hat_adaptive <- adaptive.density(bird_ppp, method = "kernel")

#Plot the output Note the use of image = TRUE
plot(lambda_u_hat_adaptive,
      main = "Adaptive kernel estimate of intensity")

```

Adaptive kernel estimate of intensity

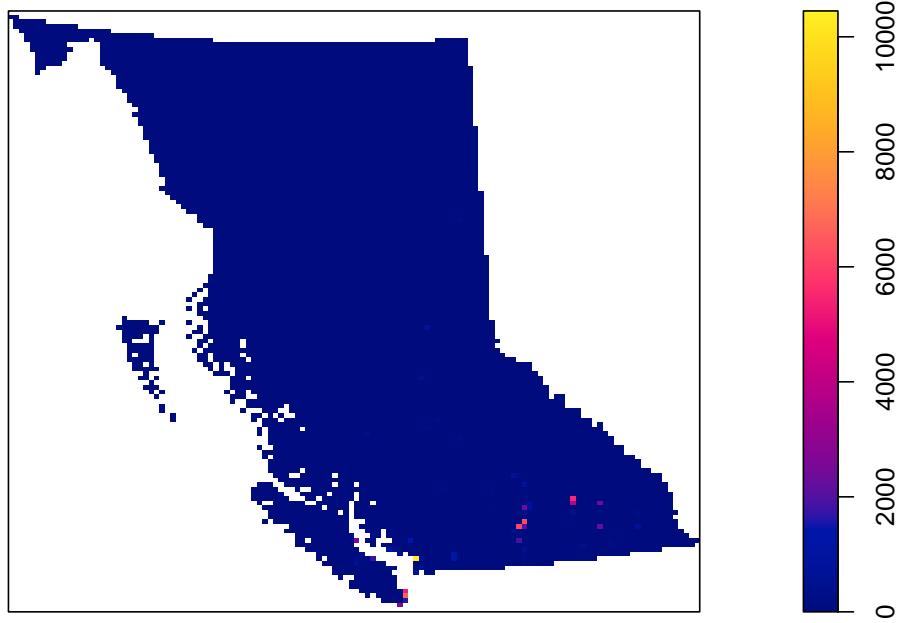


Hot spot analysis

```
# Estimate R
R <- bw.ppl(bird_ppp)
#Calculate test statistic
LR <- scanLRTS(bird_ppp, r = R)

plot(LR)
```

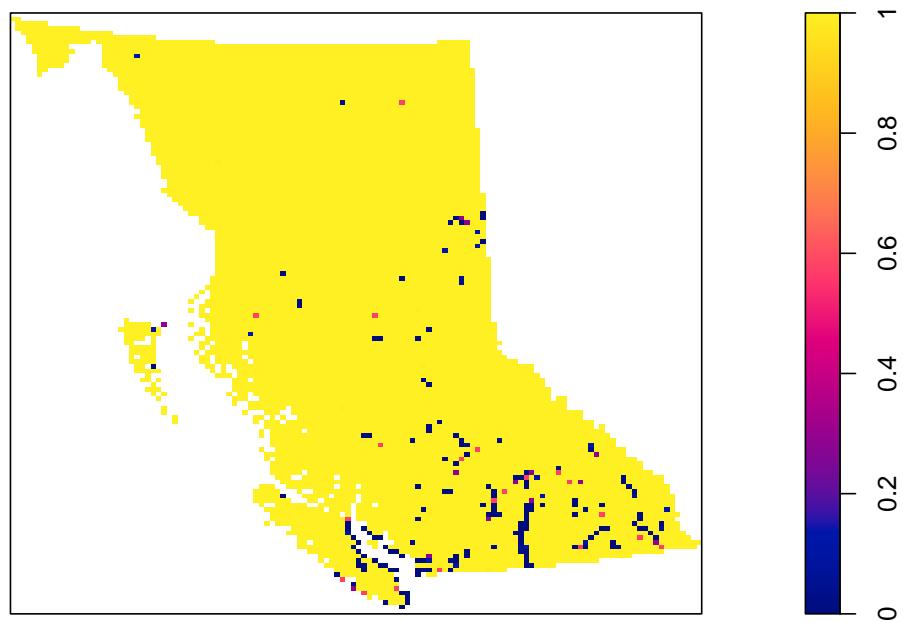
LR



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

plot(pvals, main = "Local p-values")
```

Local p-values

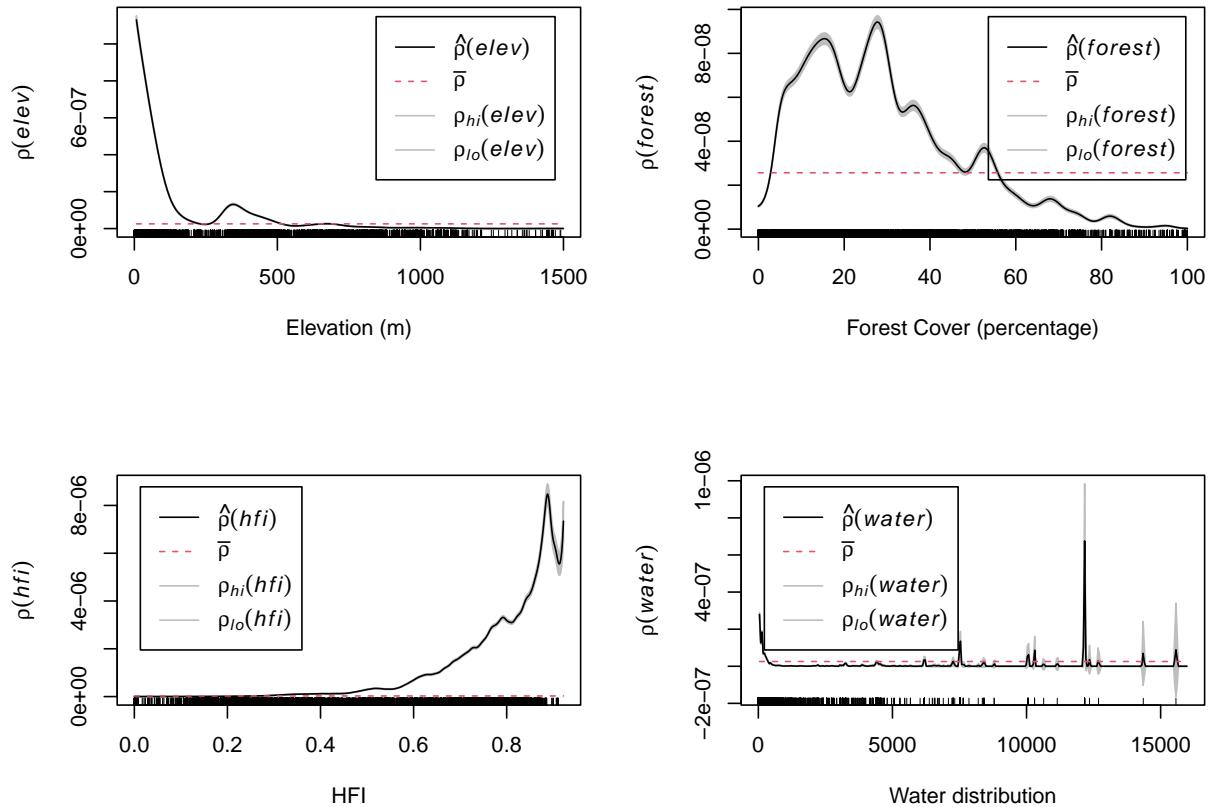


Relationships with covariates

```
elev <- DATA$Elevation
forest <- DATA$Forest
hfi <- DATA$HFI
water <- DATA$Dist_Water
rho_elev <- rhohat(bird_ppp, elev)
rho_forest <- rhohat(bird_ppp, forest)
rho_hfi <- rhohat(bird_ppp, hfi)
rho_water <- rhohat(bird_ppp, water)
```

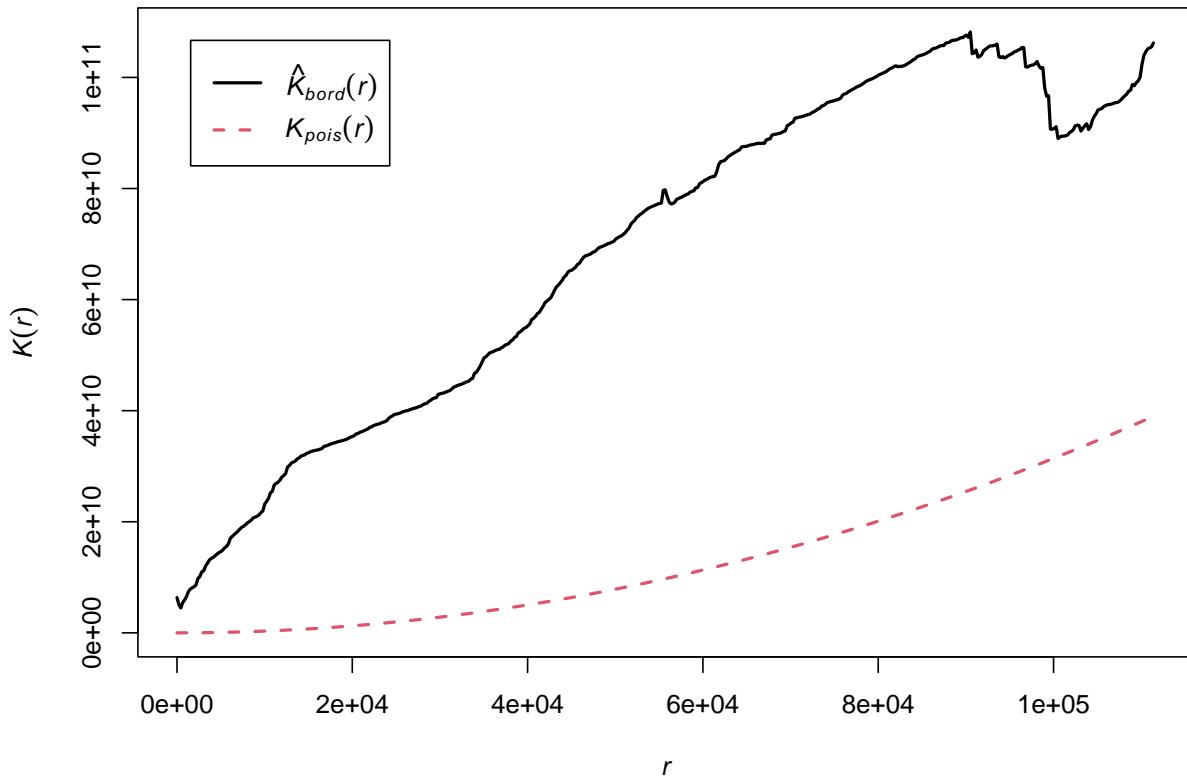
```
par(mfrow = c(2,2))
plot(rho_elev, xlim=c(0, 1500),
      main = "",
      xlab = "Elevation (m)")
plot(rho_forest,
      main = "",
      xlab = "Forest Cover (percentage)")
plot(rho_hfi,
      main = "",
      xlab = "HFI")
```

```
plot(rho_water, xlim=c(0, 16000),
      main = "",
      xlab = "Water distribution")
```



