

# Spatial Point Pattern Analysis with tree detections generated from LiDAR data

Jakob Danel

2024-02-01

## Contents

<b>1</b>	<b>Overview</b>	<b>1</b>
<b>2</b>	<b>Data</b>	<b>3</b>
<b>3</b>	<b>Research Question and Hypothesis</b>	<b>3</b>
3.1	Hypothesis . . . . .	3
<b>4</b>	<b>Methods</b>	<b>4</b>
4.1	Workflow . . . . .	4
4.2	Total Comparisions . . . . .	4
<b>5</b>	<b>Example workflow</b>	<b>4</b>
5.1	Creating <code>spatstat::ppp</code> object from <code>geopackage</code> with detections and AOI (with respect to reference systems) . . . . .	4
5.2	Select to samples and look if their upper the Monte-Carlo band or not. . . . .	6
5.3	Plot the samples bands . . . . .	7
5.4	Calculate the MC Nemar test for both bands . . . . .	9

## 1 Overview

In this research investigation, our objective is to assess the utility of spatial Point Patterns in monocultural forest tree detection as a parameter for distinguishing between different species within monocultural forest areas. The study focuses on four prevalent species in the NRW, DE region, namely beech, oak, pine, and spruce, based on their common occurrence (Blickensdorfer 2022). Three areas exhibiting predominantly monocultural tree structures were selected for each species. Utilizing LiDAR data for the designated areas, a tree detection algorithm

was applied through `lidR::locate_trees` with default settings, as outlined in (Roussel and Auty 2024). The resulting detection data (exemplified in Figure 1) serve as the foundation for examining point patterns within these areas. Specifically, the investigation delves into the behavior of Ripley’s reduced second moment function  $K(r)$  (Ripley 1977) and the distribution of ‘r values beyond a significance threshold, determined through Monte Carlo simulations—a widely used technique in addressing spatial problems (Zhou et al. 2003).

For details on data acquisition and preprocessing, please refer to (Danel and Bruch 2024).

```
library(ggplot2)
sf::sf_use_s2(FALSE)
```

Spherical geometry (s2) switched off

```
areas <- sf::read_sf("research_areas.shp")
detections <- sf::read_sf("detection.gpkg")

sample_area <- areas[1,]
sample_specie <- sample_area$species
sample_name <- sample_area$name
sample_detections <- detections |> dplyr::filter(specie == sample_specie & area == sample_

ggplot() + geom_sf(data = sample_area) + geom_sf(data = sample_detections)
```

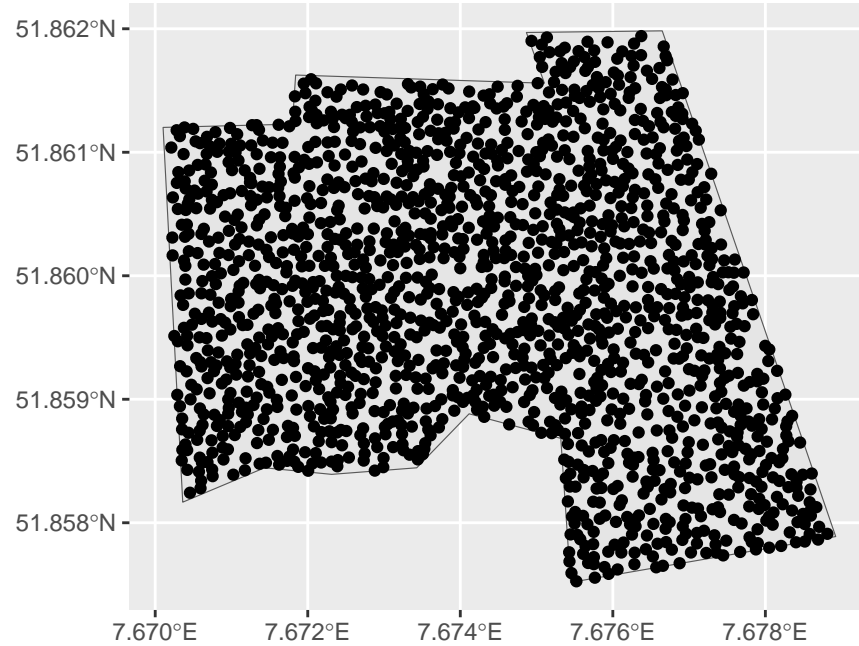


Figure 1: All tree detections with the corresponding area of interest for an sample area in Bielefeld-Brackewede (with dominantn specie beech).

## 2 Data

1. Detections of trees based on LiDAR in particular areas of interest: `detection.gpkg`
2. Areas of interests where detection-algorithm was used: `research_areas.shp`
3. Data frame mapping from area to specie: `patches.csv`

## 3 Research Question and Hypothesis

Is the comparison of the probability that the curve of Ripleys K function is under or above a significance band created via Monte-Carlo simulation a valid method to distinguish between the point patterns of trees for forests with monocultural vegetation for sample areas with the most common species of NRW?

