Assignment 3 Spatial Epidemiology

Ferrara Lorenzo, Lucchini Marco

Task 1: Introduction to Spatial Point Process

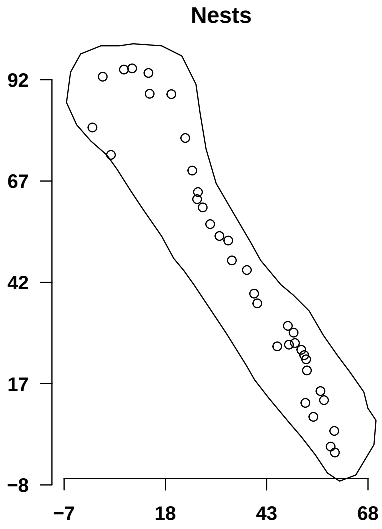
The nest data from islet "nucli 23" is stored in nucli23.txt. Additionally, the coordinates of the islet are in poly23.txt.

1) Build a ppp object using the "poly23" data as a window

```
# the data
nucli.23 = read.delim("T1/nucli23.txt")
min.X = min(nucli.23$X)
min.Y = min(nucli.23$Y)

nucli.23$X = nucli.23$X - min.X
nucli.23$Y = nucli.23$Y - min.Y

max.X = max(nucli.23$X)
max.Y = max(nucli.23$Y)
```

```
# the polygon object for the border
poligon = poly23 <- read.delim("T1/poly23.txt")
poligon$X = poligon$X - min.X
poligon$Y = poligon$Y - min.Y
pol.illa <- list(x = poligon$X, y = poligon$Y)

min.pX = min(poligon$X)
min.pY = min(poligon$Y)
max.pX = max(poligon$X)
max.pY = max(poligon$Y)
```

```
# the final object
n23p = ppp(nucli.23$X, nucli.23$Y, poly = pol.illa)
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(1, 0,
    1, 0))
plot(n23p, main = "Nests")
axis(1, at = c(floor(seq(min.pX, max.pX, by = 25))), pos = c(min.pX, min.pY - 15))
axis(2, at = c(floor(seq(min.pY, max.pY, by = 25))), pos = c(-10, 0))
```

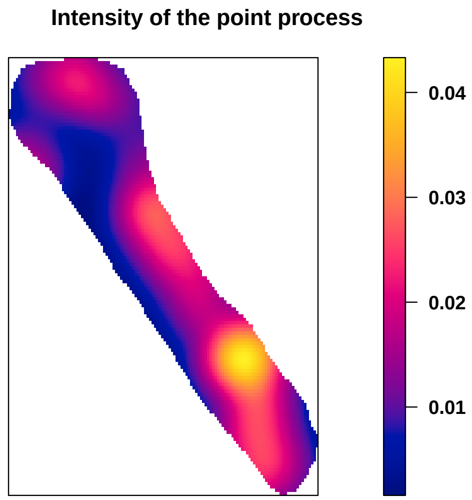


2) Describe the point pattern process and its intensity.

```
summary(n23p)
```

```
## Planar point pattern:  36 points
## Average intensity 0.01451131 points per square unit
##
## Coordinates are given to 6 decimal places
##
## Window: polygonal boundary
## single connected closed polygon with 47 vertices
## enclosing rectangle: [-6.40267, 69.97237] x [-7.06032, 100.8694] units
##                      (76.38 x 107.9 units)
## Window area = 2480.82 square units
## Fraction of frame area: 0.301
```

```
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    2, 0))
plot(density(n23p, dimax.Yx = c(256, 256), sigma = 5.5), main = "Intensity of the point process")
```



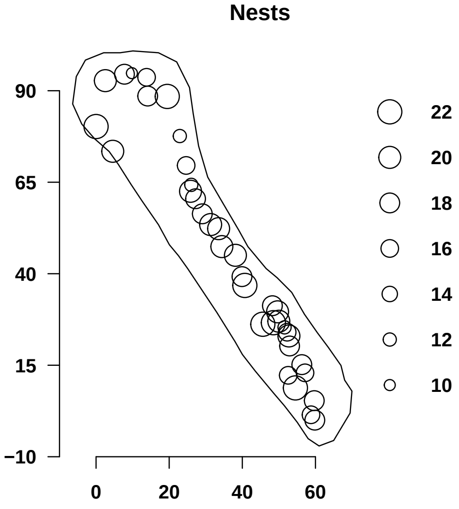
The density plot reveals that the majority of the nests are located in the center and south, along the north coast. There is a notable concentration, indicated by the yellow peak, in which a large number of nests were found.

This distribution does not appear to be completely random and appears to have a pattern.

3) Create a multi-type mark indicating the order of the nesting according to the nesting time, Using the time intervals: [10,16], [17,19], [20,22].

```
n23T = ppp(nucli.23$X, nucli.23$Y, poly = pol.illa, marks = nucli.23$data_pos)
```

```
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(2, 0,
    2, 0))
plot(n23T, main = "Nests", markscale = 0.3, leg.side = "right")
axis(1, at = c(seq(0, max.pX, by = 20)), pos = c(-10, 0))
axis(2, at = c(seq(-10, max.pY, by = 25)), pos = c(-10, 0))
```



```
DPOScat = cut(nucli.23$data_pos, breaks = c(9, 16, 19, 22), labels = c("10-16", "17-19",
    "20-22"))
table(DPOScat)
```

```
## DPOScat
## 10-16 17-19 20-22
##    10    10    16
```

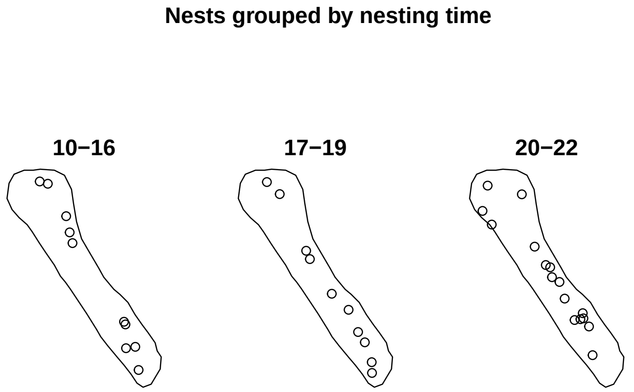
The suddivision of the nests in the 3 temporal groups seems homogeneous

```
n23Tcat = ppp(nucli.23$X, nucli.23$Y, poly = pol.illa, marks = DPOScat)
summary(n23Tcat)
```

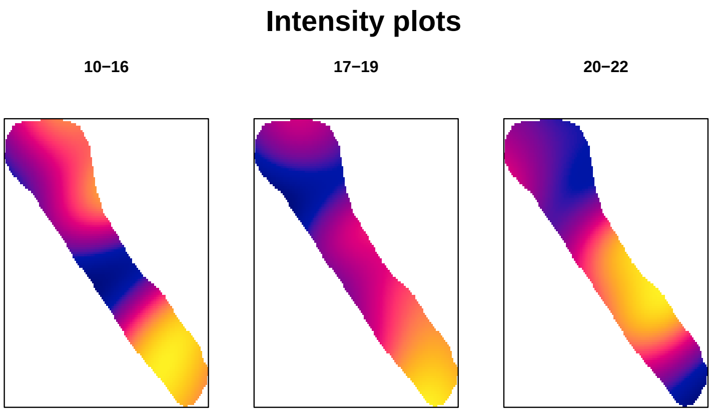
```
## Marked planar point pattern:  36 points
## Average intensity 0.01451131 points per square unit
##
## Coordinates are given to 6 decimal places
##
## Multitype:
##       frequency proportion   intensity
## 10-16        10  0.2777778 0.004030920
## 17-19        10  0.2777778 0.004030920
## 20-22        16  0.4444444 0.006449472
##
## Window: polygonal boundary
## single connected closed polygon with 47 vertices
## enclosing rectangle: [-6.40267, 69.97237] x [-7.06032, 100.8694] units
##                      (76.38 x 107.9 units)
## Window area = 2480.82 square units
## Fraction of frame area: 0.301
```

4) Describe the marked point process and its intensity

```
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    1, 0))
plot(split(n23Tcat), main = "Nests grouped by nesting time")
```



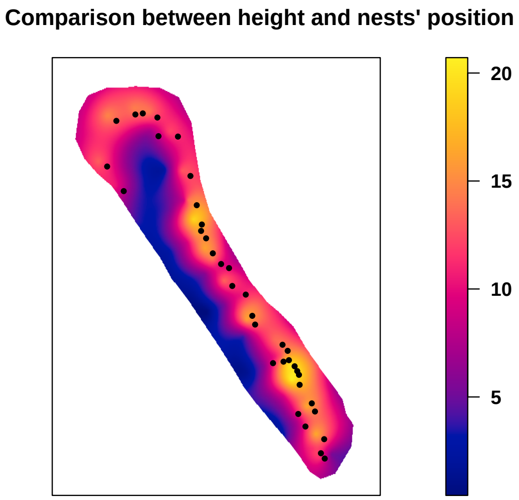
```
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    1, 0))
plot(density(split(n23Tcat), sigma = 12), ribbon = FALSE, main = "Intensity plots")
```



Different patterns can be observed in the three time intervals, with a distribution initially located in the extreme south that gradually moves towards the center.