K-function

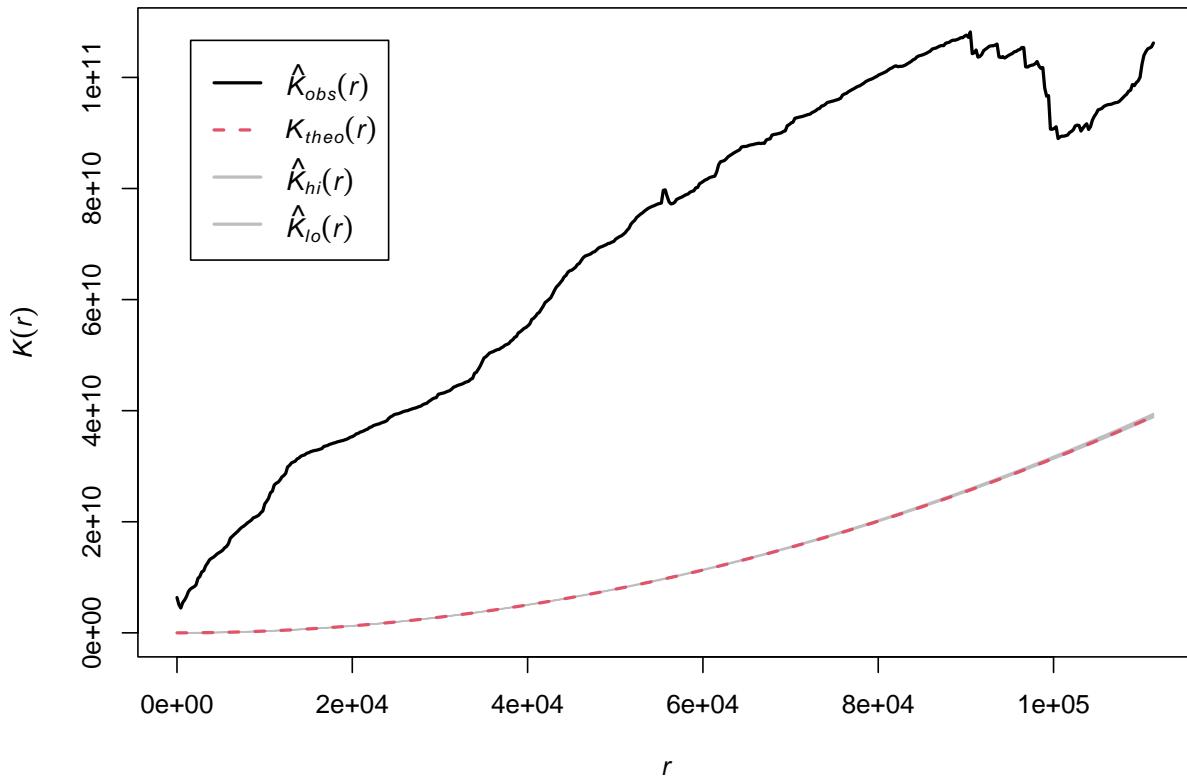
```
k_bird <- Kest(bird_ppp)
#visualise the results
plot(k_bird,
      main = "",
      lwd = 2)
```



```
E_bird <- envelope(bird_ppp,
                     Kest,
                     correction="border",
                     rank = 1,
                     nsim = 19,
                     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_bird,
     main = "",
     lwd = 2)
```



```

#Estimate intensity
lambda_bird <- density(bird_ppp, bw.ppl)

Kinhom_bird <- Kinhom(bird, lambda_bird)

Kinhom_bird

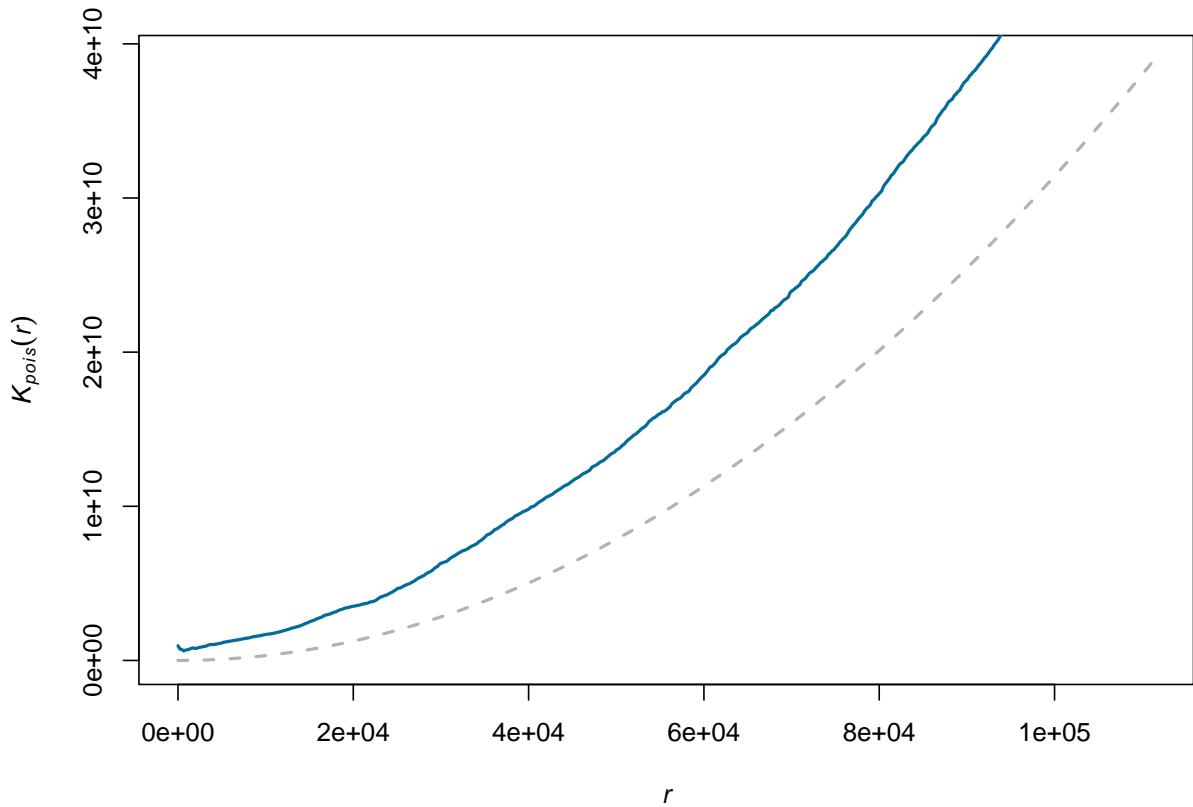
## Function value object (class 'fv')
## for the function r -> K[inhom](r)
## .....
##          Math.label
## r          r
## theo      K[pois](r)
## border    {hat(K)[inhom]^{bord}}(r)
## bord.modif {hat(K)[inhom]^{bordm}}(r)
##          Description
## r          distance argument r
## theo      theoretical Poisson K[inhom](r)
## border    border-corrected estimate of K[inhom](r)
## bord.modif modified border-corrected estimate of K[inhom](r)
## .....
## Default plot formula: .~r
## where "." stands for 'bord.modif', 'border', 'theo'

```

```
## Recommended range of argument r: [0, 111390]
## Available range of argument r: [0, 111390]
```

```
# visualise the results
plot(Kinhom_bird,
      theo ~ r,
      main = "",
      col = "grey70",
      lty = "dashed",
      lwd = 2)

plot(Kinhom_bird,
      border ~ r,
      col = c("#046C9A"),
      lwd = 2,
      add = T)
```



```
#Estimate a strictly positive density
lambda_bird_pos <- density(bird_pp,
                           sigma=bw.ppl,
                           positive=TRUE)

#Simulation envelope (with points drawn from the estimated intensity)
```

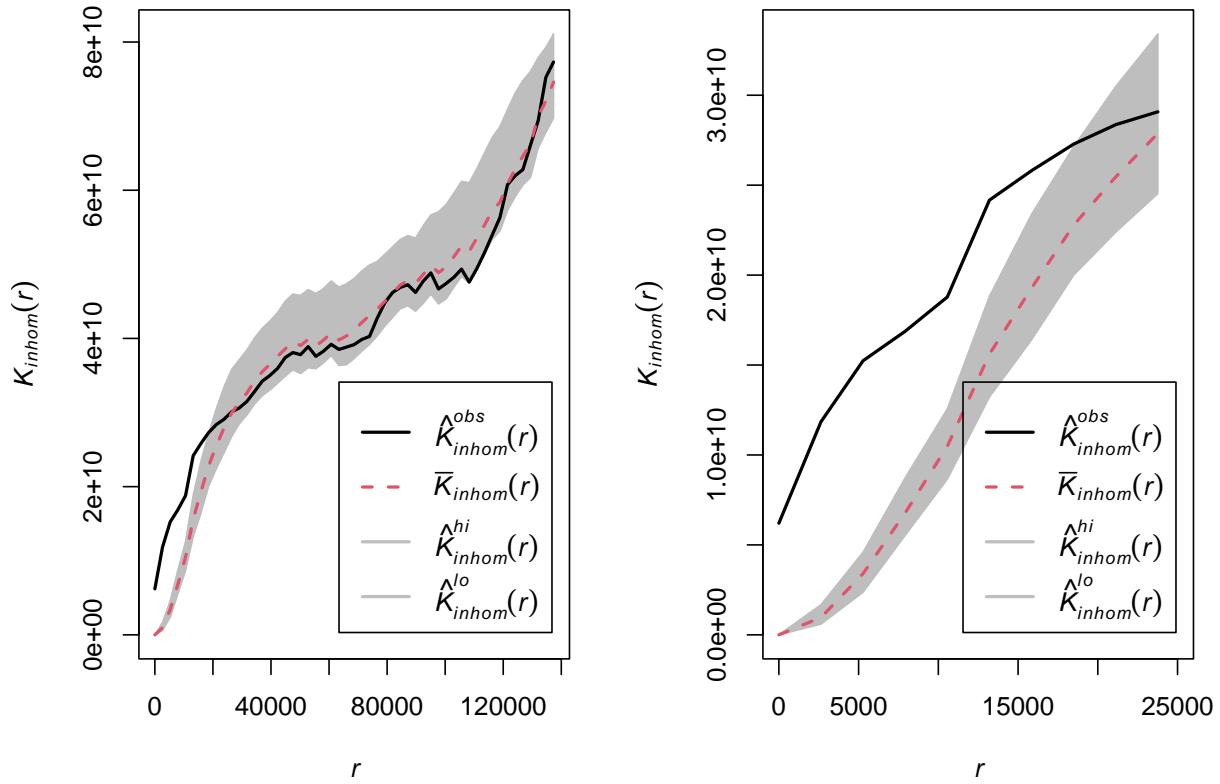
```

E_bird_inhom <- envelope(bird_ppp,
                           Kinhom,
                           simulate = expression(rpoispp(lambda_bird_pos)),
                           correction="border",
                           rank = 1,
                           nsim = 19,
                           fix.n = TRUE)

## Generating 19 simulations by evaluating expression ...
## 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,
## 19.
##
## Done.

# visualise the results
par(mfrow = c(1,2))
plot(E_bird_inhom,
     main = "",
     lwd = 2)
# Zoom in on range where significant deviations appear
plot(E_bird_inhom,
      xlim = c(0,25000),
      main = "",
      lwd = 2)

```



When corrected for inhomogeneity, significant clustering only appears to exist in and around 0-20000 units.

PCF

```
pcf_bird_hom <- envelope(bird_ppp,
                           pcf,
                           simulate = expression(rpoispp(density.ppp(bird_ppp))),
                           rank = 1,
                           nsim = 19)
```

```
## Generating 19 simulations by evaluating expression ...
## 1, 2, [3:04 remaining] 3,
## [3:05 remaining] 4, [3:06 remaining] 5, [2:57 remaining] 6,
## [2:47 remaining] 7, [2:33 remaining] 8, [2:21 remaining] 9,
## [2:10 remaining] 10, [1:57 remaining] 11, [1:45 remaining] 12,
## [1:31 remaining] 13, [1:18 remaining] 14, [1:06 remaining] 15,
## [52 sec remaining] 16, [39 sec remaining] 17, [26 sec remaining] 18,
## [13 sec remaining]
## 19.
##
## Done.
```

```

pcf_bird_inhom <- envelope(bird_ppp,
                            pcfinhom,
                             simulate = expression(rpoispp(density.ppp(bird_ppp))),
                             rank = 1,
                             nsim = 19)

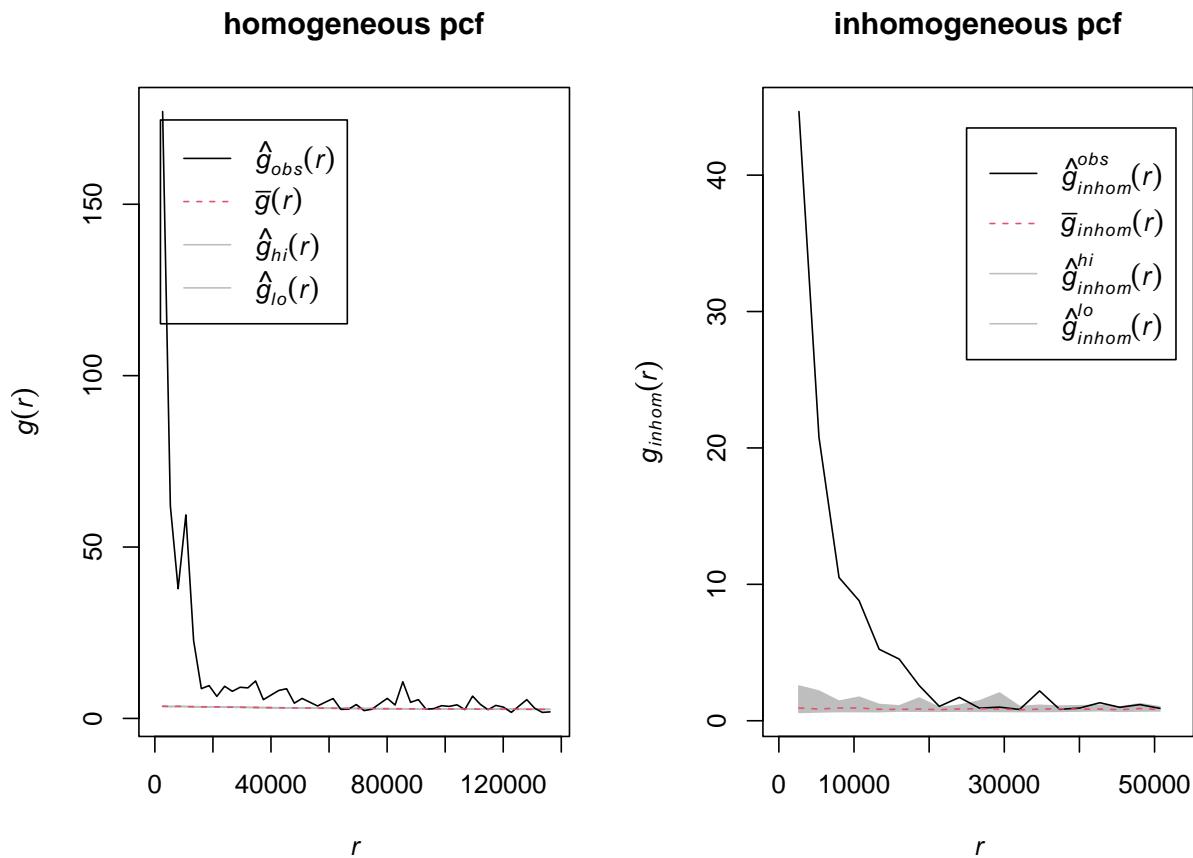
## Generating 19 simulations by evaluating expression ...
## 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,
## 19.
##
## Done.

par(mfrow = c(1,2))
plot(pcf_bird_hom, main = 'homogeneous pcf')

## Error in UseMethod("distmap") :
##   no applicable method for 'distmap' applied to an object of class "im"

plot(pcf_bird_inhom, main = 'inhomogeneous pcf')

```



Collinearity

```
cor.im(elev, forest, hfi, water, use = 'complete.obs')

##          .1          .2          .3          .4
## .1 1.00000000 -0.26225376 -0.26625626 -0.03493453
## .2 -0.26225376 1.00000000  0.06618592  0.04818598
## .3 -0.26625626  0.06618592 1.00000000  0.13246899
## .4 -0.03493453  0.04818598  0.13246899 1.00000000
```

