Point Processes 1: Spatial Intensity

Michael Noonan

DATA 589: Spatial Statistics

Table of contents

Review

Review

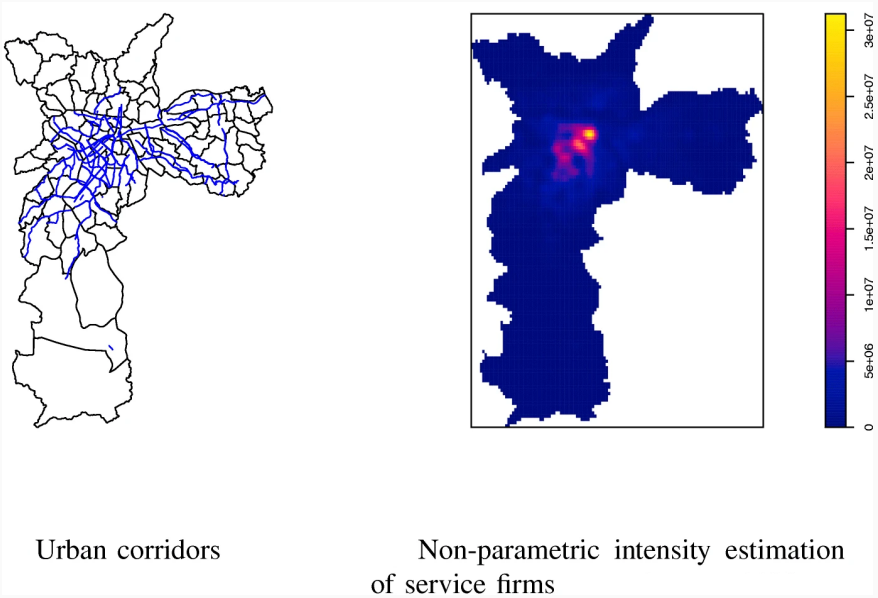
Last lecture we covered the big picture value of spatial data, and touched on the different types of spatial data (points vs. spatial measurements).

This lecture we will explore the concept of 'point' data and 'point' processes, and learn some basic descriptive statistics for describing them.

Applied Points Pattern Analysis

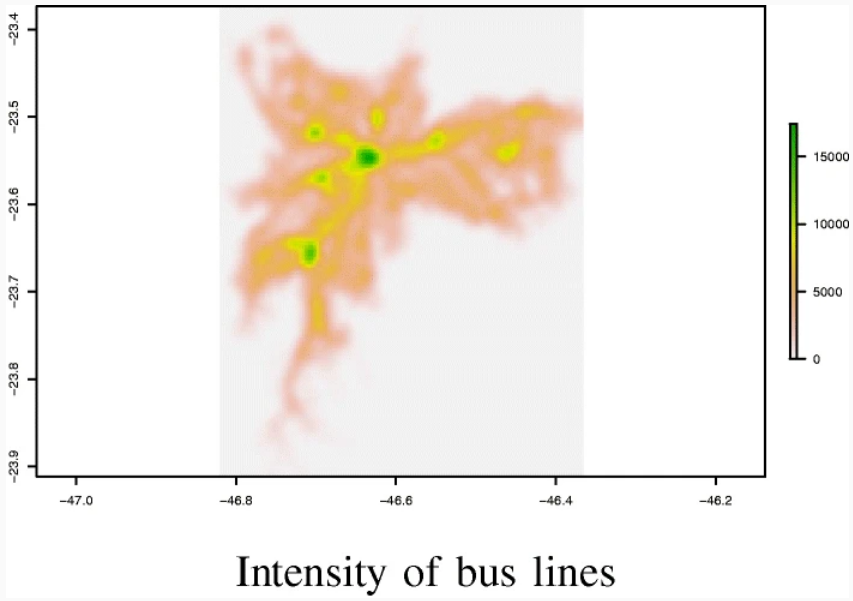
Urban Planning

Morales & Laurini (2021) used point pattern analyses to model the location patterns of new firms in the city of São Paulo.