5) Add to the analysis the height of the islet.

```
grid <- read.csv("T1/grid_height_23.txt", sep = "")
mat <- as.matrix(read.table("T1/height_23.txt"))
mat[mat == 0] = NA
height <- im(mat, grid$x, grid$y)
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    2, 0))
plot(height, axis = T, main = "Comparison between height and nests' position")
plot(n23p, add = T, cex = 0.7, pch = 16)
```



Comparison between height and nests' position

This plot allows us to observe a correlation between the positions of the nest and the elevation of the islet. It appears that the nests are concentrated in areas with higher elevation. Therefore, we can conclude that the point process is not completely random and the assumption of a Homogeneous Poisson Process can be rejected.

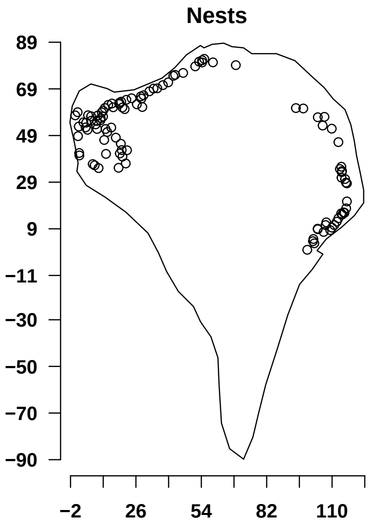
Task 2: Intensity and Randomness

The nest data from islet “nucli 84” is stored in nucli84.txt. Additionally, the coordinates of the islet are in poly84.txt.

1) Build a ppp object using the “nucli 84” data.

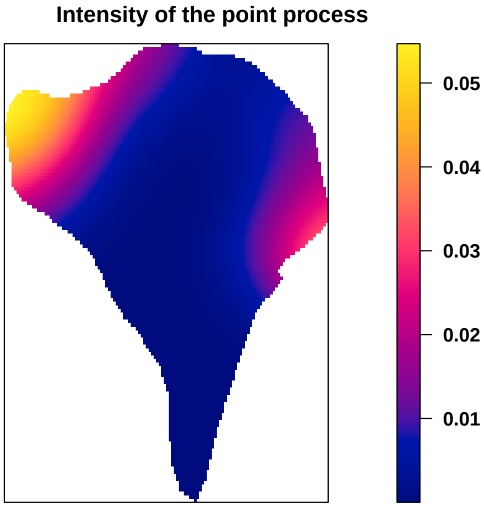
```r
rm(list = ls())
nucli84 <- read.delim("T2/nucli84.txt")

min.X = min(nucli84$X)
min.Y = min(nucli84$Y)
max.X = max(nucli84$X)
max.Y = max(nucli84$Y)

nucli84$X = nucli84$X - min.X
nucli84$Y = nucli84$Y - min.Y

n84 = ppp(x = nucli84$X, y = nucli84$Y, range(nucli84$X), range(nucli84$Y))

poligon = read.delim("T2/poly84.txt")

poligon$X = poligon$X - min.X
poligon$Y = poligon$Y - min.Y
pol.illa <- list(x = poligon$X, y = poligon$Y)

min.pX = min(poligon$X)
min.pY = min(poligon$Y)
max.pX = max(poligon$X)
max.pY = max(poligon$Y)

n84p = ppp(nucli84$X, nucli84$Y, poly = pol.illa, range(poligon$X), range(poligon$Y))

islet_window = owin(poly = pol.illa)
```

```r
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(2, 0,
    1, 0))
plot(n84p, main = "Nests")
axis(1, at = c(round(seq(min.pX, max.pX, length = 10), digits = 0)))  #, pos=c(0,0))
axis(2, at = c(round(seq(min.pY, max.pY, length = 10), digits = 0)), pos = c(min.pX -
    4, min.pY - 50))
```



2) Draw a plot with the intensity of the point process computed by the non-parametric approach. Briefly comment the results.

```
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    1, 0))
plot(density(n84p, dimax.Yx = c(256, 256), sigma = 15), main = "Intensity of the point process")
```



In the density plot, we can see that the nests are concentrated in two main areas: one in the northwest corner and one in the eastern part, both located on the seaside. No nests were observed in the central or southern areas of the islet.

This may suggest that a geographical feature is influencing the location of the nests.

We also try to fit a homogeneous Poisson process:

```
modelPois = ppm(n84, ~1)
modelPois
```

```
## Stationary Poisson process
## Intensity: 0.01091393
##                Estimate       S.E.   CI95.lo   CI95.hi Ztest      Zval
## log(lambda) -4.517715 0.09805807 -4.709905 -4.325525   *** -46.07183
```

So the intensity has point estimate `r exp(modelPois$coef)` and 95% confidence interval

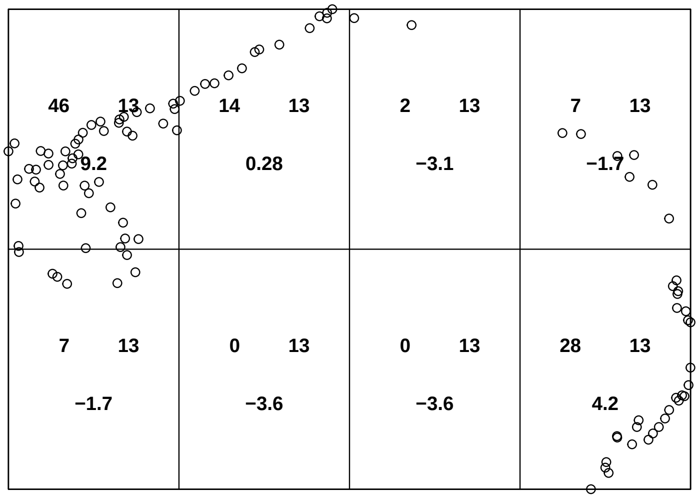
```
##        2.5 %     97.5 %
##  0.00900563 0.01322661
```

3) Assess the Completely Spatial Randomness hypothesis

- via Chi-square test:

We divide the region in 8 subareas with equal areas and under CSR we would expect more or less same number of nests in each subregion.

```
M <- quadrat.test(n84, nx = 4, ny = 2)
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    0, 0))
plot(n84, main = "")
plot(M, add = TRUE)
```



Nonetheless we notice that some subregions have more nests than other. Indeed after performing a Chi-square test we get:

```
M
```

```
##
##  Chi-squared test of CSR using quadrat counts
##
## data:  n84
## X2 = 142, df = 7, p-value < 2.2e-16
## alternative hypothesis: two.sided
##
## Quadrats: 4 by 2 grid of tiles
```

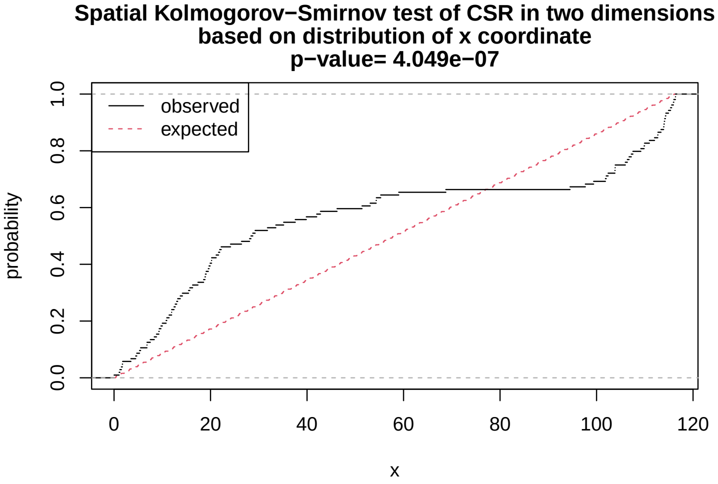
The p-value is extremely low ($p < 2.2e - 16$), which indicates that we can reject the null hypothesis of complete spatial randomness (CRS).