Model fitting

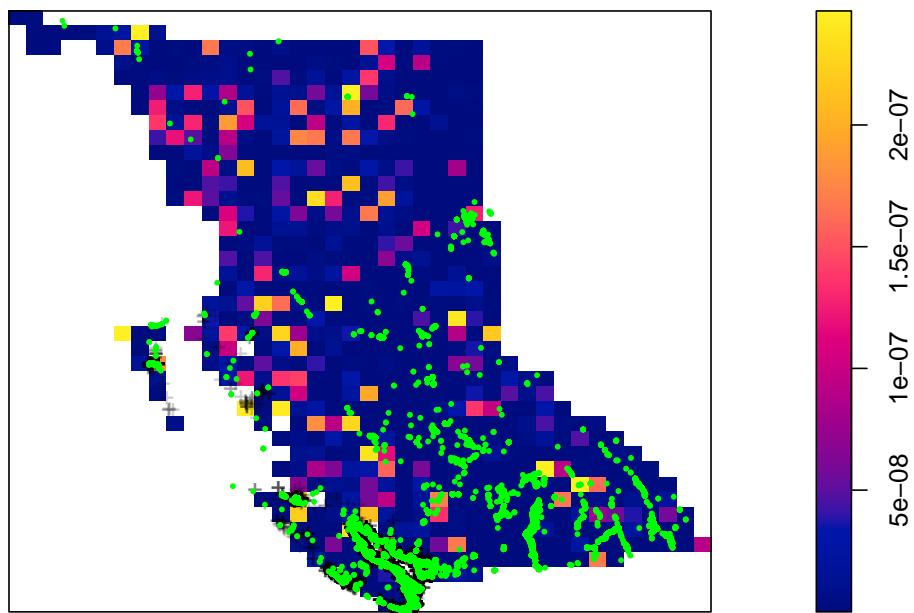
```
fit1 <- ppm(bird_ppp ~ forest + I(forest^2) + water)
fit1

## Nonstationary Poisson process
## Fitted to point pattern dataset 'bird_ppp'
##
## Log intensity: ~forest + I(forest^2) + water
##
## Fitted trend coefficients:
##   (Intercept)      forest    I(forest^2)      water
## -1.560362e+01  3.537356e-02 -7.973387e-04 -2.951399e-03
##
##             Estimate       S.E.     CI95.lo     CI95.hi Ztest
## (Intercept) -1.560362e+01 1.456965e-02 -1.563217e+01 -1.557506e+01 *** 
## forest       3.537356e-02 9.346370e-04  3.354170e-02  3.720541e-02 *** 
## I(forest^2) -7.973387e-04 1.343798e-05 -8.236767e-04 -7.710007e-04 *** 
## water        -2.951399e-03 2.305444e-05 -2.996585e-03 -2.906213e-03 *** 
##
##             Zval
## (Intercept) -1070.96714
## forest       37.84737
## I(forest^2)  -59.33472
## water        -128.01871
## Problem:
##   Values of the covariate 'forest' were NA or undefined at 0.09% (65 out of
## 68828) of the quadrature points

plot(fit1,
      se = FALSE,
      superimpose = FALSE)

plot(bird_ppp,
      pch = 16,
      cex = 0.5,
      cols = "green",
      add = TRUE)
```

Fitted trend

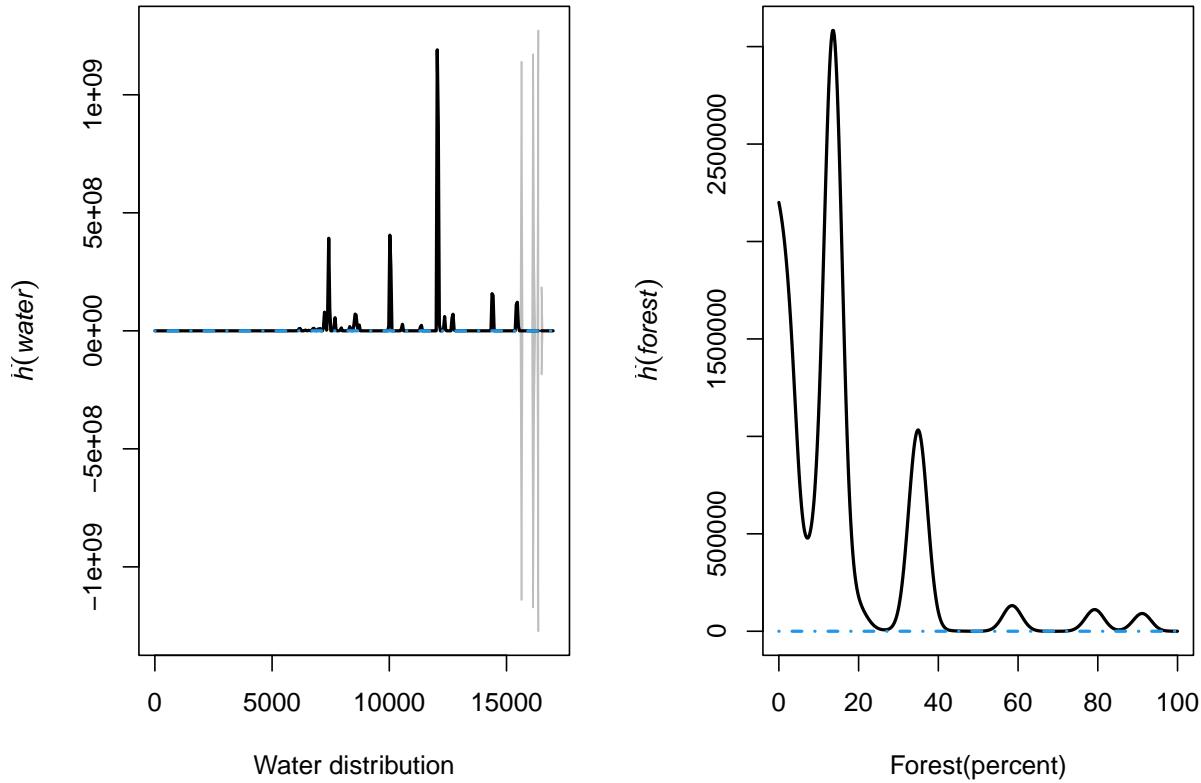


```
quadrat.test(fit1, nx = 4, ny = 2)
```

```
##  
## Chi-squared test of fitted Poisson model 'fit1' using quadrat counts  
##  
## data: data from fit1  
## X2 = 62147, df = 3, p-value < 2.2e-16  
## alternative hypothesis: two.sided  
##  
## Quadrats: 7 tiles (irregular windows)
```

```
par_res_water <- parres(fit1, "water")  
par_res_forest <- parres(fit1, "forest")  
  
par(mfrow = c(1,2))  
plot(par_res_water,  
     legend = FALSE,  
     lwd = 2,  
     main = "",  
     xlab = "Water distribution")  
plot(par_res_forest,  
     legend = FALSE,  
     lwd = 2,
```

```
main = "",  
xlab = "Forest(percent)")
```



```
fit2 <- ppm(bird_ppp, ~ water)  
fit2
```

```
## Nonstationary Poisson process  
## Fitted to point pattern dataset 'bird_ppp'  
##  
## Log intensity: ~water  
##  
## Fitted trend coefficients:  
## (Intercept) water  
## -15.7229186 -0.0033746  
##  
## Estimate S.E. CI95.lo CI95.hi Ztest  
## (Intercept) -15.7229186 7.893681e-03 -15.738389902 -15.707447242 ***  
## water -0.0033746 2.425101e-05 -0.003422131 -0.003327069 ***  
## Zval  
## (Intercept) -1991.836  
## water -139.153
```

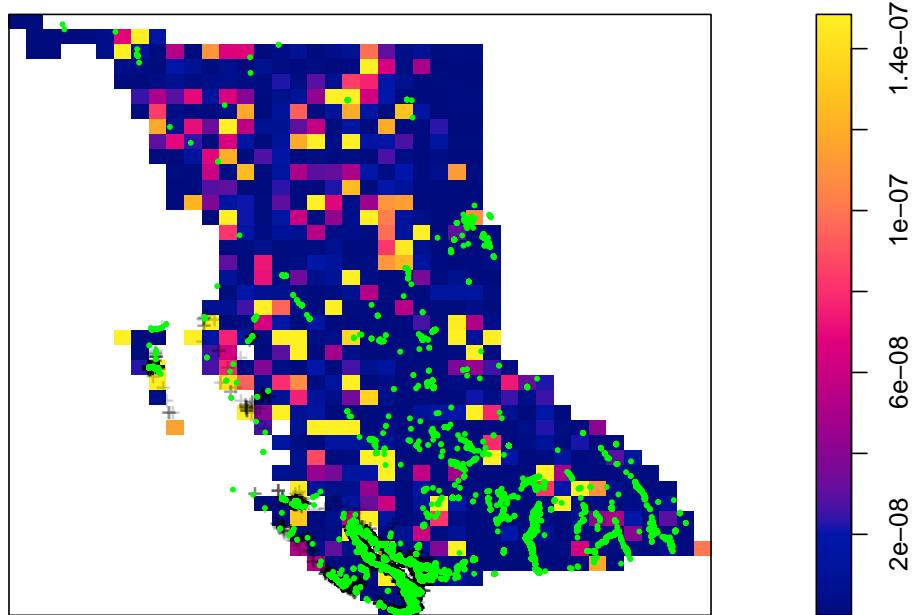
```

plot(fit2,
      se = FALSE,
      superimpose = FALSE)

plot(bird_ppp,
      pch = 16,
      cex = 0.5,
      cols = "green",
      add = TRUE)

```

Fitted trend

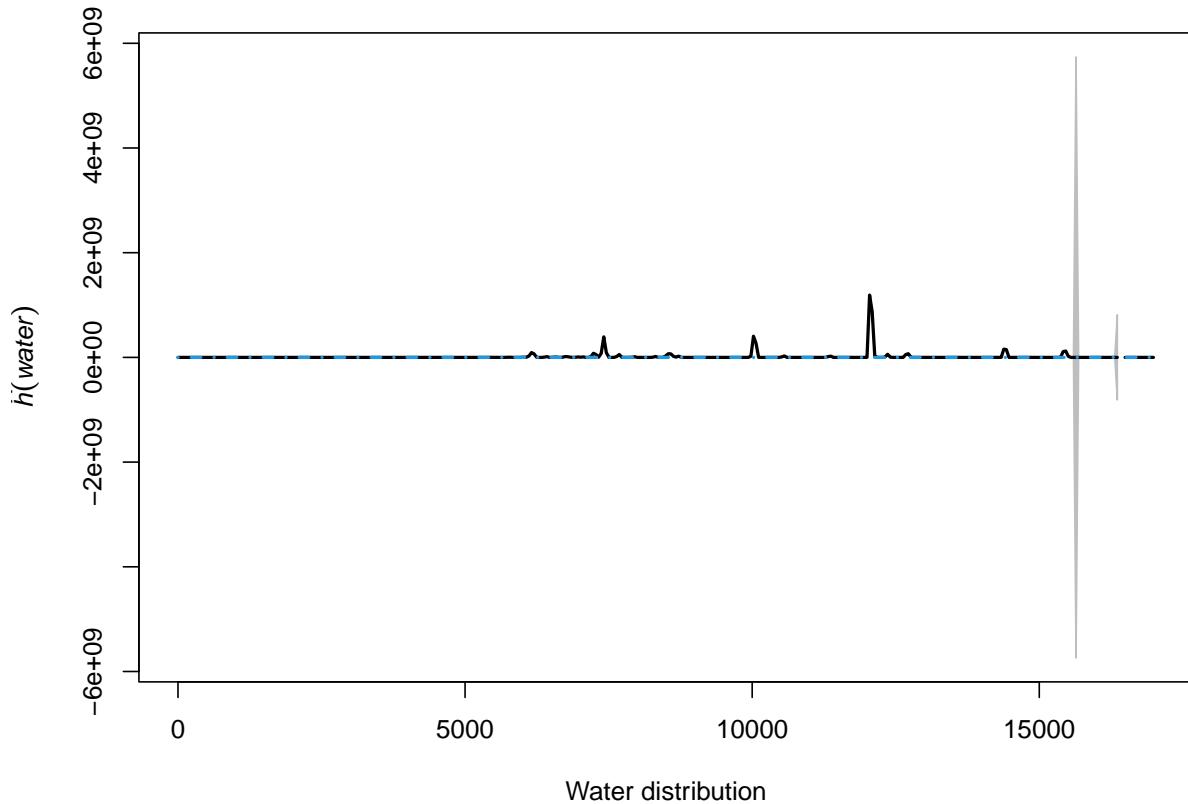


```

par_res_water <- parres(fit2, "water")

plot(par_res_water,
      legend = FALSE,
      lwd = 2,
      main = "",
      xlab = "Water distribution")