Urban corridors

Non-parametric intensity estimation of service firms



Intensity of bus lines

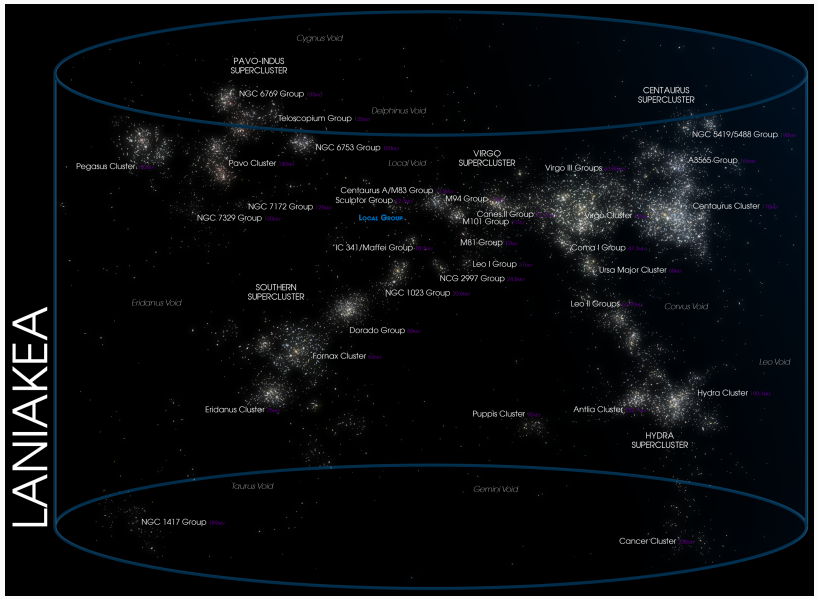
Used these models to understand what urban features lead to development, but this could also be used predict outcomes from e.g., adding new bus lines.

Points Patterns

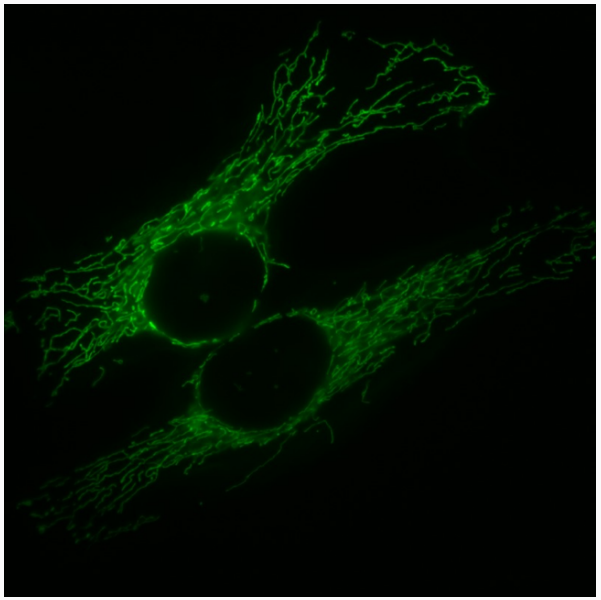
Points as data

A spatial point pattern is a dataset comprised of the locations of 'things' or 'events'.

Galaxies in Laniakea Supercluster



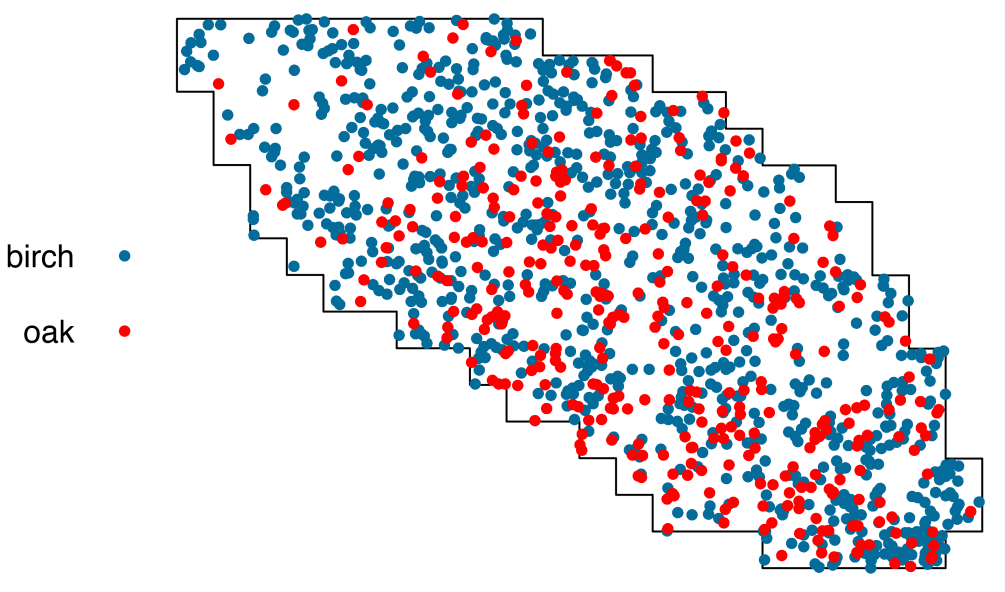
Mitochondria in a cell



Trees in a forest, locations of road traffic accidents, crimes, incidents of diseases, etc...