### 3.1 Hypothesis

1. Between species the probability that the  $K(r)$  value is over the Monte-Carlo band significantly differs.

2. Between species the probability that the  $K(r)$  value is under the Monte-Carlo band significantly differs.
3. Between samples of the same species the probability that the  $K(r)$  value is over the Monte-Carlo band is not significantly different.
4. Between samples of the same species the probability that the  $K(r)$  value is under the Monte-Carlo band is not significantly different.

## 4 Methods

### 4.1 Workflow

- Create Monte-Carlo simulations of each area.
- Calculate if the values for different  $r$  are above (under) the envelope or not (boolean)
- Use a McNemar-Test to understand if there is difference in the samples:
  - $H_0$ : The probability that a value is above/under or inside the envelope is for both samples the same:
  - $H_A$ : The probability that a value is above/under or inside the envelope significantly differs from each other.
- Calculate this test statistics pairwise between all samples from the same species, to see if the samples are representative for their species.
- Calculate pairwise McNemar tests for each pair of species.

### 4.2 Total Comparisons

- Specie intern: 3 Lower, 3 Higher (4 comparisons total)
- Specie extern: 9 Lower, 9 Higher (6 comparisons total)

## 5 Example workflow

### 5.1 Creating `spatstat::ppp` object from `geopackage` with detections and AOI (with respect to reference systems)

```
library(spatstat)
```

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

```
Loading required package: spatstat.geom

spatstat.geom 3.2-7

Loading required package: spatstat.random

spatstat.random 3.2-1

Loading required package: spatstat.explore

Loading required package: nlme

spatstat.explore 3.2-5

Loading required package: spatstat.model

Loading required package: rpart

spatstat.model 3.2-8

Loading required package: spatstat.linnet

spatstat.linnet 3.1-3

spatstat 3.0-7
For an introduction to spatstat, type 'beginner'
```

```
detections <- sf::st_read("detection.gpkg")
```

```
Reading layer `detection' from data source
  `/home/jakob/gi-master/project-courses/lidar-forest-analysis/src/aosd/detection.gpkg'
  using driver `GPKG'
Simple feature collection with 28839 features and 4 fields
Geometry type: POINT
Dimension:      XY
Bounding box:   xmin: 365546.3 ymin: 5660096 xmax: 467996.9 ymax: 5790950
Projected CRS: ETRS89 / UTM zone 32N
```

```

create_ppp <-
  function(species_identifier,
           area_identifier,
           det = detections) {
    trees <-
      det |> dplyr::filter(specie == species_identifier &
                          area == area_identifier)

    area <-
      sf::read_sf("research_areas.shp") |> dplyr::filter(species == species_identifier &
                                                          name == area_identifier) |> sf::
      sf::st_union(sf::st_geometry(area)) |> spatstat.geom::as.owin() -> w
    pp <- sf::st_geometry(trees) |> spatstat.geom::as.ppp(W = w)
    return(pp)
  }

patches <- read.csv2(file = "patches.csv")

pps <- list()
for (i in 1:nrow(patches)) {
  area <- patches[i, "area"]
  specie <- patches[i, "specie"]
  pps[[i]] <- create_ppp(specie, area, detections)
}

```

## 5.2 Select to samples and look if their upper the Monte-Carlo band or not.

```

# Fixed r values for all Kest functions calculated

rng <- seq(0, 100, by = .1)

is_upper <- function(envelope) {
  return(envelope$hi < envelope$obs)
}

env <- envelope(pps[[1]], Kest, r = rng)

```

Generating 99 simulations of CSR ...

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

Done.

```
upper <- is_upper(env)  
  
env2 <- envelope(pps[[4]], Kest, r = rng)
```

Generating 99 simulations of CSR ...