```



```
fit3 <- ppm(bird_ppp, ~ forest + I(forest^2) + elev + hfi)
fit3
```

```
## Nonstationary Poisson process
## Fitted to point pattern dataset 'bird_ppp'
##
## Log intensity: ~forest + I(forest^2) + elev + hfi
##
## Fitted trend coefficients:
##   (Intercept)      forest     I(forest^2)       elev        hfi
## -1.736741e+01  3.538133e-02 -5.367716e-04 -2.920751e-03  6.043144e+00
##
##             Estimate       S.E.    CI95.lo    CI95.hi Ztest
## (Intercept) -1.736741e+01 2.988399e-02 -1.742598e+01 -1.730884e+01 *** 
## forest       3.538133e-02 8.753478e-04  3.366568e-02  3.709698e-02 *** 
## I(forest^2) -5.367716e-04 1.139484e-05 -5.591051e-04 -5.144381e-04 *** 
## elev        -2.920751e-03 2.600330e-05 -2.971717e-03 -2.869786e-03 *** 
## hfi         6.043144e+00 3.275634e-02  5.978942e+00  6.107345e+00 *** 
##
##            Zval
## (Intercept) -581.16118
## forest       40.41974
## I(forest^2) -47.10654
## elev        -112.32235
```

```

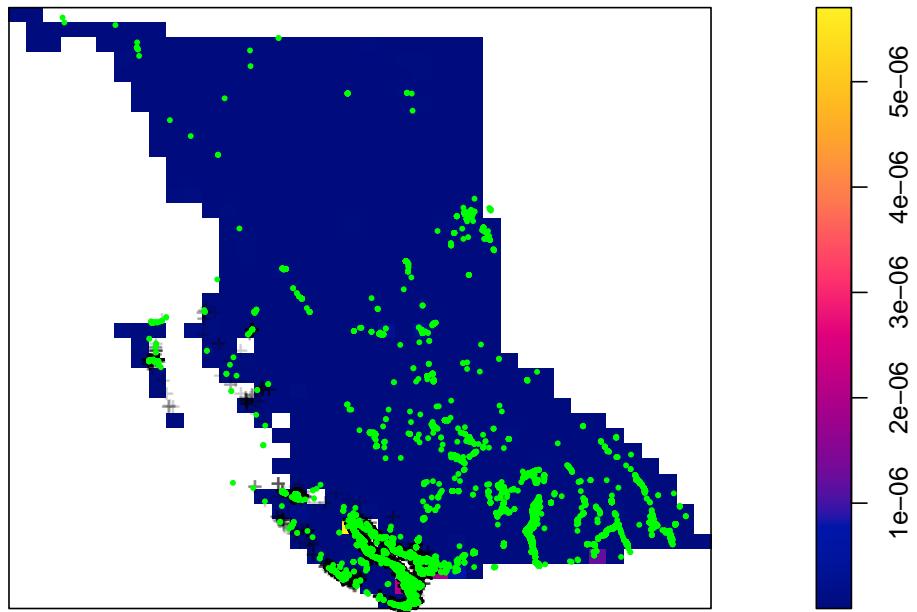
## hfi          184.48777
## Problem:
##   Values of the covariates 'forest', 'elev', 'hfi' were NA or undefined at 0.77%
## (528 out of 68828) of the quadrature points

plot(fit3,
      se = FALSE,
      superimpose = FALSE)

plot(bird_ppp,
      pch = 16,
      cex = 0.5,
      cols = "green",
      add = TRUE)

```

Fitted trend



```
print(AIC(fit1))
```

```
## [1] 828613.2
```

```
print(AIC(fit2))
```

```
## [1] 842327.1
```

```

print(AIC(fit3))

## [1] 719736

print(anova(fit1, fit2, fit3, test="LRT"))

## Analysis of Deviance Table
##
## Model 1: ~forest + I(forest^2) + water    Poisson
## Model 2: ~water    Poisson
## Model 3: ~forest + I(forest^2) + elev + hfi    Poisson
##   Npar Df Deviance  Pr(>Chi)
## 1  467
## 2  465 -2    -11680 < 2.2e-16 ***
## 3  468  3    109661 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

fit4 <- ppm(bird_ppp, ~ forest + I(forest^2) + elev + hfi + water)
fit4

## Nonstationary Poisson process
## Fitted to point pattern dataset 'bird_ppp'
##
## Log intensity: ~forest + I(forest^2) + elev + hfi + water
##
## Fitted trend coefficients:
##   (Intercept)      forest  I(forest^2)       elev        hfi
## -1.609201e+01  7.201047e-03 -2.323025e-04 -2.575927e-03  5.806896e+00
##   water
## -1.503770e-03
##
##           Estimate       S.E.     CI95.lo     CI95.hi Ztest
## (Intercept) -1.609201e+01 2.912923e-02 -1.614911e+01 -1.603492e+01 *** 
## forest       7.201047e-03 9.181537e-04  5.401499e-03  9.000595e-03 *** 
## I(forest^2) -2.323025e-04 1.194929e-05 -2.557227e-04 -2.088823e-04 *** 
## elev         -2.575927e-03 2.444987e-05 -2.623848e-03 -2.528006e-03 *** 
## hfi          5.806896e+00 3.111368e-02  5.745914e+00  5.867877e+00 *** 
## water        -1.503770e-03 1.687063e-05 -1.536836e-03 -1.470704e-03 *** 
##
##           Zval
## (Intercept) -552.435237
## forest       7.842965
## I(forest^2) -19.440691
## elev        -105.355429
## hfi          186.634792
## water        -89.135380
## Problem:
## Values of the covariates 'forest', 'elev', 'hfi' were NA or undefined at 0.77%
## (528 out of 68828) of the quadrature points

```

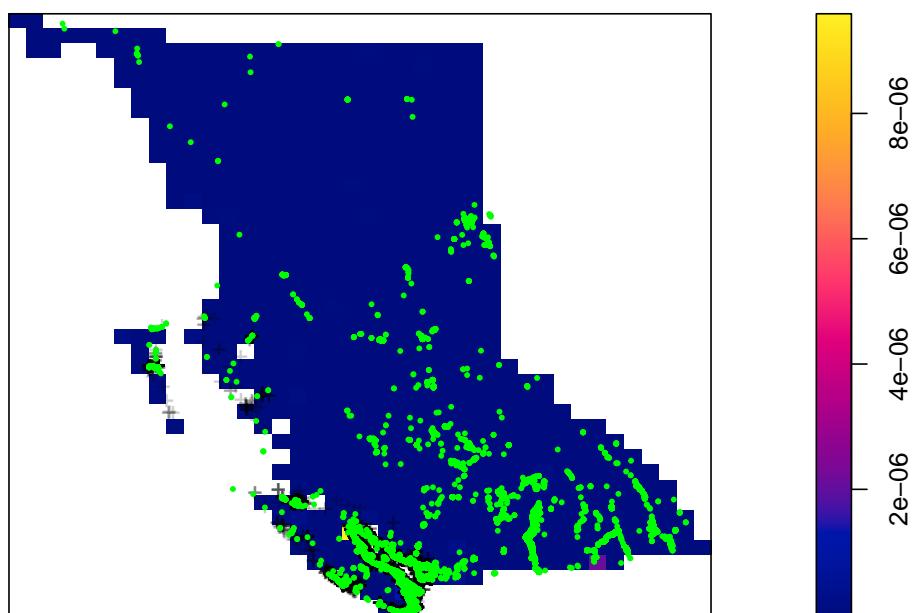
```

plot(fit4,
      se = FALSE,
      superimpose = FALSE)

plot(bird_ppp,
      pch = 16,
      cex = 0.5,
      cols = "green",
      add = TRUE)

```

Fitted trend



```

AIC(fit3); AIC(fit4)

## [1] 719736

## [1] 695016.8

anova(fit3, fit4, test = "LRT")

## Analysis of Deviance Table
##
## Model 1: ~forest + I(forest^2) + elev + hfi    Poisson

```

```

## Model 2: ~forest + I(forest^2) + elev + hfi + water    Poisson
##   Npar Df Deviance  Pr(>Chi)
## 1     5
## 2     6  1    24721 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

fit5 <- ppm(bird_ppp ~ forest + I(forest^2) + elev + hfi + water + I(water^2))
fit5

## Error in solve.default(M) :
##   system is computationally singular: reciprocal condition number = 7.91152e-17
## Error in solve.default(M) :
##   system is computationally singular: reciprocal condition number = 7.91152e-17
## Nonstationary Poisson process
## Fitted to point pattern dataset 'bird_ppp'
##
## Log intensity: ~forest + I(forest^2) + elev + hfi + water + I(water^2)
##
## Fitted trend coefficients:
##   (Intercept)      forest    I(forest^2)       elev        hfi
## -1.602669e+01  6.151803e-03 -2.163719e-04 -2.550862e-03  5.778963e+00
##   water      I(water^2)
## -1.835384e-03  8.241579e-08
##
## Standard errors unavailable; Fisher information matrix is singular
## Problem:
##   Values of the covariates 'forest', 'elev', 'hfi' were NA or undefined at 0.77%
## (528 out of 68828) of the quadrature points

```

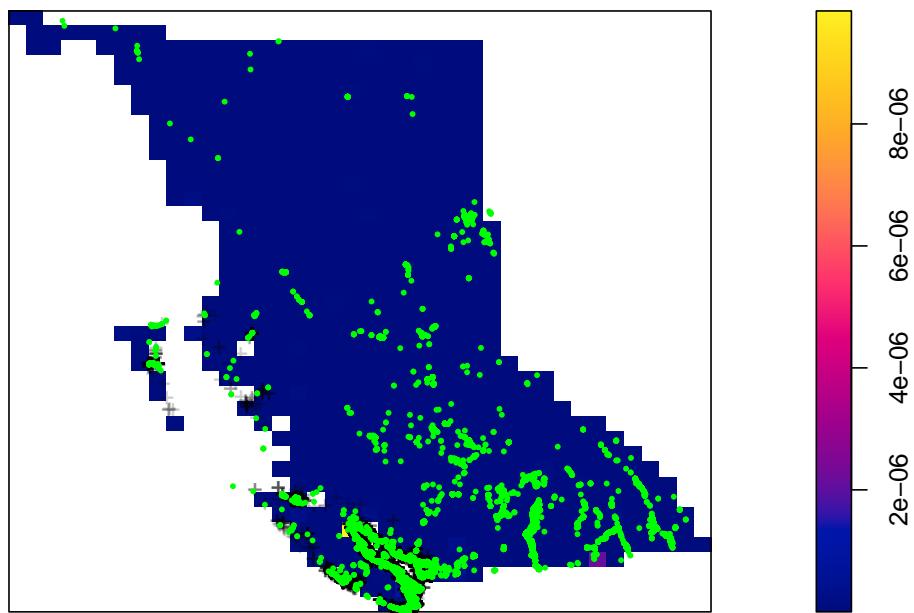
```

plot(fit5,
      se = FALSE,
      superimpose = FALSE)

plot(bird_ppp,
      pch = 16,
      cex = 0.5,
      cols = "green",
      add = TRUE)

```

Fitted trend



```
AIC(fit4); AIC(fit5)

## [1] 695016.8

## [1] 693035.7

anova(fit4, fit5, test = "LRT")

## Analysis of Deviance Table
##
## Model 1: ~forest + I(forest^2) + elev + hfi + water    Poisson
## Model 2: ~forest + I(forest^2) + elev + hfi + water + I(water^2)      Poisson
##   Npar Df Deviance  Pr(>Chi)
## 1     6
## 2     7  1  1983.1 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

fit6 <- ppm(bird_ppp ~ forest + I(forest^2) + elev + I(elev^2) + hfi + water + I(water^2))
fit6
```

```

## Error in solve.default(M) :
##   system is computationally singular: reciprocal condition number = 8.11593e-17
## Error in solve.default(M) :
##   system is computationally singular: reciprocal condition number = 8.11593e-17
## Nonstationary Poisson process
## Fitted to point pattern dataset 'bird_ppp'
##
## Log intensity: ~forest + I(forest^2) + elev + I(elev^2) + hfi + water +
## I(water^2)
##
## Fitted trend coefficients:
##   (Intercept)      forest    I(forest^2)      elev    I(elev^2)
## -1.608612e+01  5.776975e-03 -2.143370e-04 -1.706166e-03 -9.083512e-07
##   hfi        water    I(water^2)
##  5.770464e+00 -1.827034e-03  8.191155e-08
##
## Standard errors unavailable; Fisher information matrix is singular
## Problem:
## Values of the covariates 'forest', 'elev', 'hfi' were NA or undefined at 0.77%
## (528 out of 68828) of the quadrature points

```

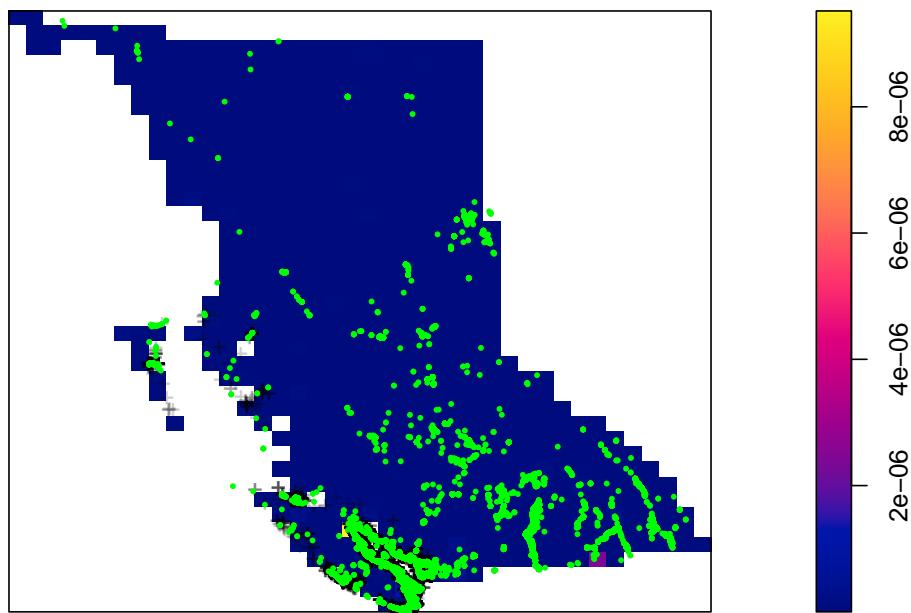
```

plot(fit6,
      se = FALSE,
      superimpose = FALSE)

plot(bird_ppp,
      pch = 16,
      cex = 0.5,
      cols = "green",
      add = TRUE)