- via Kolmogorov-Smirnov test:

```
KS = cdf.test(n84, covariate = "x")
KS
```

```
##
##  Spatial Kolmogorov-Smirnov test of CSR in two dimensions
##
## data:  covariate 'x' evaluated at points of 'n84'
##      and transformed to uniform distribution under CSR
## D = 0.27221, p-value = 4.049e-07
## alternative hypothesis: two-sided
```

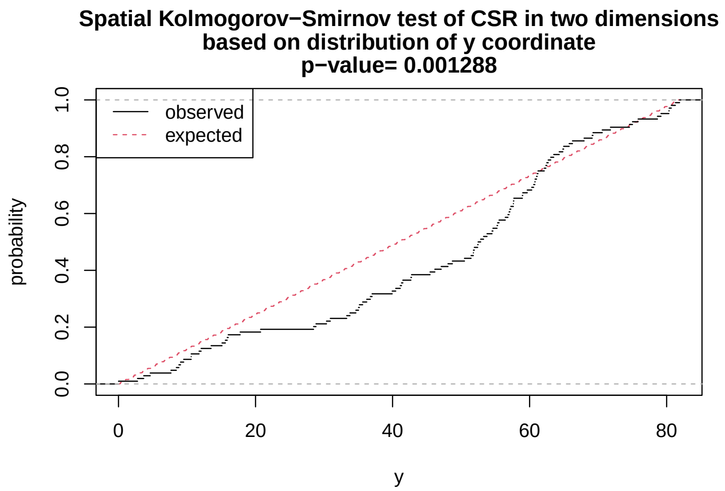
```
plot(KS)
```



```
KS = cdf.test(n84, covariate = "y")
KS
```

```
##
##  Spatial Kolmogorov-Smirnov test of CSR in two dimensions
##
## data:  covariate 'y' evaluated at points of 'n84'
##      and transformed to uniform distribution under CSR
## D = 0.18795, p-value = 0.001288
## alternative hypothesis: two-sided
```

```
plot(KS)
```



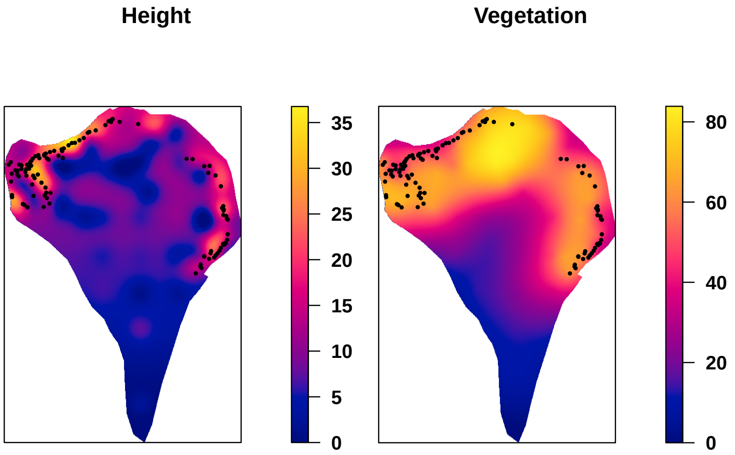
The observed distribution of Z at data points differs significantly from the expected distribution, allowing us to reject the null hypothesis of CSR and conclude that there is a dependence between the intensity of the points and both the Cartesian coordinates.

4) Assess the relation between the intensity of the point process and the covariates height and vegetation.

```
grid <- read.delim("T2/grid.txt")
grid.veg = read.delim("T2/grid_veg.txt")
veg = as.matrix(read.delim("T2/veg.txt", header = FALSE))
height = as.matrix(read.delim("T2/height.txt", header = FALSE))
```

First we visualize the values of Height and Vegetation in the islet:

```
par(mfrow = c(1, 2), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    1, 2))

Height = im(mat = height, xcol = grid$x, yrow = grid$y)
plot(Height, main = "Height", clipwin = islet_window)
plot(n84p, add = T, cex = 0.5, pch = 16)

Vegetation = im(mat = veg, xcol = grid.veg$x, yrow = grid.veg$y)
plot(Vegetation, main = "Vegetation", clipwin = islet_window)
plot(n84p, add = T, cex = 0.5, pch = 16)
```



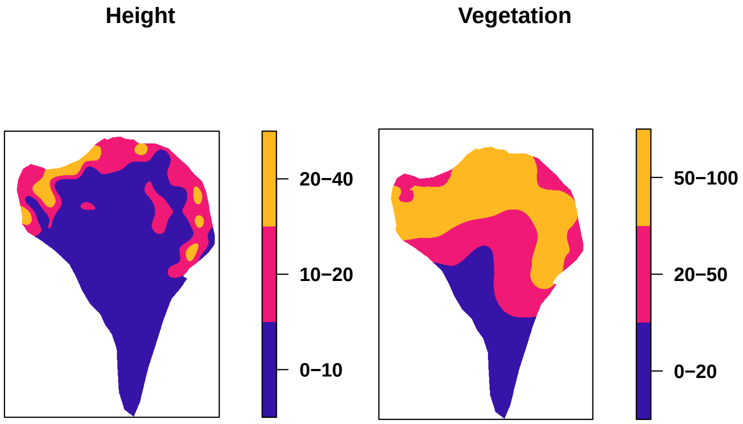
The nests seem to be concentrated in areas with higher vegetation and elevation.

Now we categorize the covariates using the suggested intervals:

- Height. [0,10], (10,20], (20,40]
- Vegetation. [0,20], (20,50], (50,100]

```
# height
brks.h <- c(0, 10, 20, 40)
Hcut <- cut(Height, breaks = brks.h, labels = c("0-10", "10-20", "20-40"))
H <- tess(image = Hcut)

# vegetation
brks.v <- c(0, 20, 50, 100)
Vcut <- cut(Vegetation, breaks = brks.v, labels = c("0-20", "20-50", "50-100"))
V <- tess(image = Vcut)
```

```
par(mfrow = c(1, 2), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    1, 3.5))
plot(H, main = "Height")
plot(V, main = "Vegetation")
```



And to assess quantitatively this relation we use:

- a Chi-squared test:

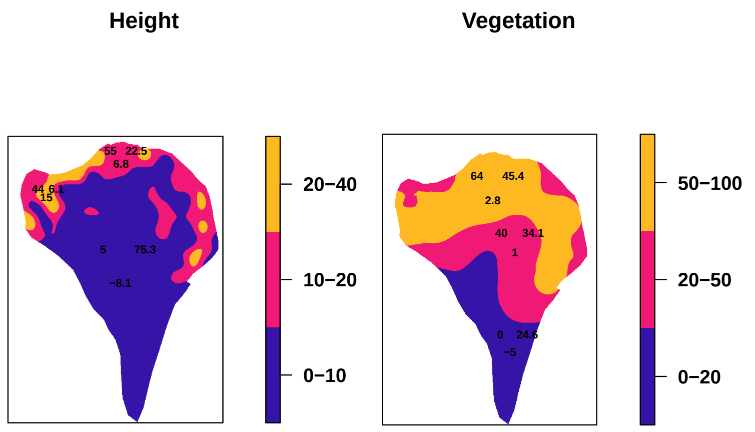
```
M.h = quadrat.test(n84, tess = H)
M.h
```

```
##
##  Chi-squared test of CSR using quadrat counts
##
## data:  n84
## X2 = 345.67, df = 2, p-value < 2.2e-16
## alternative hypothesis: two.sided
##
## Quadrats: 3 tiles (levels of a pixel image)
```

```
M.v = quadrat.test(n84p, tess = V)
M.v
```

```
##
##  Chi-squared test of CSR using quadrat counts
##
## data:  n84p
## X2 = 33.266, df = 2, p-value = 1.195e-07
## alternative hypothesis: two.sided
##
## Quadrats: 3 tiles (levels of a pixel image)
```

```
par(mfrow = c(1, 2), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    1, 3.5))
plot(M.h, valuesAreColours = FALSE, main = "Height", cex = 0.6)
plot(M.v, valuesAreColours = FALSE, main = "Vegetation", cex = 0.6)
```