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

Done.

```
upper2 <- is_upper(env2)
```

### 5.3 Plot the samples bands

```
plot(env)
```

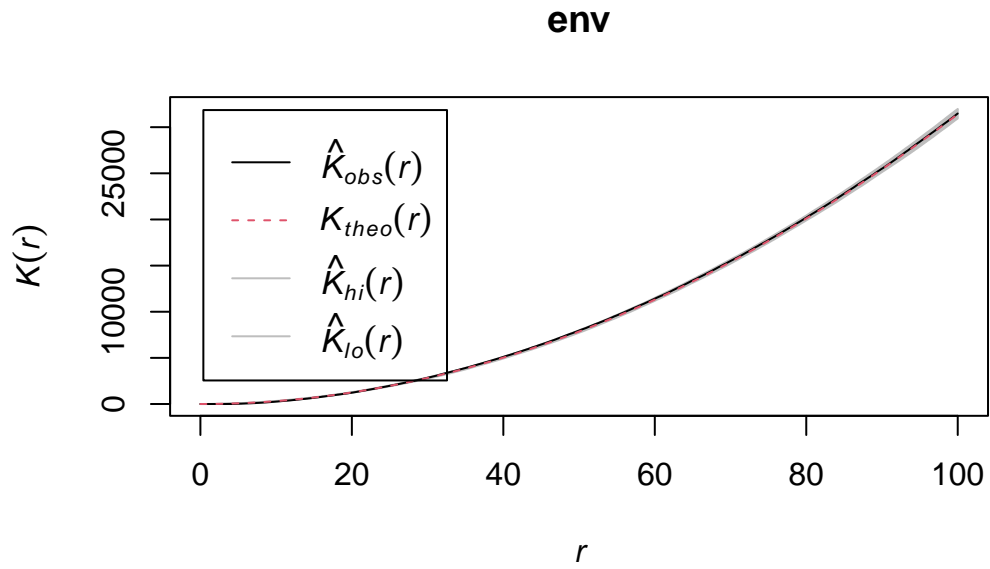


Figure 2: Monte-Carlo Simulation around the  $K(r)$  function of the point pattern in Bielefeld Brackwede (Beech)

```
plot(env2)
```

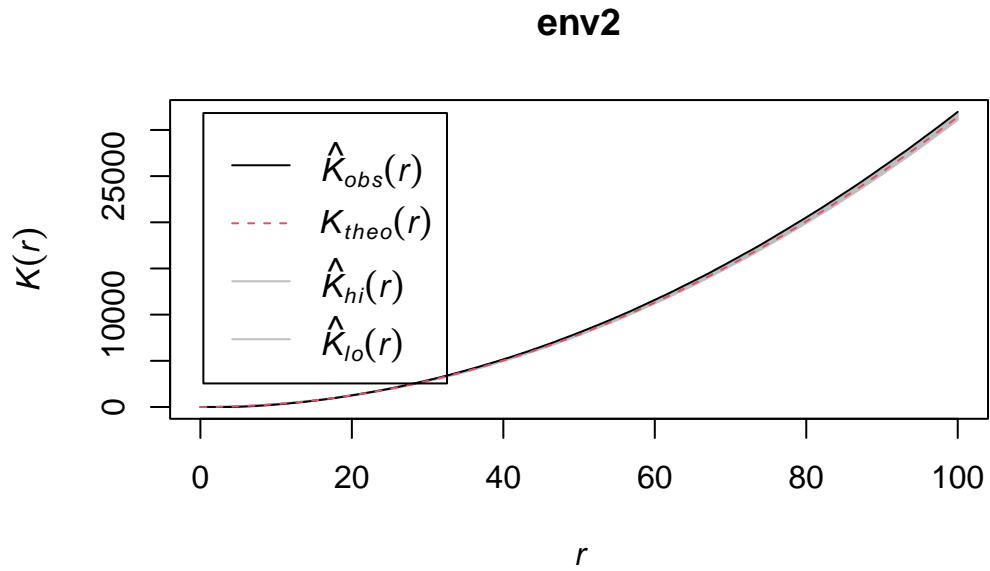


Figure 3: Monte-Carlo Simulation around the  $K(r)$  function of the point pattern in Hamm (Oak)



## 5.4 Calculate the MC Nemar test for both bands

```
calculate_mcnemar_test <- function(vector1, vector2) {  
  # Check if vectors have the same length  
  if (length(vector1) != length(vector2)) {  
    stop("Vectors must have the same length.")  
  }  
  
  # Ensure vectors are logical  
  if (!all(is.logical(vector1)) || !all(is.logical(vector2))) {  
    stop("Vectors must contain only boolean values.")  
  }  
  
  tbl <- table(vector1, vector2)  
  print(tbl)  
  # Perform McNemar test  
  result <- mcnemar.test(tbl)  
  return(result)  
}  
  
calculate_mcnemar_test(upper, upper2)
```

```
      vector2  
vector1 FALSE TRUE  
  FALSE   334  654  
  TRUE     0   13
```

McNemar's Chi-squared test with continuity correction

```
data:  tbl  
McNemar's chi-squared = 652, df = 1, p-value < 2.2e-16
```

- Blickensdoerfer, Lukas. 2022. "Dominant Tree Species for Germany (2017/2018)." *Walddatlas-Wald Und Waldnutzung*. Thünen Atlas. [https://atlas.thuenen.de/layers/geonode:Dominant\\_Species\\_Class](https://atlas.thuenen.de/layers/geonode:Dominant_Species_Class).
- Danel, Jakob, and Frederick Bruch. 2024. *Lfa: LiDAR Forest Analysis: Diversity of Tree Species in an Oecosystem*. <https://gtihub.com/jakobdanel/lidar-forest-analysis>.
- Ripley, Brian D. 1977. "Modelling Spatial Patterns." *Journal of the Royal Statistical Society: Series B (Methodological)* 39 (2): 172–92.
- Roussel, Jean-Romain, and David Auty. 2024. *Airborne LiDAR Data Manipulation and Visualization for Forestry Applications*. <https://cran.r-project.org/package=lidR>.

Zhou, G, T Esaki, Y Mitani, M Xie, and J Mori. 2003. "Spatial Probabilistic Modeling of Slope Failure Using an Integrated GIS Monte Carlo Simulation Approach." *Engineering Geology* 68 (3-4): 373-86.