```

Fitted trend



```

AIC(fit4); AIC(fit5)

## [1] 695016.8

## [1] 693035.7

anova(fit5, fit6, test = "LRT")

## Analysis of Deviance Table
##
## Model 1: ~forest + I(forest^2) + elev + hfi + water + I(water^2)      Poisson
## Model 2: ~forest + I(forest^2) + elev + I(elev^2) + hfi + water + I(water^2)  Poisson
##   Npar Df Deviance  Pr(>Chi)
## 1     7
## 2     8  1  277.76 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# Fit a point process model with selected covariates, considering potential interactions
fit7 <- ppm(bird_ppp ~ hfi + I(hfi^2) + forest + I(forest^2) + water + I(water^2)+ elev + I(elev^2))

# Check the summary of the model
summary(fit7)

```

```

## Error in solve.default(M) :
##   system is computationally singular: reciprocal condition number = 2.83187e-18
## Error in solve.default(M) :
##   system is computationally singular: reciprocal condition number = 2.83187e-18

## Point process model
## Fitted to data: bird_ppp
## Fitting method: maximum likelihood (Berman-Turner approximation)
## Model was fitted using glm()
## Algorithm converged
## Call:
## ppm.formula(Q = bird_ppp ~ hfi + I(hfi^2) + forest + I(forest^2) +
##   water + I(water^2) + elev + I(elev^2))
## Edge correction: "border"
## [border correction distance r = 0 ]
##
## -----
## Quadrature scheme (Berman-Turner) = data + dummy + weights
##
## Data pattern:
## Planar point pattern: 24325 points
## Average intensity 2.57e-08 points per square unit
## Window: polygonal boundary
## 379 separate polygons (2 holes)
##           vertices      area relative.area
## polygon 1          5454 889865000000  9.38e-01
## polygon 2          2129 319023000000  3.36e-02
## polygon 3 (hole)    55  -49871200  -5.26e-05
## polygon 4 (hole)    21  -15073400  -1.59e-05
## polygon 5          751  6476610000  6.83e-03
## polygon 6          764  2633530000  2.78e-03
## polygon 7          379  2286840000  2.41e-03
## polygon 8          281  1397560000  1.47e-03
## polygon 9          155  1020880000  1.08e-03
## polygon 10         178  822976000  8.68e-04
## polygon 11         189  531172000  5.60e-04
## polygon 12         135  523600000  5.52e-04
## polygon 13          93  429077000  4.52e-04
## polygon 14          59  375017000  3.95e-04
## polygon 15          80  366555000  3.87e-04
## polygon 16          130 389457000  4.11e-04
## polygon 17          129 361761000  3.81e-04
## polygon 18          73  325642000  3.43e-04
## polygon 19          72  284295000  3.00e-04
## polygon 20          98  291606000  3.08e-04
## polygon 21          87  301693000  3.18e-04
## polygon 22          60  271131000  2.86e-04
## polygon 23          105 273020000  2.88e-04
## polygon 24           67  238816000  2.52e-04
## polygon 25           72  240998000  2.54e-04
## polygon 26           51  207196000  2.19e-04
## polygon 27           81  175770000  1.85e-04
## polygon 28           72  186213000  1.96e-04
## polygon 29           60  170593000  1.80e-04
## polygon 30           52  171879000  1.81e-04

```

## polygon 31	62	174668000	1.84e-04
## polygon 32	55	163246000	1.72e-04
## polygon 33	63	151745000	1.60e-04
## polygon 34	62	164385000	1.73e-04
## polygon 35	70	155140000	1.64e-04
## polygon 36	63	152920000	1.61e-04
## polygon 37	72	134301000	1.42e-04
## polygon 38	88	135277000	1.43e-04
## polygon 39	93	128722000	1.36e-04
## polygon 40	45	127343000	1.34e-04
## polygon 41	84	130472000	1.38e-04
## polygon 42	49	114370000	1.21e-04
## polygon 43	44	106635000	1.12e-04
## polygon 44	53	102951000	1.09e-04
## polygon 45	52	100309000	1.06e-04
## polygon 46	45	95770200	1.01e-04
## polygon 47	43	95377600	1.01e-04
## polygon 48	39	92824200	9.79e-05
## polygon 49	47	81225900	8.57e-05
## polygon 50	40	80617200	8.50e-05
## polygon 51	64	83219400	8.78e-05
## polygon 52	33	76223000	8.04e-05
## polygon 53	34	67360500	7.10e-05
## polygon 54	54	68347400	7.21e-05
## polygon 55	44	64136400	6.76e-05
## polygon 56	42	66265500	6.99e-05
## polygon 57	29	62988400	6.64e-05
## polygon 58	31	59626500	6.29e-05
## polygon 59	25	54751800	5.77e-05
## polygon 60	31	55352900	5.84e-05
## polygon 61	30	53162300	5.61e-05
## polygon 62	44	57621600	6.08e-05
## polygon 63	26	56707500	5.98e-05
## polygon 64	36	56979200	6.01e-05
## polygon 65	32	55472500	5.85e-05
## polygon 66	21	50455600	5.32e-05
## polygon 67	40	51039400	5.38e-05
## polygon 68	26	53043500	5.59e-05
## polygon 69	29	51938100	5.48e-05
## polygon 70	27	50295800	5.30e-05
## polygon 71	33	47938300	5.06e-05
## polygon 72	18	44300200	4.67e-05
## polygon 73	25	45691800	4.82e-05
## polygon 74	29	43313200	4.57e-05
## polygon 75	36	46561100	4.91e-05
## polygon 76	31	46220100	4.87e-05
## polygon 77	26	40401700	4.26e-05
## polygon 78	30	43204000	4.56e-05
## polygon 79	26	41430100	4.37e-05
## polygon 80	25	36988000	3.90e-05
## polygon 81	28	37700100	3.98e-05
## polygon 82	20	35892800	3.79e-05
## polygon 83	34	38217700	4.03e-05
## polygon 84	21	34874700	3.68e-05

## polygon 85	25	33536100	3.54e-05
## polygon 86	30	36738900	3.87e-05
## polygon 87	30	35393400	3.73e-05
## polygon 88	23	32140100	3.39e-05
## polygon 89	28	31469400	3.32e-05
## polygon 90	23	32736900	3.45e-05
## polygon 91	28	34117200	3.60e-05
## polygon 92	20	31802600	3.35e-05
## polygon 93	26	30659000	3.23e-05
## polygon 94	24	29572700	3.12e-05
## polygon 95	27	27289800	2.88e-05
## polygon 96	16	23094400	2.44e-05
## polygon 97	23	24218600	2.55e-05
## polygon 98	18	21846600	2.30e-05
## polygon 99	14	23702300	2.50e-05
## polygon 100	21	24325800	2.57e-05
## polygon 101	23	22590100	2.38e-05
## polygon 102	24	19177800	2.02e-05
## polygon 103	22	18657800	1.97e-05
## polygon 104	15	17454800	1.84e-05
## polygon 105	14	16276800	1.72e-05
## polygon 106	13	15792300	1.67e-05
## polygon 107	17	16117900	1.70e-05
## polygon 108	12	16217100	1.71e-05
## polygon 109	19	16610700	1.75e-05
## polygon 110	18	16809400	1.77e-05
## polygon 111	15	15096400	1.59e-05
## polygon 112	14	14571400	1.54e-05
## polygon 113	11	13567300	1.43e-05
## polygon 114	16	13623600	1.44e-05
## polygon 115	15	13657400	1.44e-05
## polygon 116	18	14346200	1.51e-05
## polygon 117	15	12418500	1.31e-05
## polygon 118	22	13085700	1.38e-05
## polygon 119	11	13446800	1.42e-05
## polygon 120	14	13234700	1.40e-05
## polygon 121	15	13532800	1.43e-05
## polygon 122	16	12575800	1.33e-05
## polygon 123	15	13060700	1.38e-05
## polygon 124	15	11766100	1.24e-05
## polygon 125	18	11038100	1.16e-05
## polygon 126	12	12863200	1.36e-05
## polygon 127	19	11135700	1.17e-05
## polygon 128	11	10881600	1.15e-05
## polygon 129	14	12500500	1.32e-05
## polygon 130	12	10733900	1.13e-05
## polygon 131	15	11201200	1.18e-05
## polygon 132	13	10674100	1.13e-05
## polygon 133	13	9776920	1.03e-05
## polygon 134	14	10503500	1.11e-05
## polygon 135	12	9391460	9.90e-06
## polygon 136	10	9402110	9.92e-06
## polygon 137	9	9160880	9.66e-06
## polygon 138	11	8674110	9.15e-06

## polygon 139	12	9159420	9.66e-06
## polygon 140	14	8676590	9.15e-06
## polygon 141	12	9527540	1.00e-05
## polygon 142	12	8797870	9.28e-06
## polygon 143	11	8837670	9.32e-06
## polygon 144	13	9067690	9.56e-06
## polygon 145	12	7950440	8.38e-06
## polygon 146	9	8727650	9.20e-06
## polygon 147	8	8565660	9.03e-06
## polygon 148	8	6790020	7.16e-06
## polygon 149	11	6780640	7.15e-06
## polygon 150	11	7326980	7.73e-06
## polygon 151	10	6152880	6.49e-06
## polygon 152	13	6721370	7.09e-06
## polygon 153	8	6343870	6.69e-06
## polygon 154	11	6807170	7.18e-06
## polygon 155	10	6927150	7.31e-06
## polygon 156	9	6352310	6.70e-06
## polygon 157	13	6217600	6.56e-06
## polygon 158	15	6263440	6.61e-06
## polygon 159	11	5769240	6.08e-06
## polygon 160	8	5965270	6.29e-06
## polygon 161	10	6826030	7.20e-06
## polygon 162	10	5749960	6.06e-06
## polygon 163	15	5352810	5.64e-06
## polygon 164	9	5781470	6.10e-06
## polygon 165	13	5358790	5.65e-06
## polygon 166	10	6016930	6.35e-06
## polygon 167	9	5928820	6.25e-06
## polygon 168	8	5424110	5.72e-06
## polygon 169	9	5468680	5.77e-06
## polygon 170	11	5209670	5.49e-06
## polygon 171	9	4705250	4.96e-06
## polygon 172	9	4891810	5.16e-06
## polygon 173	11	5295110	5.58e-06
## polygon 174	7	4839440	5.10e-06
## polygon 175	7	4322930	4.56e-06
## polygon 176	7	4514530	4.76e-06
## polygon 177	7	4573450	4.82e-06
## polygon 178	9	5080950	5.36e-06
## polygon 179	12	4492120	4.74e-06
## polygon 180	8	4783060	5.04e-06
## polygon 181	8	3800770	4.01e-06
## polygon 182	5	4657790	4.91e-06
## polygon 183	11	4323150	4.56e-06
## polygon 184	9	3549270	3.74e-06
## polygon 185	6	4103470	4.33e-06
## polygon 186	8	3898980	4.11e-06
## polygon 187	8	4009440	4.23e-06
## polygon 188	9	3672080	3.87e-06
## polygon 189	11	3540120	3.73e-06
## polygon 190	6	3790670	4.00e-06
## polygon 191	8	3695580	3.90e-06
## polygon 192	7	3708650	3.91e-06

## polygon 193	7	3843670	4.05e-06
## polygon 194	7	3157020	3.33e-06
## polygon 195	9	2913680	3.07e-06
## polygon 196	9	3368170	3.55e-06
## polygon 197	7	3190600	3.36e-06
## polygon 198	6	3073660	3.24e-06
## polygon 199	9	3016490	3.18e-06
## polygon 200	8	3129780	3.30e-06
## polygon 201	6	3101410	3.27e-06
## polygon 202	7	2895440	3.05e-06
## polygon 203	8	2998870	3.16e-06
## polygon 204	7	2619570	2.76e-06
## polygon 205	10	3007320	3.17e-06
## polygon 206	7	2324090	2.45e-06
## polygon 207	10	2935450	3.10e-06
## polygon 208	10	2523640	2.66e-06
## polygon 209	6	2232050	2.35e-06
## polygon 210	6	2131190	2.25e-06
## polygon 211	6	1679950	1.77e-06
## polygon 212	6	2226470	2.35e-06
## polygon 213	8	2267800	2.39e-06
## polygon 214	6	2584690	2.73e-06
## polygon 215	5	2151150	2.27e-06
## polygon 216	6	1536960	1.62e-06
## polygon 217	6	2616700	2.76e-06
## polygon 218	7	2205520	2.33e-06
## polygon 219	8	2008210	2.12e-06
## polygon 220	5	2181540	2.30e-06
## polygon 221	7	2700450	2.85e-06
## polygon 222	7	2493440	2.63e-06
## polygon 223	7	2752380	2.90e-06
## polygon 224	5	2036350	2.15e-06
## polygon 225	5	2344480	2.47e-06
## polygon 226	5	1786340	1.88e-06
## polygon 227	8	2269610	2.39e-06
## polygon 228	6	1857450	1.96e-06
## polygon 229	5	1978280	2.09e-06
## polygon 230	5	1944240	2.05e-06
## polygon 231	6	2728340	2.88e-06
## polygon 232	8	1905200	2.01e-06
## polygon 233	7	2258780	2.38e-06
## polygon 234	5	1891180	1.99e-06
## polygon 235	5	1957860	2.06e-06
## polygon 236	7	2335470	2.46e-06
## polygon 237	4	2055120	2.17e-06
## polygon 238	6	1829730	1.93e-06
## polygon 239	5	1807400	1.91e-06
## polygon 240	5	2014070	2.12e-06
## polygon 241	5	1882630	1.99e-06
## polygon 242	5	1928800	2.03e-06
## polygon 243	5	2010320	2.12e-06
## polygon 244	5	1521870	1.60e-06
## polygon 245	6	1935590	2.04e-06
## polygon 246	5	1647440	1.74e-06