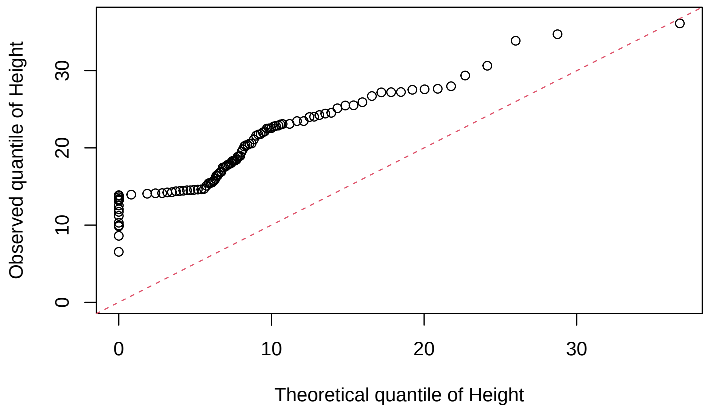
The Chi-squared tests on the two covariates Height and Vegetation have p-values: $p_1 < 1 \cdot 10^{-16}$ and $p_2 = 1.195 \cdot 10^{-7}$.

- a Kolmogorov-Smirnov test:

```
KS <- cdf.test(n84, Height)
KS
```

```
##
##  Spatial Kolmogorov-Smirnov test of CSR in two dimensions
##
## data:  covariate 'Height' evaluated at points of 'n84'
##      and transformed to uniform distribution under CSR
## D = 0.71846, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

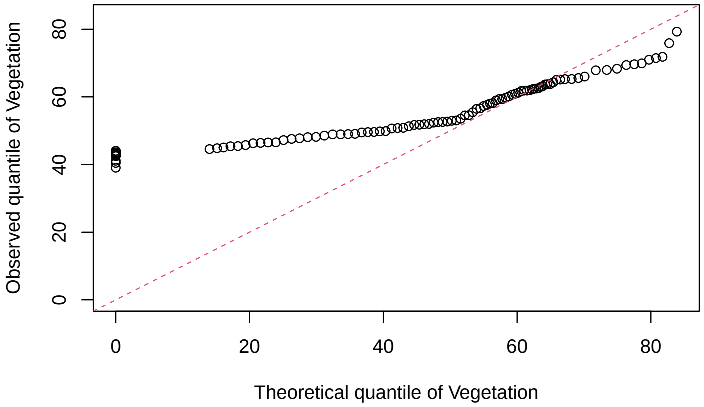
```
plot(KS, style = "QQ")
```



```
KS <- cdf.test(n84, Vegetation)
KS
```

```
##
##  Spatial Kolmogorov-Smirnov test of CSR in two dimensions
##
## data:  covariate 'Vegetation' evaluated at points of 'n84'
##      and transformed to uniform distribution under CSR
## D = 0.38911, p-value = 4.208e-14
## alternative hypothesis: two-sided
```

```
plot(KS, style = "QQ")
```



Also the Kolmogorov test has very little p-values and the plots suggest a general difference of the observed quantiles from the theoretical ones.

So, in the light of the above considerations, we can state there is in fact a relation between the covariates and the point process.

5) Fit an inhomogeneous Poisson model to data

We'll fit the model:

$$\lambda \sim x + y + \text{Height} + \text{Vegetation}$$

```
model = ppm(n84, ~x + y + Height + Vegetation)
model
```

```
## Nonstationary Poisson process
##
## Log intensity:  ~x + y + Height + Vegetation
##
## Fitted trend coefficients:
##  (Intercept)            x            y       Height   Vegetation
## -6.773462579 -0.001683221 -0.014827821  0.162420734  0.016598983
```

```
##
##                 Estimate        S.E.      CI95.lo      CI95.hi Ztest
## (Intercept) -6.773462579 0.486190000 -7.726377469 -5.820547689   ***
## x           -0.001683221 0.002823778 -0.007217724  0.003851283
## y           -0.014827821 0.005412996 -0.025437099 -0.004218544    **
## Height       0.162420734 0.012634209  0.137658140  0.187183328   ***
## Vegetation   0.016598983 0.007100232  0.002682784  0.030515183     *
##                    Zval
## (Intercept) -13.9317192
## x            -0.5960882
## y            -2.7393001
## Height       12.8556318
## Vegetation    2.3378085
```

We notice that we can remove the x variable as it's not significant, since it has the Z statistic very close to 0.

```
model1 = ppm(n84, ~y + Height + Vegetation)
model1
```

```
## Nonstationary Poisson process
##
## Log intensity:  ~y + Height + Vegetation
##
## Fitted trend coefficients:
## (Intercept)           y      Height  Vegetation
## -6.91203934 -0.01415209  0.16458833  0.01620508
##
##                Estimate        S.E.      CI95.lo      CI95.hi Ztest       Zval
## (Intercept) -6.91203934 0.433891191 -7.762450448 -6.061628232   *** -15.930352
## y           -0.01415209 0.005238198 -0.024418771 -0.003885411    **  -2.701710
## Height       0.16458833 0.012134934  0.140804296  0.188372364   ***  13.563183
## Vegetation   0.01620508 0.007127567  0.002235302  0.030174850     *   2.273578
```

Now all the covariates are significant (they have the Z statistic far enough from zero and their confidence intervals don't contain 0), so we can keep the model:

$$\lambda \sim y + \text{Height} + Vegetation$$

whose fitted parameters have point estimates:

```
chosen.model = model1

coef(chosen.model)
```

```
## (Intercept)           y      Height  Vegetation
## -6.91203934 -0.01415209  0.16458833  0.01620508
```

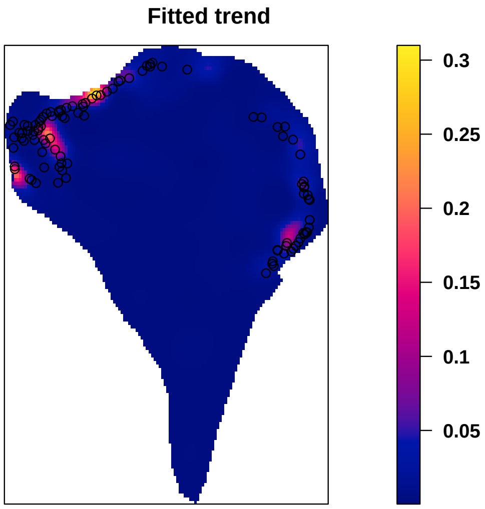
and 95% confidence interval:

```
confint(chosen.model)
```

```
##                     2.5 %       97.5 %
## (Intercept) -7.762450448 -6.061628232
## y           -0.024418771 -0.003885411
## Height       0.140804296  0.188372364
## Vegetation   0.002235302  0.030174850
```

The trend fitted by the model is:

```
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    1, 0))
plot(chosen.model, se = F, locations = islet_window)
```



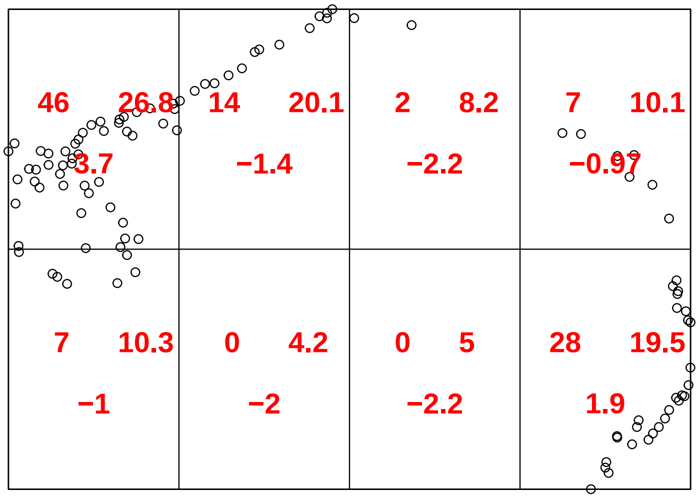
Now we assess the goodness of fit of the chosen model:

- via Chi-squared test

```
M <- quadrat.test(chosen.model, nx = 4, ny = 2)
M
```

```
##
##  Chi-squared test of fitted Poisson model 'chosen.model' using quadrat
##  counts
##
## data:  data from chosen.model
## X2 = 35.127, df = 4, p-value = 8.749e-07
## alternative hypothesis: two.sided
##
## Quadrats: 4 by 2 grid of tiles
```

```
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    0, 0))
plot(n84, main = "")
plot(M, add = TRUE, cex = 1.5, col = "red")
```



The p-value is very low, even though in almost all the subregions the standardized residuals have absolute value around 2 or smaller.

- via Kolmogorov-Smirnov test, to check the fitting of the model by each covariate separately.