## polygon 247	4	1518380	1.60e-06
## polygon 248	6	1623490	1.71e-06
## polygon 249	5	1654240	1.74e-06
## polygon 250	5	1691180	1.78e-06
## polygon 251	3	765559	8.07e-07
## polygon 252	6	1792550	1.89e-06
## polygon 253	5	1428170	1.51e-06
## polygon 254	5	1356230	1.43e-06
## polygon 255	5	1536990	1.62e-06
## polygon 256	5	1188700	1.25e-06
## polygon 257	4	846360	8.93e-07
## polygon 258	5	1449400	1.53e-06
## polygon 259	4	1654610	1.74e-06
## polygon 260	6	1472190	1.55e-06
## polygon 261	4	1163890	1.23e-06
## polygon 262	7	1548860	1.63e-06
## polygon 263	6	1317820	1.39e-06
## polygon 264	5	1220870	1.29e-06
## polygon 265	4	1251210	1.32e-06
## polygon 266	6	1443280	1.52e-06
## polygon 267	5	1430870	1.51e-06
## polygon 268	5	1201830	1.27e-06
## polygon 269	7	1083120	1.14e-06
## polygon 270	5	1315200	1.39e-06
## polygon 271	5	1173590	1.24e-06
## polygon 272	5	1283640	1.35e-06
## polygon 273	5	1243830	1.31e-06
## polygon 274	5	1418840	1.50e-06
## polygon 275	6	1194150	1.26e-06
## polygon 276	3	732692	7.73e-07
## polygon 277	4	724047	7.64e-07
## polygon 278	4	1110880	1.17e-06
## polygon 279	4	1068690	1.13e-06
## polygon 280	5	1080070	1.14e-06
## polygon 281	5	1105010	1.17e-06
## polygon 282	4	1101910	1.16e-06
## polygon 283	5	1202270	1.27e-06
## polygon 284	5	1193280	1.26e-06
## polygon 285	5	1104420	1.16e-06
## polygon 286	5	1337410	1.41e-06
## polygon 287	6	1293100	1.36e-06
## polygon 288	4	715019	7.54e-07
## polygon 289	3	772725	8.15e-07
## polygon 290	4	856683	9.03e-07
## polygon 291	4	986692	1.04e-06
## polygon 292	4	808047	8.52e-07
## polygon 293	4	740351	7.81e-07
## polygon 294	4	839774	8.86e-07
## polygon 295	5	997422	1.05e-06
## polygon 296	4	841098	8.87e-07
## polygon 297	7	1180430	1.24e-06
## polygon 298	4	1072900	1.13e-06
## polygon 299	5	1071650	1.13e-06
## polygon 300	4	937949	9.89e-07

## polygon 301	4	714701	7.54e-07
## polygon 302	4	854803	9.01e-07
## polygon 303	4	965776	1.02e-06
## polygon 304	4	875646	9.23e-07
## polygon 305	4	682625	7.20e-07
## polygon 306	4	799391	8.43e-07
## polygon 307	4	915082	9.65e-07
## polygon 308	4	647125	6.82e-07
## polygon 309	5	850200	8.97e-07
## polygon 310	4	856777	9.04e-07
## polygon 311	5	926662	9.77e-07
## polygon 312	4	702365	7.41e-07
## polygon 313	4	877724	9.26e-07
## polygon 314	4	625455	6.60e-07
## polygon 315	4	604921	6.38e-07
## polygon 316	4	564256	5.95e-07
## polygon 317	5	943607	9.95e-07
## polygon 318	4	725841	7.65e-07
## polygon 319	4	838391	8.84e-07
## polygon 320	4	725920	7.66e-07
## polygon 321	4	607726	6.41e-07
## polygon 322	3	377075	3.98e-07
## polygon 323	4	801044	8.45e-07
## polygon 324	4	707601	7.46e-07
## polygon 325	4	701275	7.40e-07
## polygon 326	3	567574	5.99e-07
## polygon 327	4	656976	6.93e-07
## polygon 328	4	697610	7.36e-07
## polygon 329	3	429629	4.53e-07
## polygon 330	4	692723	7.31e-07
## polygon 331	5	1123680	1.18e-06
## polygon 332	4	638721	6.74e-07
## polygon 333	4	649515	6.85e-07
## polygon 334	5	885457	9.34e-07
## polygon 335	3	397429	4.19e-07
## polygon 336	3	311243	3.28e-07
## polygon 337	4	692962	7.31e-07
## polygon 338	4	637359	6.72e-07
## polygon 339	3	347030	3.66e-07
## polygon 340	4	545022	5.75e-07
## polygon 341	4	630075	6.64e-07
## polygon 342	4	682861	7.20e-07
## polygon 343	4	582135	6.14e-07
## polygon 344	4	685738	7.23e-07
## polygon 345	3	666827	7.03e-07
## polygon 346	4	638063	6.73e-07
## polygon 347	3	415934	4.39e-07
## polygon 348	3	496699	5.24e-07
## polygon 349	4	523038	5.52e-07
## polygon 350	4	629200	6.64e-07
## polygon 351	4	545075	5.75e-07
## polygon 352	4	523200	5.52e-07
## polygon 353	4	534278	5.63e-07
## polygon 354	4	617265	6.51e-07

```

## polygon 355          4      554460      5.85e-07
## polygon 356          3      422197      4.45e-07
## polygon 357          4      450587      4.75e-07
## polygon 358          3      506804      5.34e-07
## polygon 359          4      666360      7.03e-07
## polygon 360          4      648581      6.84e-07
## polygon 361          3      453422      4.78e-07
## polygon 362          4      572761      6.04e-07
## polygon 363          4      515850      5.44e-07
## polygon 364          4      501575      5.29e-07
## polygon 365          4      501226      5.29e-07
## polygon 366          4      601941      6.35e-07
## polygon 367          4      593073      6.25e-07
## polygon 368          3      427855      4.51e-07
## polygon 369          3      295928      3.12e-07
## polygon 370          4      640706      6.76e-07
## polygon 371          5      714352      7.53e-07
## polygon 372          4      494594      5.22e-07
## polygon 373          3      413224      4.36e-07
## polygon 374          3      254908      2.69e-07
## polygon 375          4      458639      4.84e-07
## polygon 376          3      374163      3.95e-07
## polygon 377          4      514470      5.43e-07
## polygon 378          4      419372      4.42e-07
## polygon 379          3      405491      4.28e-07
## enclosing rectangle: [273874.9, 1870573.4] x [369042.8, 1735666.4] units
##                               (1597000 x 1367000 units)
## Window area = 9.48262e+11 square units
## Fraction of frame area: 0.435
##
## Dummy quadrature points:
##     320 x 320 grid of dummy points, plus 4 corner points
##     dummy spacing: 4989.683 x 4270.699 units
##
## Original dummy parameters: =
## Planar point pattern: 44503 points
## Average intensity 4.69e-08 points per square unit
## Window: polygonal boundary
## 379 separate polygons (2 holes)
##           vertices      area relative.area
## polygon 1          5454 889865000000  9.38e-01
## polygon 2          2129 319023000000  3.36e-02
## polygon 3 (hole)    55   -49871200  -5.26e-05
## polygon 4 (hole)    21   -15073400  -1.59e-05
## polygon 5          751   6476610000  6.83e-03
## polygon 6          764   2633530000  2.78e-03
## polygon 7          379   2286840000  2.41e-03
## polygon 8          281   1397560000  1.47e-03
## polygon 9          155   1020880000  1.08e-03
## polygon 10         178   822976000  8.68e-04
## polygon 11         189   531172000  5.60e-04
## polygon 12         135   523600000  5.52e-04
## polygon 13          93   429077000  4.52e-04
## polygon 14          59   375017000  3.95e-04

```

## polygon 15	80	366555000	3.87e-04
## polygon 16	130	389457000	4.11e-04
## polygon 17	129	361761000	3.81e-04
## polygon 18	73	325642000	3.43e-04
## polygon 19	72	284295000	3.00e-04
## polygon 20	98	291606000	3.08e-04
## polygon 21	87	301693000	3.18e-04
## polygon 22	60	271131000	2.86e-04
## polygon 23	105	273020000	2.88e-04
## polygon 24	67	238816000	2.52e-04
## polygon 25	72	240998000	2.54e-04
## polygon 26	51	207196000	2.19e-04
## polygon 27	81	175770000	1.85e-04
## polygon 28	72	186213000	1.96e-04
## polygon 29	60	170593000	1.80e-04
## polygon 30	52	171879000	1.81e-04
## polygon 31	62	174668000	1.84e-04
## polygon 32	55	163246000	1.72e-04
## polygon 33	63	151745000	1.60e-04
## polygon 34	62	164385000	1.73e-04
## polygon 35	70	155140000	1.64e-04
## polygon 36	63	152920000	1.61e-04
## polygon 37	72	134301000	1.42e-04
## polygon 38	88	135277000	1.43e-04
## polygon 39	93	128722000	1.36e-04
## polygon 40	45	127343000	1.34e-04
## polygon 41	84	130472000	1.38e-04
## polygon 42	49	114370000	1.21e-04
## polygon 43	44	106635000	1.12e-04
## polygon 44	53	102951000	1.09e-04
## polygon 45	52	100309000	1.06e-04
## polygon 46	45	95770200	1.01e-04
## polygon 47	43	95377600	1.01e-04
## polygon 48	39	92824200	9.79e-05
## polygon 49	47	81225900	8.57e-05
## polygon 50	40	80617200	8.50e-05
## polygon 51	64	83219400	8.78e-05
## polygon 52	33	76223000	8.04e-05
## polygon 53	34	67360500	7.10e-05
## polygon 54	54	68347400	7.21e-05
## polygon 55	44	64136400	6.76e-05
## polygon 56	42	66265500	6.99e-05
## polygon 57	29	62988400	6.64e-05
## polygon 58	31	59626500	6.29e-05
## polygon 59	25	54751800	5.77e-05
## polygon 60	31	55352900	5.84e-05
## polygon 61	30	53162300	5.61e-05
## polygon 62	44	57621600	6.08e-05
## polygon 63	26	56707500	5.98e-05
## polygon 64	36	56979200	6.01e-05
## polygon 65	32	55472500	5.85e-05
## polygon 66	21	50455600	5.32e-05
## polygon 67	40	51039400	5.38e-05
## polygon 68	26	53043500	5.59e-05

## polygon 69	29	51938100	5.48e-05
## polygon 70	27	50295800	5.30e-05
## polygon 71	33	47938300	5.06e-05
## polygon 72	18	44300200	4.67e-05
## polygon 73	25	45691800	4.82e-05
## polygon 74	29	43313200	4.57e-05
## polygon 75	36	46561100	4.91e-05
## polygon 76	31	46220100	4.87e-05
## polygon 77	26	40401700	4.26e-05
## polygon 78	30	43204000	4.56e-05
## polygon 79	26	41430100	4.37e-05
## polygon 80	25	36988000	3.90e-05
## polygon 81	28	37700100	3.98e-05
## polygon 82	20	35892800	3.79e-05
## polygon 83	34	38217700	4.03e-05
## polygon 84	21	34874700	3.68e-05
## polygon 85	25	33536100	3.54e-05
## polygon 86	30	36738900	3.87e-05
## polygon 87	30	35393400	3.73e-05
## polygon 88	23	32140100	3.39e-05
## polygon 89	28	31469400	3.32e-05
## polygon 90	23	32736900	3.45e-05
## polygon 91	28	34117200	3.60e-05
## polygon 92	20	31802600	3.35e-05
## polygon 93	26	30659000	3.23e-05
## polygon 94	24	29572700	3.12e-05
## polygon 95	27	27289800	2.88e-05
## polygon 96	16	23094400	2.44e-05
## polygon 97	23	24218600	2.55e-05
## polygon 98	18	21846600	2.30e-05
## polygon 99	14	23702300	2.50e-05
## polygon 100	21	24325800	2.57e-05
## polygon 101	23	22590100	2.38e-05
## polygon 102	24	19177800	2.02e-05
## polygon 103	22	18657800	1.97e-05
## polygon 104	15	17454800	1.84e-05
## polygon 105	14	16276800	1.72e-05
## polygon 106	13	15792300	1.67e-05
## polygon 107	17	16117900	1.70e-05
## polygon 108	12	16217100	1.71e-05
## polygon 109	19	16610700	1.75e-05
## polygon 110	18	16809400	1.77e-05
## polygon 111	15	15096400	1.59e-05
## polygon 112	14	14571400	1.54e-05
## polygon 113	11	13567300	1.43e-05
## polygon 114	16	13623600	1.44e-05
## polygon 115	15	13657400	1.44e-05
## polygon 116	18	14346200	1.51e-05
## polygon 117	15	12418500	1.31e-05
## polygon 118	22	13085700	1.38e-05
## polygon 119	11	13446800	1.42e-05
## polygon 120	14	13234700	1.40e-05
## polygon 121	15	13532800	1.43e-05
## polygon 122	16	12575800	1.33e-05