```
KS1 = cdf.test(chosen.model, "y")
KS1
```

```
##
##  Spatial Kolmogorov-Smirnov test of inhomogeneous Poisson process in two
##  dimensions
##
## data:  covariate ’y’ evaluated at points of ’n84’
##      and transformed to uniform distribution under ’chosen.model’
## D = 0.09978, p-value = 0.2516
## alternative hypothesis: two-sided
```

```
KS2 = cdf.test(chosen.model, Vegetation)
KS2
```

```
##
##  Spatial Kolmogorov-Smirnov test of inhomogeneous Poisson process in two
##  dimensions
##
## data:  covariate ’Vegetation’ evaluated at points of ’n84’
##      and transformed to uniform distribution under ’chosen.model’
## D = 0.14381, p-value = 0.0271
## alternative hypothesis: two-sided
```

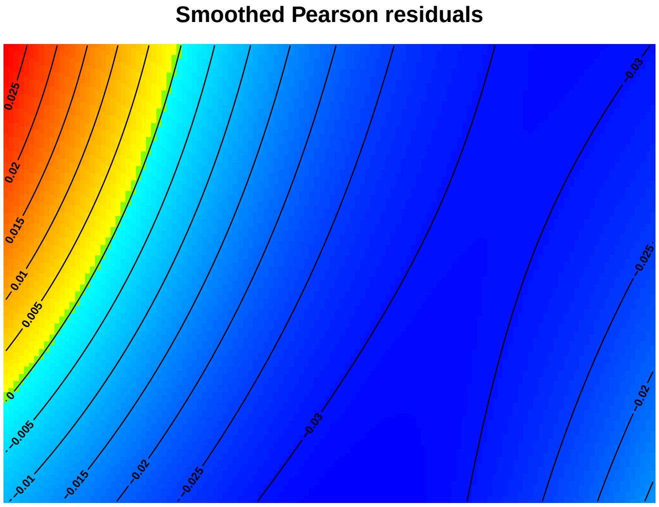
```
KS3 = cdf.test(chosen.model, Height)
KS3
```

```
##
##  Spatial Kolmogorov-Smirnov test of inhomogeneous Poisson process in two
##  dimensions
##
## data:  covariate 'Height' evaluated at points of 'n84'
##      and transformed to uniform distribution under 'chosen.model'
## D = 0.20742, p-value = 0.0002597
## alternative hypothesis: two-sided
```

The y variable seems to fits quite well the data, while Vegetation and Height have very little significance.

- looking at the Smoothed Pearson residuals

```
band = 40
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    1, 0))
smooth = diagnose.ppm(chosen.model, which = "smooth", type = "pearson", sigma = band)
```



```
smooth
```

```
## Model diagnostics (Pearson residuals)
## Diagnostics available:
##  smoothed residual field
## range of smoothed field =  [-0.03264, 0.02842]
## Null standard deviation of smoothed Pearson residual field: 0.007052
```

The standard deviation of the smoothed residuals under the null hypothesis of correct fitting is

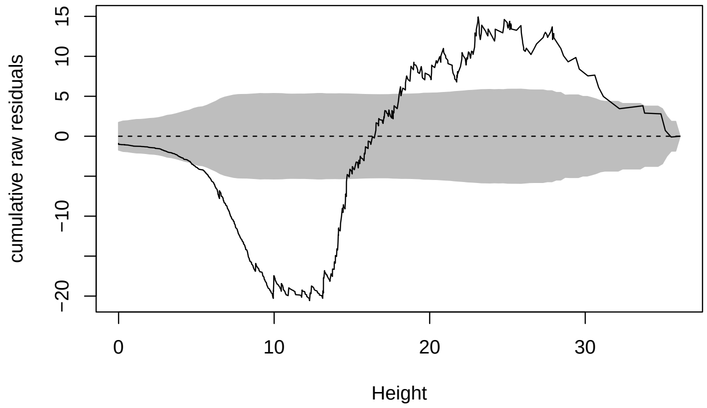
```
smooth$smooth$sdp
```

```
## [1] 0.00705237
```

so residuals whose modulus is larger than this value can be considered as extreme. This means that almost all subregions are not well fitted.

- through a Lurking plot

```
lurking(chosen.model, Height, type = "raw")
```



The residuals lie outside the envelope. The fit is not good:

- When the height is lower than 10 the residuals are consistently negative. Therefore, there are less points than expected with low height.
- From $height = 10$ to $height = 25$ the residuals are growing, so there are more points than expected.

After conducting these analyses, we can conclude that the model we have built, although it had the best fit from a statistical standpoint, does not actually fit the data well. Despite this, it was the most significant model based on the data we have.

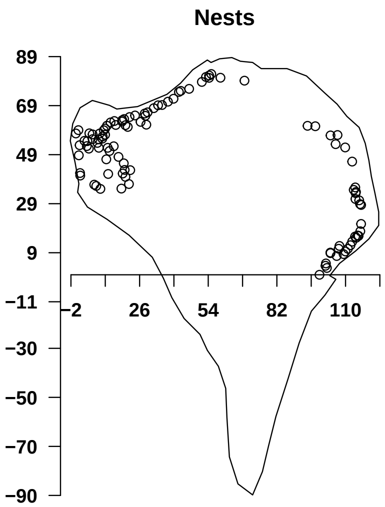
Task 3: Interaction

The nest data from islet "nucli 84" is stored in nucli84.txt. Additionally, the coordinates of the islet are in poly84.txt

1. Build a ppp object using the "nucli 84" data.

```
rm(list = ls())
nucli84 <- read.delim("T3/nucli84.txt")

min.X = min(nucli84$X)
min.Y = min(nucli84$Y)
max.X = max(nucli84$X)
max.Y = max(nucli84$Y)

nucli84$X = nucli84$X - min.X
nucli84$Y = nucli84$Y - min.Y

n84 = ppp(x = nucli84$X, y = nucli84$Y, range(nucli84$X), range(nucli84$Y))

poligon = read.delim("T3/poly84.txt")

poligon$X = poligon$X - min.X
poligon$Y = poligon$Y - min.Y
pol.illa <- list(x = poligon$X, y = poligon$Y)

min.pX = min(poligon$X)
min.pY = min(poligon$Y)
max.pX = max(poligon$X)
max.pY = max(poligon$Y)

n84p = ppp(nucli84$X, nucli84$Y, poly = pol.illa, range(poligon$X), range(poligon$Y))
```

```
par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    2, 0))
plot(n84p, main = "Nests")
axis(1, at = c(round(seq(min.pX, max.pX, length = 10), digits = 0)), pos = c(0, 0))
axis(2, at = c(round(seq(min.pY, max.pY, length = 10), digits = 0)), pos = c(min.pX -
    4, min.pY - 50))
```



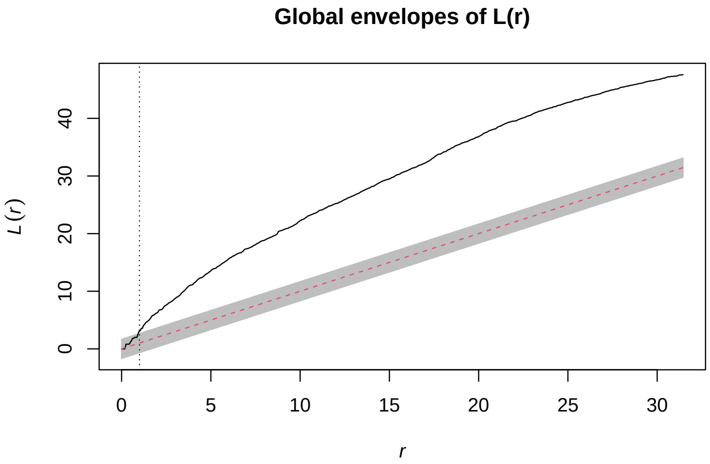
2) Check the interaction pattern.

To check the interaction pattern, we utilize the L-function defined as $L(r) = \sqrt{\frac{K(r)}{\pi}}$ as it is an estimator with a variance that does not vary with respect to r. This makes it more stable than other estimators, such as F, G, and K.

```
E <- envelope(n84p, Lest, nsim = 19, rank = 1, global = TRUE, correction = "best")
```

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

```
plot(E, main = "Global envelopes of L(r)", legend = F)
abline(v = 1, lty = 3)
```



The observed $L(r)$ goes above the theoretical function, indicating that the observed distance between locations is greater than expected under the assumption of a homogeneous Poisson process. This suggests the presence of a regular pattern. Additionally, we notice that the observed line falls outside of the envelopes from $r = 1$ onward.

3) Model the relation between the intensity of the point process and the covariates height and vegetation accounting for the interaction pattern. Interpret the estimates.

Since we have a regular pattern, we will use Gibbs models.

First we try to fit a Hardcore model:

```
grid <- read.delim("T2/grid.txt")
grid.veg = read.delim("T2/grid_veg.txt")
veg = as.matrix(read.delim("T2/veg.txt", header = FALSE))
height = as.matrix(read.delim("T2/height.txt", header = FALSE))

Height = im(mat = height, xcol = grid$x, yrow = grid$y)
Vegetation = im(mat = veg, xcol = grid.veg$x, yrow = grid.veg$y)

fitH <- ppm(n84p, ~Height + Vegetation, Hardcore)
```

```
summary(fitH)
```

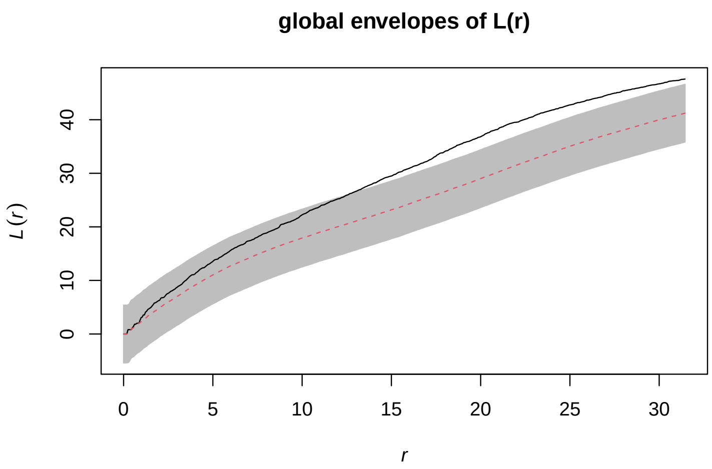
```
## Point process model
## Fitting method: maximum pseudolikelihood (Berman-Turner approximation)
## Model was fitted using glm()
## Algorithm converged
## Call:
## ppm.ppp(Q = n84p, trend = ~Height + Vegetation, interaction = Hardcore)
## Edge correction: "border"
##  [border correction distance r = 0.241749245771188 ]
## --------------------------------------------------------------------------------
## Quadrature scheme (Berman-Turner) = data + dummy + weights
##
## Data pattern:
## Planar point pattern:  104 points
## Average intensity 0.00926 points per square unit
## Window: polygonal boundary
## single connected closed polygon with 60 vertices
## enclosing rectangle: [-2.26699, 123.5272] x [-89.78583, 88.59802] units
##                       (125.8 x 178.4 units)
## Window area = 11231.3 square units
## Fraction of frame area: 0.501
##
## Dummy quadrature points:
##      32 x 32 grid of dummy points, plus 4 corner points
##      dummy spacing: 3.931068 x 5.574495 units
##
## Original dummy parameters: =
## Planar point pattern:  560 points
## Average intensity 0.0499 points per square unit
## Window: polygonal boundary
## single connected closed polygon with 60 vertices
## enclosing rectangle: [-2.26699, 123.5272] x [-89.78583, 88.59802] units
##                       (125.8 x 178.4 units)
## Window area = 11231.3 square units
## Fraction of frame area: 0.501
## Quadrature weights:
##      (counting weights based on 32 x 32 array of rectangular tiles)
## All weights:
##  range: [0.71, 21.9] total: 11200
## Weights on data points:
##  range: [3.1, 11]    total: 711
## Weights on dummy points:
##  range: [0.71, 21.9] total: 10500
## --------------------------------------------------------------------------------
## FITTED :
##
## Nonstationary Hard core process
##
## ---- Trend: ----
##
## Log trend: ~Height + Vegetation
## Model depends on external covariates 'Height' and 'Vegetation'
## Covariates provided:
##  Height: im
##  Vegetation: im
##
## Fitted trend coefficients:
## (Intercept)      Height  Vegetation
## -7.54677892  0.15928955  0.01532893
##
##                Estimate       S.E.      CI95.lo     CI95.hi Ztest      Zval
## (Intercept) -7.54677892 0.75595463 -9.028422773 -6.06513506   *** -9.983111
## Height       0.15928955 0.01722086  0.125537275  0.19304182   ***  9.249800
## Vegetation   0.01532893 0.01097206 -0.006175909  0.03683378        1.397088
##
##  ---- Interaction: -----
##
## Interaction: Hard core process
## Hard core distance:  0.2417492
##
## ----------- gory details -----
##
## Fitted regular parameters (theta):
## (Intercept)      Height  Vegetation
## -7.54677892  0.15928955  0.01532893
##
## Fitted exp(theta):
##  (Intercept)       Height   Vegetation
## 0.0005278075 1.1726774457 1.0154470245
```

We assess the goodness of fit of the model by looking at the theoretical envelopes of $L(r)$ and comparing it with the observed values.

```
E <- envelope(fitH, Lest, nsim = 19, rank = 1, global = TRUE, correction = "best")
```

```
## Generating 38 simulated realisations of fitted Gibbs model (19 to estimate the
## mean and 19 to calculate envelopes) ...
## 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, 2
##
## Done.
```

```
plot(E, main = "global envelopes of L(r)", legend = F)
```



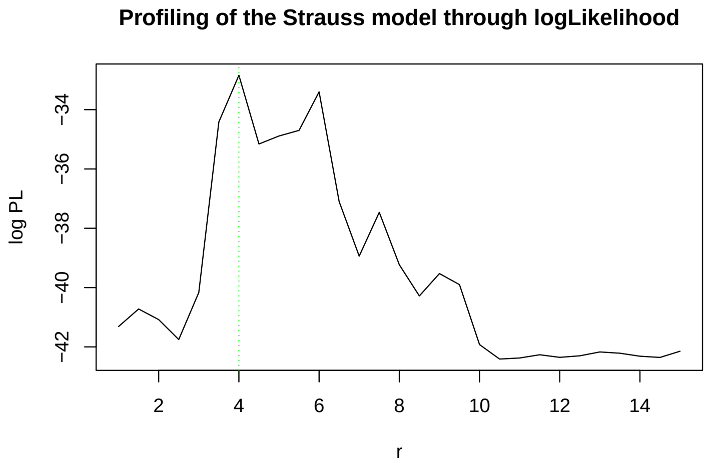
The observed function falls outside of the envelopes, indicating that the Hardcore model does not fit the data well.

As an alternative, we will try using a Strauss model. Our exploratory analysis showed that the observed L-function lies outside the envelope for values of r greater than 1. To determine the optimal value for r, we will analyze the profile likelihood using a range of values inside the interval $[1, 15]$ and choose the one that provides the best fit to the data.