## polygon 123	15	13060700	1.38e-05
## polygon 124	15	11766100	1.24e-05
## polygon 125	18	11038100	1.16e-05
## polygon 126	12	12863200	1.36e-05
## polygon 127	19	11135700	1.17e-05
## polygon 128	11	10881600	1.15e-05
## polygon 129	14	12500500	1.32e-05
## polygon 130	12	10733900	1.13e-05
## polygon 131	15	11201200	1.18e-05
## polygon 132	13	10674100	1.13e-05
## polygon 133	13	9776920	1.03e-05
## polygon 134	14	10503500	1.11e-05
## polygon 135	12	9391460	9.90e-06
## polygon 136	10	9402110	9.92e-06
## polygon 137	9	9160880	9.66e-06
## polygon 138	11	8674110	9.15e-06
## polygon 139	12	9159420	9.66e-06
## polygon 140	14	8676590	9.15e-06
## polygon 141	12	9527540	1.00e-05
## polygon 142	12	8797870	9.28e-06
## polygon 143	11	8837670	9.32e-06
## polygon 144	13	9067690	9.56e-06
## polygon 145	12	7950440	8.38e-06
## polygon 146	9	8727650	9.20e-06
## polygon 147	8	8565660	9.03e-06
## polygon 148	8	6790020	7.16e-06
## polygon 149	11	6780640	7.15e-06
## polygon 150	11	7326980	7.73e-06
## polygon 151	10	6152880	6.49e-06
## polygon 152	13	6721370	7.09e-06
## polygon 153	8	6343870	6.69e-06
## polygon 154	11	6807170	7.18e-06
## polygon 155	10	6927150	7.31e-06
## polygon 156	9	6352310	6.70e-06
## polygon 157	13	6217600	6.56e-06
## polygon 158	15	6263440	6.61e-06
## polygon 159	11	5769240	6.08e-06
## polygon 160	8	5965270	6.29e-06
## polygon 161	10	6826030	7.20e-06
## polygon 162	10	5749960	6.06e-06
## polygon 163	15	5352810	5.64e-06
## polygon 164	9	5781470	6.10e-06
## polygon 165	13	5358790	5.65e-06
## polygon 166	10	6016930	6.35e-06
## polygon 167	9	5928820	6.25e-06
## polygon 168	8	5424110	5.72e-06
## polygon 169	9	5468680	5.77e-06
## polygon 170	11	5209670	5.49e-06
## polygon 171	9	4705250	4.96e-06
## polygon 172	9	4891810	5.16e-06
## polygon 173	11	5295110	5.58e-06
## polygon 174	7	4839440	5.10e-06
## polygon 175	7	4322930	4.56e-06
## polygon 176	7	4514530	4.76e-06

## polygon 177	7	4573450	4.82e-06
## polygon 178	9	5080950	5.36e-06
## polygon 179	12	4492120	4.74e-06
## polygon 180	8	4783060	5.04e-06
## polygon 181	8	3800770	4.01e-06
## polygon 182	5	4657790	4.91e-06
## polygon 183	11	4323150	4.56e-06
## polygon 184	9	3549270	3.74e-06
## polygon 185	6	4103470	4.33e-06
## polygon 186	8	3898980	4.11e-06
## polygon 187	8	4009440	4.23e-06
## polygon 188	9	3672080	3.87e-06
## polygon 189	11	3540120	3.73e-06
## polygon 190	6	3790670	4.00e-06
## polygon 191	8	3695580	3.90e-06
## polygon 192	7	3708650	3.91e-06
## polygon 193	7	3843670	4.05e-06
## polygon 194	7	3157020	3.33e-06
## polygon 195	9	2913680	3.07e-06
## polygon 196	9	3368170	3.55e-06
## polygon 197	7	3190600	3.36e-06
## polygon 198	6	3073660	3.24e-06
## polygon 199	9	3016490	3.18e-06
## polygon 200	8	3129780	3.30e-06
## polygon 201	6	3101410	3.27e-06
## polygon 202	7	2895440	3.05e-06
## polygon 203	8	2998870	3.16e-06
## polygon 204	7	2619570	2.76e-06
## polygon 205	10	3007320	3.17e-06
## polygon 206	7	2324090	2.45e-06
## polygon 207	10	2935450	3.10e-06
## polygon 208	10	2523640	2.66e-06
## polygon 209	6	2232050	2.35e-06
## polygon 210	6	2131190	2.25e-06
## polygon 211	6	1679950	1.77e-06
## polygon 212	6	2226470	2.35e-06
## polygon 213	8	2267800	2.39e-06
## polygon 214	6	2584690	2.73e-06
## polygon 215	5	2151150	2.27e-06
## polygon 216	6	1536960	1.62e-06
## polygon 217	6	2616700	2.76e-06
## polygon 218	7	2205520	2.33e-06
## polygon 219	8	2008210	2.12e-06
## polygon 220	5	2181540	2.30e-06
## polygon 221	7	2700450	2.85e-06
## polygon 222	7	2493440	2.63e-06
## polygon 223	7	2752380	2.90e-06
## polygon 224	5	2036350	2.15e-06
## polygon 225	5	2344480	2.47e-06
## polygon 226	5	1786340	1.88e-06
## polygon 227	8	2269610	2.39e-06
## polygon 228	6	1857450	1.96e-06
## polygon 229	5	1978280	2.09e-06
## polygon 230	5	1944240	2.05e-06

## polygon 231	6	2728340	2.88e-06
## polygon 232	8	1905200	2.01e-06
## polygon 233	7	2258780	2.38e-06
## polygon 234	5	1891180	1.99e-06
## polygon 235	5	1957860	2.06e-06
## polygon 236	7	2335470	2.46e-06
## polygon 237	4	2055120	2.17e-06
## polygon 238	6	1829730	1.93e-06
## polygon 239	5	1807400	1.91e-06
## polygon 240	5	2014070	2.12e-06
## polygon 241	5	1882630	1.99e-06
## polygon 242	5	1928800	2.03e-06
## polygon 243	5	2010320	2.12e-06
## polygon 244	5	1521870	1.60e-06
## polygon 245	6	1935590	2.04e-06
## polygon 246	5	1647440	1.74e-06
## polygon 247	4	1518380	1.60e-06
## polygon 248	6	1623490	1.71e-06
## polygon 249	5	1654240	1.74e-06
## polygon 250	5	1691180	1.78e-06
## polygon 251	3	765559	8.07e-07
## polygon 252	6	1792550	1.89e-06
## polygon 253	5	1428170	1.51e-06
## polygon 254	5	1356230	1.43e-06
## polygon 255	5	1536990	1.62e-06
## polygon 256	5	1188700	1.25e-06
## polygon 257	4	846360	8.93e-07
## polygon 258	5	1449400	1.53e-06
## polygon 259	4	1654610	1.74e-06
## polygon 260	6	1472190	1.55e-06
## polygon 261	4	1163890	1.23e-06
## polygon 262	7	1548860	1.63e-06
## polygon 263	6	1317820	1.39e-06
## polygon 264	5	1220870	1.29e-06
## polygon 265	4	1251210	1.32e-06
## polygon 266	6	1443280	1.52e-06
## polygon 267	5	1430870	1.51e-06
## polygon 268	5	1201830	1.27e-06
## polygon 269	7	1083120	1.14e-06
## polygon 270	5	1315200	1.39e-06
## polygon 271	5	1173590	1.24e-06
## polygon 272	5	1283640	1.35e-06
## polygon 273	5	1243830	1.31e-06
## polygon 274	5	1418840	1.50e-06
## polygon 275	6	1194150	1.26e-06
## polygon 276	3	732692	7.73e-07
## polygon 277	4	724047	7.64e-07
## polygon 278	4	1110880	1.17e-06
## polygon 279	4	1068690	1.13e-06
## polygon 280	5	1080070	1.14e-06
## polygon 281	5	1105010	1.17e-06
## polygon 282	4	1101910	1.16e-06
## polygon 283	5	1202270	1.27e-06
## polygon 284	5	1193280	1.26e-06

## polygon 285	5	1104420	1.16e-06
## polygon 286	5	1337410	1.41e-06
## polygon 287	6	1293100	1.36e-06
## polygon 288	4	715019	7.54e-07
## polygon 289	3	772725	8.15e-07
## polygon 290	4	856683	9.03e-07
## polygon 291	4	986692	1.04e-06
## polygon 292	4	808047	8.52e-07
## polygon 293	4	740351	7.81e-07
## polygon 294	4	839774	8.86e-07
## polygon 295	5	997422	1.05e-06
## polygon 296	4	841098	8.87e-07
## polygon 297	7	1180430	1.24e-06
## polygon 298	4	1072900	1.13e-06
## polygon 299	5	1071650	1.13e-06
## polygon 300	4	937949	9.89e-07
## polygon 301	4	714701	7.54e-07
## polygon 302	4	854803	9.01e-07
## polygon 303	4	965776	1.02e-06
## polygon 304	4	875646	9.23e-07
## polygon 305	4	682625	7.20e-07
## polygon 306	4	799391	8.43e-07
## polygon 307	4	915082	9.65e-07
## polygon 308	4	647125	6.82e-07
## polygon 309	5	850200	8.97e-07
## polygon 310	4	856777	9.04e-07
## polygon 311	5	926662	9.77e-07
## polygon 312	4	702365	7.41e-07
## polygon 313	4	877724	9.26e-07
## polygon 314	4	625455	6.60e-07
## polygon 315	4	604921	6.38e-07
## polygon 316	4	564256	5.95e-07
## polygon 317	5	943607	9.95e-07
## polygon 318	4	725841	7.65e-07
## polygon 319	4	838391	8.84e-07
## polygon 320	4	725920	7.66e-07
## polygon 321	4	607726	6.41e-07
## polygon 322	3	377075	3.98e-07
## polygon 323	4	801044	8.45e-07
## polygon 324	4	707601	7.46e-07
## polygon 325	4	701275	7.40e-07
## polygon 326	3	567574	5.99e-07
## polygon 327	4	656976	6.93e-07
## polygon 328	4	697610	7.36e-07
## polygon 329	3	429629	4.53e-07
## polygon 330	4	692723	7.31e-07
## polygon 331	5	1123680	1.18e-06
## polygon 332	4	638721	6.74e-07
## polygon 333	4	649515	6.85e-07
## polygon 334	5	885457	9.34e-07
## polygon 335	3	397429	4.19e-07
## polygon 336	3	311243	3.28e-07
## polygon 337	4	692962	7.31e-07
## polygon 338	4	637359	6.72e-07

```

## polygon 339          3    347030    3.66e-07
## polygon 340          4    545022    5.75e-07
## polygon 341          4    630075    6.64e-07
## polygon 342          4    682861    7.20e-07
## polygon 343          4    582135    6.14e-07
## polygon 344          4    685738    7.23e-07
## polygon 345          3    666827    7.03e-07
## polygon 346          4    638063    6.73e-07
## polygon 347          3    415934    4.39e-07
## polygon 348          3    496699    5.24e-07
## polygon 349          4    523038    5.52e-07
## polygon 350          4    629200    6.64e-07
## polygon 351          4    545075    5.75e-07
## polygon 352          4    523200    5.52e-07
## polygon 353          4    534278    5.63e-07
## polygon 354          4    617265    6.51e-07
## polygon 355          4    554460    5.85e-07
## polygon 356          3    422197    4.45e-07
## polygon 357          4    450587    4.75e-07
## polygon 358          3    506804    5.34e-07
## polygon 359          4    666360    7.03e-07
## polygon 360          4    648581    6.84e-07
## polygon 361          3    453422    4.78e-07
## polygon 362          4    572761    6.04e-07
## polygon 363          4    515850    5.44e-07
## polygon 364          4    501575    5.29e-07
## polygon 365          4    501226    5.29e-07
## polygon 366          4    601941    6.35e-07
## polygon 367          4    593073    6.25e-07
## polygon 368          3    427855    4.51e-07
## polygon 369          3    295928    3.12e-07
## polygon 370          4    640706    6.76e-07
## polygon 371          5    714352    7.53e-07
## polygon 372          4    494594    5.22e-07
## polygon 373          3    413224    4.36e-07
## polygon 374          3    254908    2.69e-07
## polygon 375          4    458639    4.84e-07
## polygon 376          3    374163    3.95e-07
## polygon 377          4    514470    5.43e-07
## polygon 378          4    419372    4.42e-07
## polygon 379          3    405491    4.28e-07
## enclosing rectangle: [273874.9, 1870573.4] x [369042.8, 1735666.4] units
##                               (1597000 x 1367000 units)
## Window area = 9.48262e+11 square units
## Fraction of frame area: 0.435
## Quadrature weights:
##           (counting weights based on 320 x 320 array of rectangular tiles)
## All weights:
## range: [3120, 21300000] total: 9.39e+11
## Weights on data points:
## range: [3120, 14500000] total: 1.3e+10
## Weights on dummy points:
## range: [16000, 21300000]      total: 9.26e+11
## -----