```
df = data.frame(r = seq(1, 15, by = 0.5))
pfit = profilepl(df, Strauss, n84p, ~Height + Vegetation)
```

```
## 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, 2
```

```
plot(pfit, main = "Profiling of the Strauss model through logLikelihood")
```



```
r.opt = pfit$fit$fitin$interaction$par$r
```

The optimal value is reached at $r = 4$.

So now we can finally estimate the model:

```
fitSt <- ppm(n84p, ~Height + Vegetation, Strauss(r = r.opt))
fitSt
```

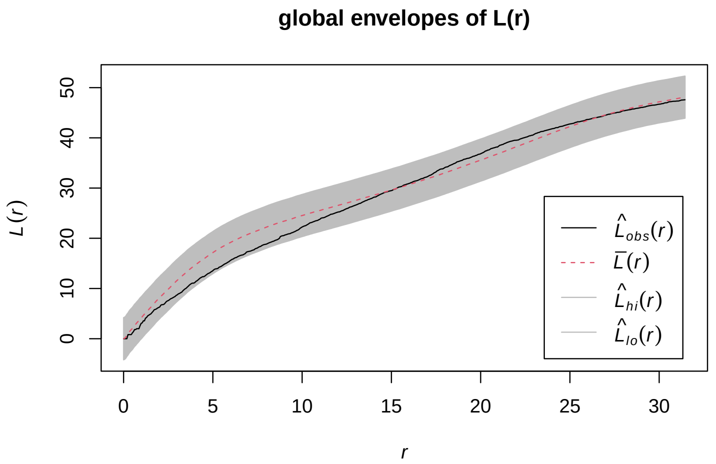
```
## Nonstationary Strauss process
##
## Log trend:  ~Height + Vegetation
##
## Fitted trend coefficients:
## (Intercept)      Height  Vegetation
## -8.50843012  0.10802394  0.03107911
##
## Interaction distance:    4
## Fitted interaction parameter gamma:  1.5200358
##
## Relevant coefficients:
## Interaction
##   0.4187339
##
## For standard errors, type coef(summary(x))
##
## *** Model is not valid ***
## *** Interaction parameters are outside valid range ***
```

and assess his validity:

```
E <- envelope(fitSt, Lest, nsim = 19, rank = 1, global = TRUE, correction = "best")
```

```
## Generating 38 simulated realisations of fitted Gibbs model (19 to estimate the
## mean and 19 to calculate envelopes) ...
## 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, 2
##
## Done.
```

```
plot(E, main = "global envelopes of L(r)")
```



The observed L-function is inside the envelopes, therefore the Strauss model can be valid and we proceed to interpret it.

```
summary(fitSt, fine = F)
```

```
## Point process model
## Fitting method: maximum pseudolikelihood (Berman-Turner approximation)
## Model was fitted using glm()
## Algorithm converged
## Call:
## ppm.ppp(Q = n84p, trend = ~Height + Vegetation, interaction = Strauss(r = r.opt))
## Edge correction: "border"
##  [border correction distance r = 4 ]
## --------------------------------------------------------------------------------
## Quadrature scheme (Berman-Turner) = data + dummy + weights
##
## Data pattern:
## Planar point pattern:  104 points
## Average intensity 0.00926 points per square unit
## Window: polygonal boundary
## single connected closed polygon with 60 vertices
## enclosing rectangle: [-2.26699, 123.5272] x [-89.78583, 88.59802] units
##                      (125.8 x 178.4 units)
## Window area = 11231.3 square units
## Fraction of frame area: 0.501
##
## Dummy quadrature points:
##      32 x 32 grid of dummy points, plus 4 corner points
##      dummy spacing: 3.931068 x 5.574495 units
```

```
##
## Original dummy parameters: =
## Planar point pattern:  560 points
## Average intensity 0.0499 points per square unit
## Window: polygonal boundary
## single connected closed polygon with 60 vertices
## enclosing rectangle: [-2.26699, 123.5272] x [-89.78583, 88.59802] units
##                      (125.8 x 178.4 units)
## Window area = 11231.3 square units
## Fraction of frame area: 0.501
## Quadrature weights:
##      (counting weights based on 32 x 32 array of rectangular tiles)
## All weights:
##  range: [0.71, 21.9] total: 11200
## Weights on data points:
##  range: [3.1, 11]    total: 711
## Weights on dummy points:
##  range: [0.71, 21.9] total: 10500
## --------------------------------------------------------------------------------
## FITTED :
##
## Nonstationary Strauss process
##
## ---- Trend: ----
##
## Log trend: ~Height + Vegetation
## Model depends on external covariates 'Height' and 'Vegetation'
## Covariates provided:
##  Height: im
##  Vegetation: im
##
## Fitted trend coefficients:
## (Intercept)      Height  Vegetation
## -8.50843012  0.10802394  0.03107911
##
##                Estimate       S.E.       CI95.lo     CI95.hi Ztest       Zval
## (Intercept) -8.50843012 0.84655940 -10.167656050 -6.84920419   *** -10.050600
## Height       0.10802394 0.01884424   0.071089902  0.14495798   ***   5.732463
## Vegetation   0.03107911 0.01163310   0.008278644  0.05387957    **   2.671609
## Interaction  0.41873389 0.09133454   0.239721476  0.59774630   ***   4.584617
##
##  ---- Interaction: -----
##
## Interaction: Strauss process
## Interaction distance:    4
## Fitted interaction parameter gamma:  1.5200358
##
## Relevant coefficients:
## Interaction
##   0.4187339
##
## ----------- gory details -----
##
## Fitted regular parameters (theta):
```

```
## (Intercept)      Height  Vegetation Interaction
## -8.50843012  0.10802394  0.03107911  0.41873389
##
## Fitted exp(theta):
##  (Intercept)       Height   Vegetation  Interaction
## 0.0002017603 1.1140744182 1.0315671056 1.5200358028
##
## *** Model is not valid ***
## *** Interaction parameters are outside valid range ***
```

The estimates of its coefficients are:

```
fitSt$coef
```

```
## (Intercept)      Height  Vegetation Interaction
## -8.50843012  0.10802394  0.03107911  0.41873389
```

The estimate for the interaction parameter is $\gamma = 0.4187339$. The parameter γ controls the strength of interaction between points. If $\gamma = 1$ the model reduces to a Poisson process. If $\gamma = 0$ the model is a hard core process. For values $0 < \gamma < 1$, like in our case, the process exhibits a moderate strength of interaction between the presence of points in different locations.

TASK 4: Case-control studies

Using the Primary Biliary Cirrhosis data, perform a case-control point pattern analysis. Point locations and marks are stored in *PBCdata.txt” file. Coordinates of the window are in “PBCpoly.txt” file.

The data are:

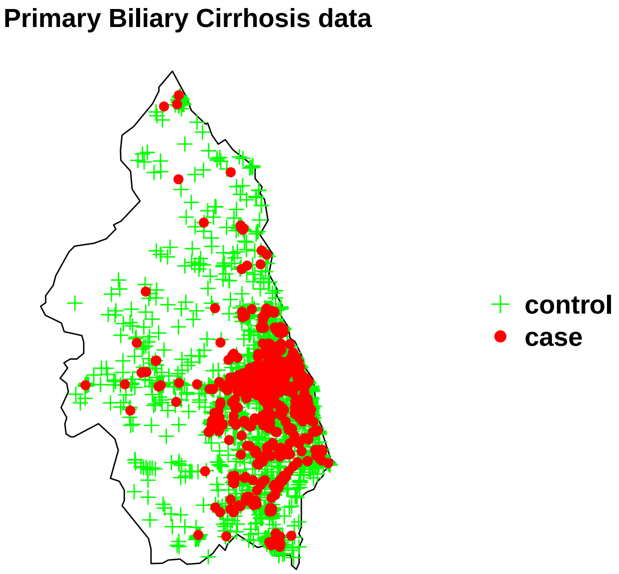
```
library(spatstat)
rm(list = ls())

pbc.data <- read.delim("T4/PBCdata2.txt")
head(pbc.data)
```

```
##       x     y marks
## 1 43830 56150  case
## 2 42840 56510  case
## 3 43740 54320  case
## 4 42480 56240  case
## 5 42280 56930  case
## 6 42610 55850  case
```

1) Create the ppp object.

```
poligon = read.delim("T4/PBCpoly.txt")
min.pX = min(poligon$x)
max.pX = max(poligon$x)
min.pY = min(poligon$y)
max.pY = max(poligon$y)

pol.illa <- list(x = poligon$x, y = poligon$y)

pbc.data$marks = factor(pbc.data$marks)

pbc.p = ppp(pbc.data$x, pbc.data$y, poly = pol.illa, range(poligon$x), range(poligon$y),
    marks = pbc.data$marks)
```

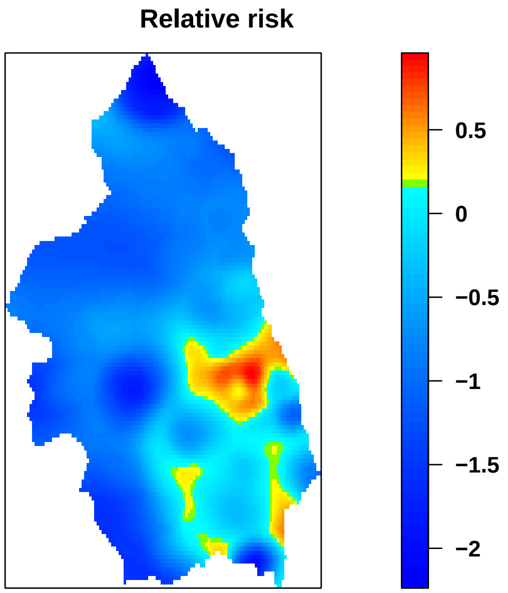
```
cases = split(pbc.p, pbc.data$marks)$case
controls = split(pbc.p, pbc.data$marks)$control

par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    2.5, 0))
plot(pbc.p$window, main = "Primary Biliary Cirrhosis data")
points(controls, pch = 3, col = "green")
points(cases, pch = 16, col = "red")
legend("right", c("control", "case"), pch = c(3, 16), col = c("green", "red"), bty = "n",
    cex = 1.2)
```



2) Assess the spatial variation of the risk

We plot the relative risk in logarithmic scale, so the locations colored in red have a Relative risk greater than 1, while locations colored in blue have a Relative risk smaller than 1.

```
chp <- risk(cases, controls, adapt = T)

M = max(chp$rr)
m = min(chp$rr)

par(mfrow = c(1, 1), font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(0, 0,
    1, 0))
plot(chp$rr, gamma = 1.3, main = "Relative risk", col = beachcolours(c(m, M)))
```



Let's use a permutation test to see if there is a significant difference between cases and controls risk, computing on a regular grid the statistic:

$$\hat{T} = |c| \sum_{i=1}^{p} (\hat{\rho}(s_i) - \hat{\rho}_0)^2$$

where c is the cell of the grid and (under the null hypothesis H_0) $\rho_0 = 0$.

```
cellsize <- chp$rr$xstep * chp$rr$xstep
rho0 <- 0
```

The value of the statistic is:

```
ratiorho <- cellsize * sum((chp$rr$v - rho0)^2, na.rm = T)
ratiorho
```

```
## [1] 35687966
```

```
# Permutation function
perm_rr <- function() {
    new_ch <- rlabel(pbc.p)

    num = length(pbc.p$x)
    indices = sample(1:num)

    new_cases <- split(new_ch, f = pbc.p$marks[indices])$case
    new_controls <- split(new_ch, f = pbc.p$marks[indices])$control
    new_chp <- risk(new_cases, new_controls)
    cellsize <- new_chp$rr$xstep * new_chp$rr$ystep
    ratio_perm <- cellsize * sum((new_chp$rr$v - rho0)^2, na.rm = T)
    ratio_perm
}
```

And its p-value (obtained through a permutation approach) is:

```
(sum(rperm > ratiorho) + 1)/(nsim + 1)
```

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

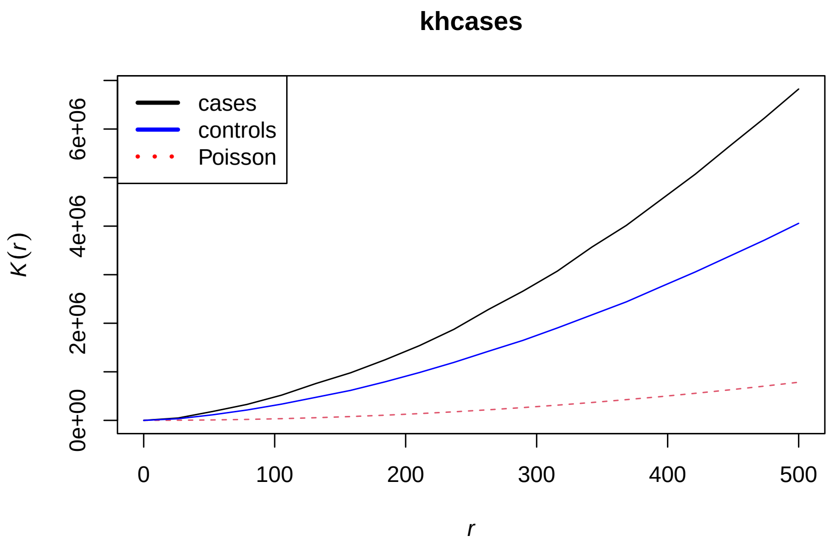
So we don't have enough evidence to reject the null hypothesis.

3) Compare the interaction patterns.

We draw the K-functions for cases and controls:

```
s = seq(0, 500, length = 20)
khcases <- Kest(cases, r = s, correction = "best")
khcontrols <- Kest(controls, r = s, correction = "best")
```

```
plot(khcases, legend = F)
lines(khcontrols$r, khcontrols$iso, lty = 1, col = "blue")
legend("topleft", legend = c("cases", "controls", "Poisson"), lty = c(1, 1, 3), col = c("black",
    "blue", "red"), lwd = 3)
```

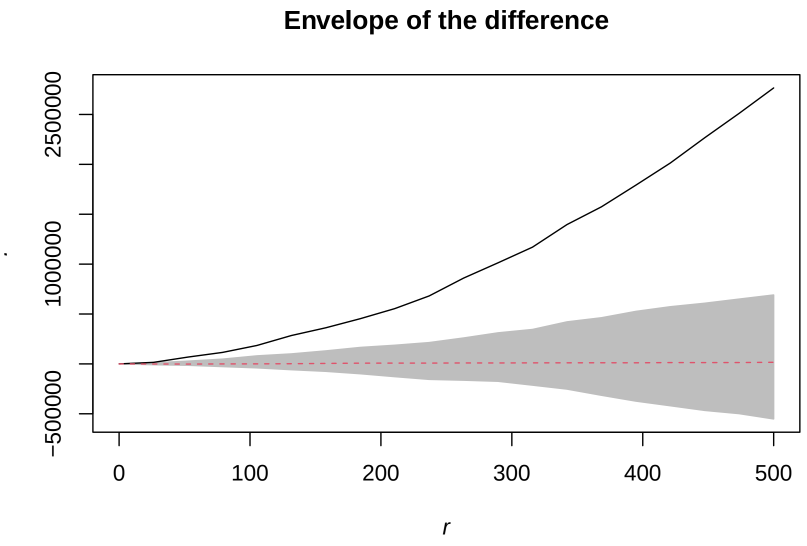


The K-functions for both cases and controls are above the Poisson curve, indicating a cluster interaction. To test the null hypothesis that the two K-functions are equal, we will use a permutational approach.

```
Kdif <- function(X, r, cr = "iso") {
    k1 <- Kest(X[marks(X) == "case"], r = r, correction = cr)
    k2 <- Kest(X[marks(X) == "control"], r = r, correction = cr)
    D = k1[[cr]] - k2[[cr]]
    res <- data.frame(r = r, D = D)
    return(fv(res, valu = "D", fname = "D"))
}

nsim <- 39
envKdif <- envelope(pbc.p, Kdif, r = s, nsim = nsim, savefuns = TRUE, simulate = expression(rlabel(pbc.p
```

```
## Generating 39 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, 2
##
## Done.
```

```
plot(envKdif, legend = F, main = "Envelope of the difference")
```



As shown in the plot, the difference between the K-functions falls outside the envelopes for all ranges, indicating that the difference is not significant. The p-value for this test is as follows:

```
simfuns <- as.data.frame(attr(envKdif, "simfuns"))[, -1]
khcovdiag <- apply(simfuns, 1, var)
T0 <- sum(((khcases$iso - khcontrols$iso)/sqrt(khcovdiag))[-1])
T_pm <- apply(simfuns, 2, function(X) {
    sum((X/sqrt(khcovdiag))[-1])
})
pvalue <- 2 * (sum(abs(T_pm) > abs(T0)) + 1)/(nsim + 1)
pvalue
```

```
## [1] 0.05
```

Therefore, we can reject the null hypothesis at a significance level of 95% and conclude that the interaction pattern differs between cases and controls.