```

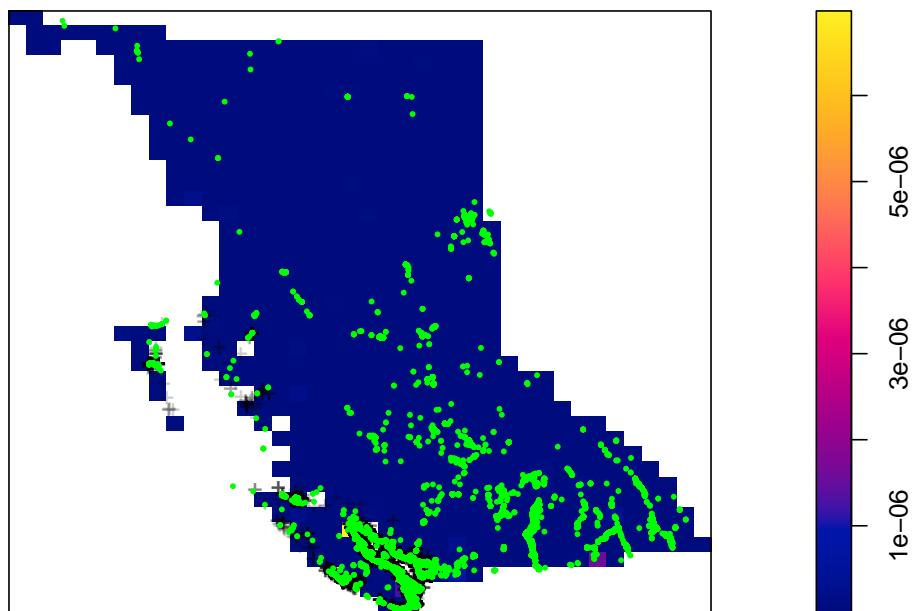
```

## FITTED :
##
## Nonstationary Poisson process
##
## ---- Intensity: ----
##
## Log intensity: ~hfi + I(hfi^2) + forest + I(forest^2) + water + I(water^2) +
## elev + I(elev^2)
## Model depends on external covariates 'hfi', 'forest', 'water' and 'elev'
## Covariates provided:
##   hfi: im
##   forest: im
##   water: im
##   elev: im
##
## Fitted trend coefficients:
##   (Intercept)      hfi      I(hfi^2)      forest      I(forest^2)
## -1.737796e+01  1.447282e+01 -8.888066e+00 -7.481589e-03 -4.582808e-05
##   water      I(water^2)      elev      I(elev^2)
## -1.801111e-03  7.942545e-08 -2.012736e-03 -6.000812e-07
##
## ----- gory details -----
##
## Fitted regular parameters (theta):
##   (Intercept)      hfi      I(hfi^2)      forest      I(forest^2)
## -1.737796e+01  1.447282e+01 -8.888066e+00 -7.481589e-03 -4.582808e-05
##   water      I(water^2)      elev      I(elev^2)
## -1.801111e-03  7.942545e-08 -2.012736e-03 -6.000812e-07
##
## Fitted exp(theta):
##   (Intercept)      hfi      I(hfi^2)      forest      I(forest^2)      water
## 2.836911e-08  1.929593e+06  1.380264e-04  9.925463e-01  9.999542e-01  9.982005e-01
##   I(water^2)      elev      I(elev^2)
## 1.000000e+00  9.979893e-01  9.999994e-01
## Problem:
## Values of the covariates 'hfi', 'forest', 'elev' were NA or undefined at 0.77% (528 out of 68828) o

# Visualize the fitted model against observed data
plot(fit7, se = FALSE, superimpose = FALSE)
plot(bird_ppp,
  pch = 16,
  cex = 0.5,
  cols = "green",
  add = TRUE)

```

Fitted trend



```
AIC(fit6); AIC(fit7)
```

```
## [1] 692760
```

```
## [1] 688681
```

```
anova(fit6, fit7, test = "LRT")
```

```
## Analysis of Deviance Table
```

```
##
```

```
## Model 1: ~forest + I(forest^2) + elev + I(elev^2) + hfi + water + I(water^2)      Poisson
```

```
## Model 2: ~hfi + I(hfi^2) + forest + I(forest^2) + water + I(water^2) + elev + I(elev^2)  Poisson
```

```
##   Npar Df Deviance Pr(>Chi)
```

```
## 1     8
```

```
## 2     9    1  4080.9 < 2.2e-16 ***
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#Run the quadrat test for fit7
```

```
quadrat.test(fit7, nx = 5, ny = 6)
```

```

##  

## Chi-squared test of fitted Poisson model 'fit7' using quadrat counts  

##  

## data: data from fit7  

## X2 = 9777.6, df = 15, p-value < 2.2e-16  

## alternative hypothesis: two.sided  

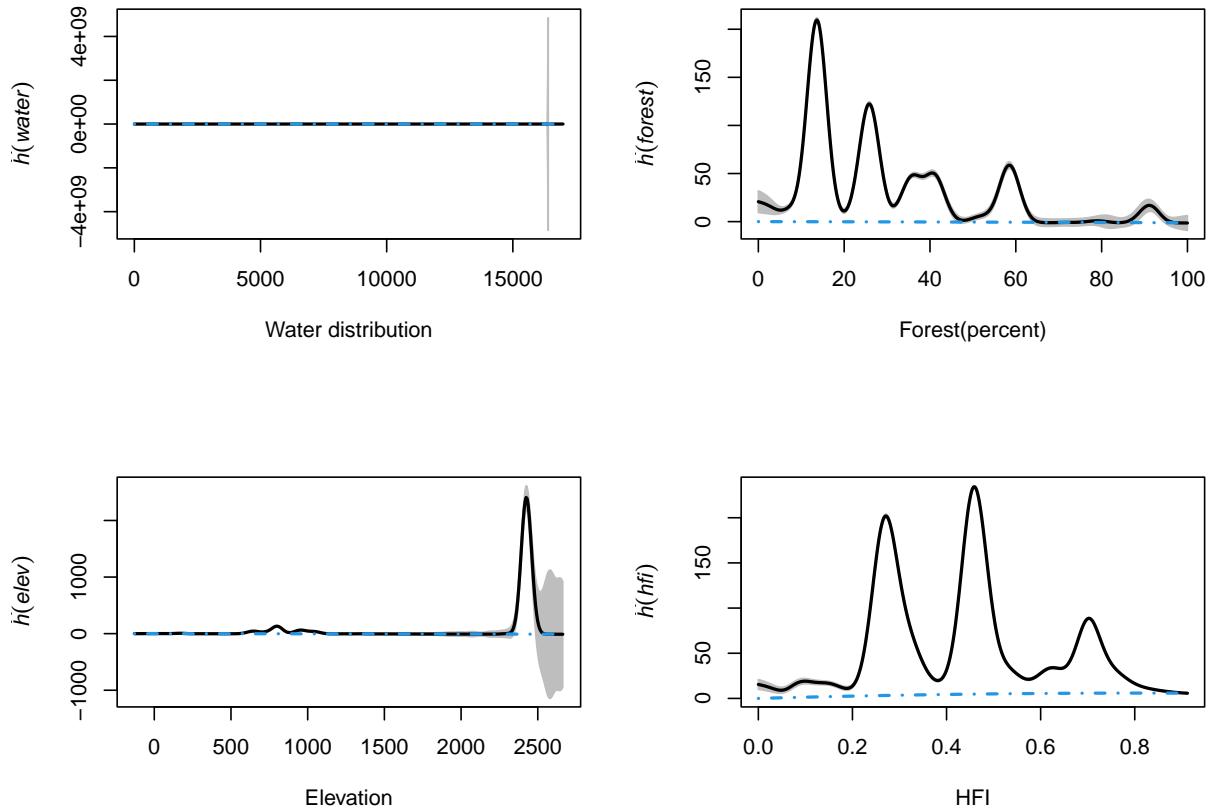
##  

## Quadrats: 24 tiles (irregular windows)

par_res_water <- parres(fit7, "water")
par_res_forest <- parres(fit7, "forest")
par_res_elev <- parres(fit7, "elev")
par_res_hfi <- parres(fit7, "hfi")

par(mfrow = c(2,2))
plot(par_res_water,
      legend = FALSE,
      lwd = 2,
      main = "",
      xlab = "Water distribution")
plot(par_res_forest,
      legend = FALSE,
      lwd = 2,
      main = "",
      xlab = "Forest(percent)")
plot(par_res_elev,
      legend = FALSE,
      lwd = 2,
      main = "",
      xlab = "Elevation")
plot(par_res_hfi,
      legend = FALSE,
      lwd = 2,
      main = "",
      xlab = "HFI")

```



```

## smooth model
library(splines)

#Fit the PPP model
fit_smooth <- ppm(bird_ppp ~ bs(elev, 7) + bs(forest, 8) + bs(water, 8) + bs(hfi, 7), use.gam = TRUE, na.action = na.omit)
fit_smooth

## Nonstationary Poisson process
## Fitted to point pattern dataset 'bird_ppp'
##
## Log intensity: ~bs(elev, 7) + bs(forest, 8) + bs(water, 8) + bs(hfi, 7)
##
## Fitted trend coefficients:
##   (Intercept)  bs(elev, 7)1  bs(elev, 7)2  bs(elev, 7)3  bs(elev, 7)4
## -14.48980264 -3.51431820 -3.37659638 -4.97792436 -5.29046223
##   bs(elev, 7)5  bs(elev, 7)6  bs(elev, 7)7  bs(forest, 8)1  bs(forest, 8)2
## -10.09293266 -17.67829689 -3.16236691 -0.49589816 -0.62643611
##   bs(forest, 8)3  bs(forest, 8)4  bs(forest, 8)5  bs(forest, 8)6  bs(forest, 8)7
## -0.01306355 -1.04201842 -0.43904502 -1.15815516 -1.68473960
##   bs(forest, 8)8  bs(water, 8)1  bs(water, 8)2  bs(water, 8)3  bs(water, 8)4
## -0.62394785 -0.31632648 -0.16567559 -0.34526231 -2.06818122
##   bs(water, 8)5  bs(water, 8)6  bs(water, 8)7  bs(water, 8)8  bs(hfi, 7)1

```

```

##      -4.25265977    -2.71386306    -5.16563355    -5.48413092    0.18082051
##  bs(hfi, 7)2    bs(hfi, 7)3    bs(hfi, 7)4    bs(hfi, 7)5    bs(hfi, 7)6
##      2.30714075    1.46607954    6.39160747    5.21248012    7.53188406
##  bs(hfi, 7)7
##      6.12080730
##
## For standard errors, type coef(summary(x))
## Problem:
## Values of the covariates 'elev', 'forest', 'hfi' were NA or undefined at 0.77%
## (528 out of 68828) of the quadrature points

AIC(fit7); AIC(fit_smooth)

## [1] 688681

## [1] 683118.2

anova(fit7, fit_smooth, test = "LRT")

## Analysis of Deviance Table
##
## Model 1: ~hfi + I(hfi^2) + forest + I(forest^2) + water + I(water^2) + elev + I(elev^2)  Poisson
## Model 2: ~bs(elev, 7) + bs(forest, 8) + bs(water, 8) + bs(hfi, 7)      Poisson
##   Npar Df Deviance  Pr(>Chi)
## 1     9
## 2    31 22    5606.9 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

par_res_water <- parres(fit_smooth, "water")
par_res_forest <- parres(fit_smooth, "forest")
par_res_elev <- parres(fit_smooth, "elev")
par_res_hfi <- parres(fit_smooth, "hfi")

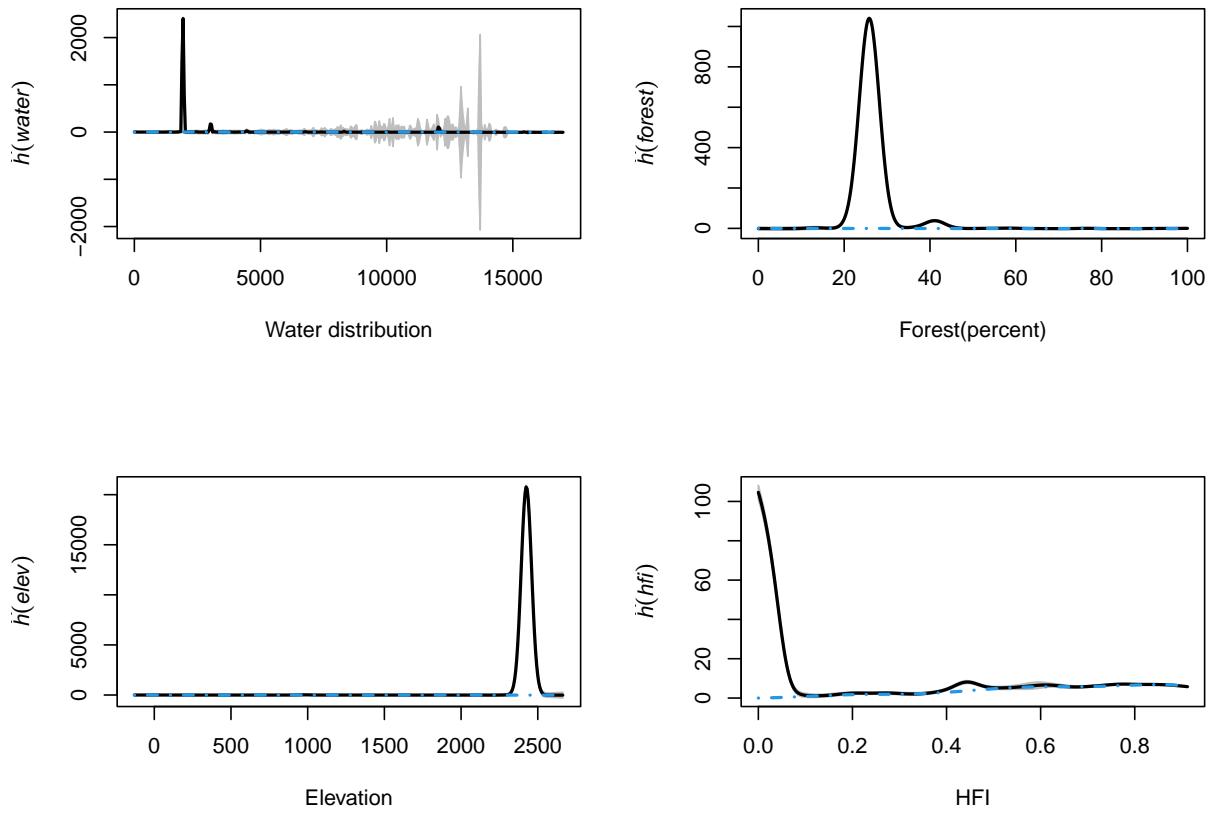
par(mfrow = c(2,2))
plot(par_res_water,
      legend = FALSE,
      lwd = 2,
      main = "",
      xlab = "Water distribution")
plot(par_res_forest,
      legend = FALSE,
      lwd = 2,
      main = "",
      xlab = "Forest(percent)")
plot(par_res_elev,
      legend = FALSE,
      lwd = 2,
      main = "",
      xlab = "Elevation")
plot(par_res_hfi,
      legend = FALSE,

```

```

lwd = 2,
main = "",
xlab = "HFI")

```



```

#Plot the model predictions
plot(fit_smooth,
      se = FALSE,
      superimpose = TRUE)

#Overlay the occurrence
plot(bird_ppp,
      pch = 16,
      cex = 0.6,
      cols = "green",
      add = TRUE)

```

Fitted trend