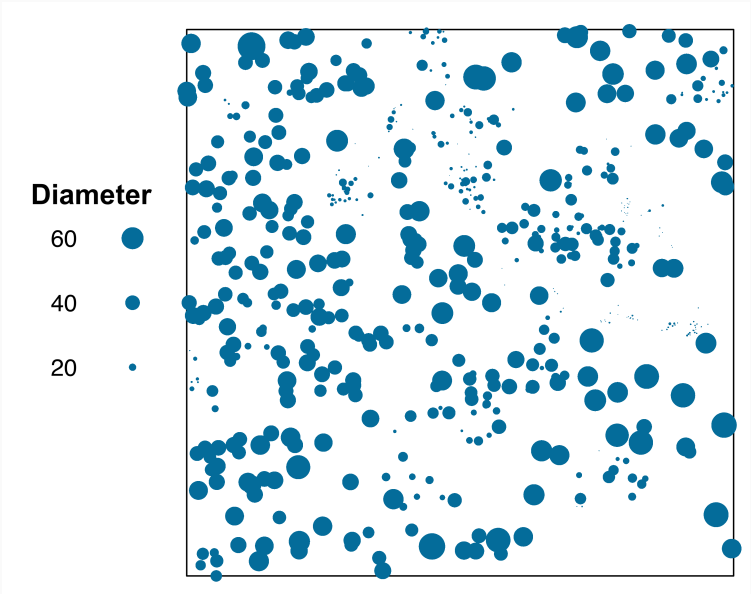
Marked points

Sometimes we have points of several types



Source: `spatstat` package

...or a marked point pattern (i.e., auxiliary information).

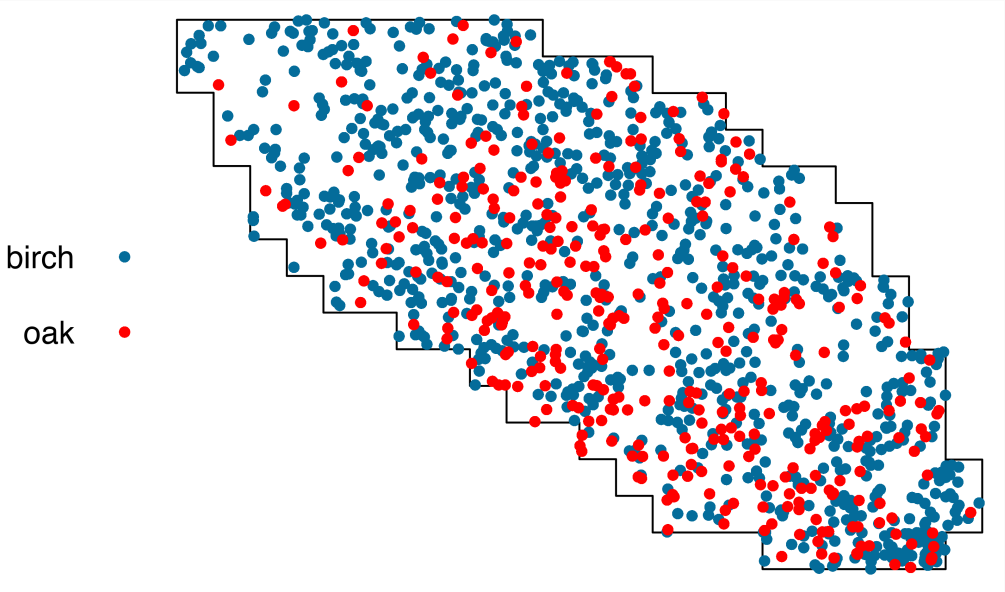


Source: `spatstat` package

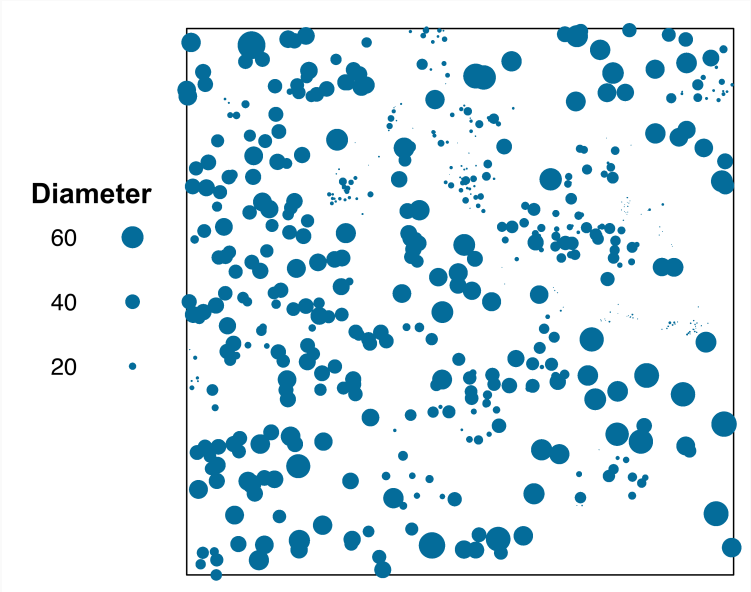
Sampling window

Point processes always need to be accompanied by information on the sampling 'window' (critical).

Without the sampling window we can't estimate metrics correctly.



Source: spatstat package



Source: spatstat package

Sampling window cont.

Without the sampling window we can't estimate metrics correctly.

Are these trees clustered?

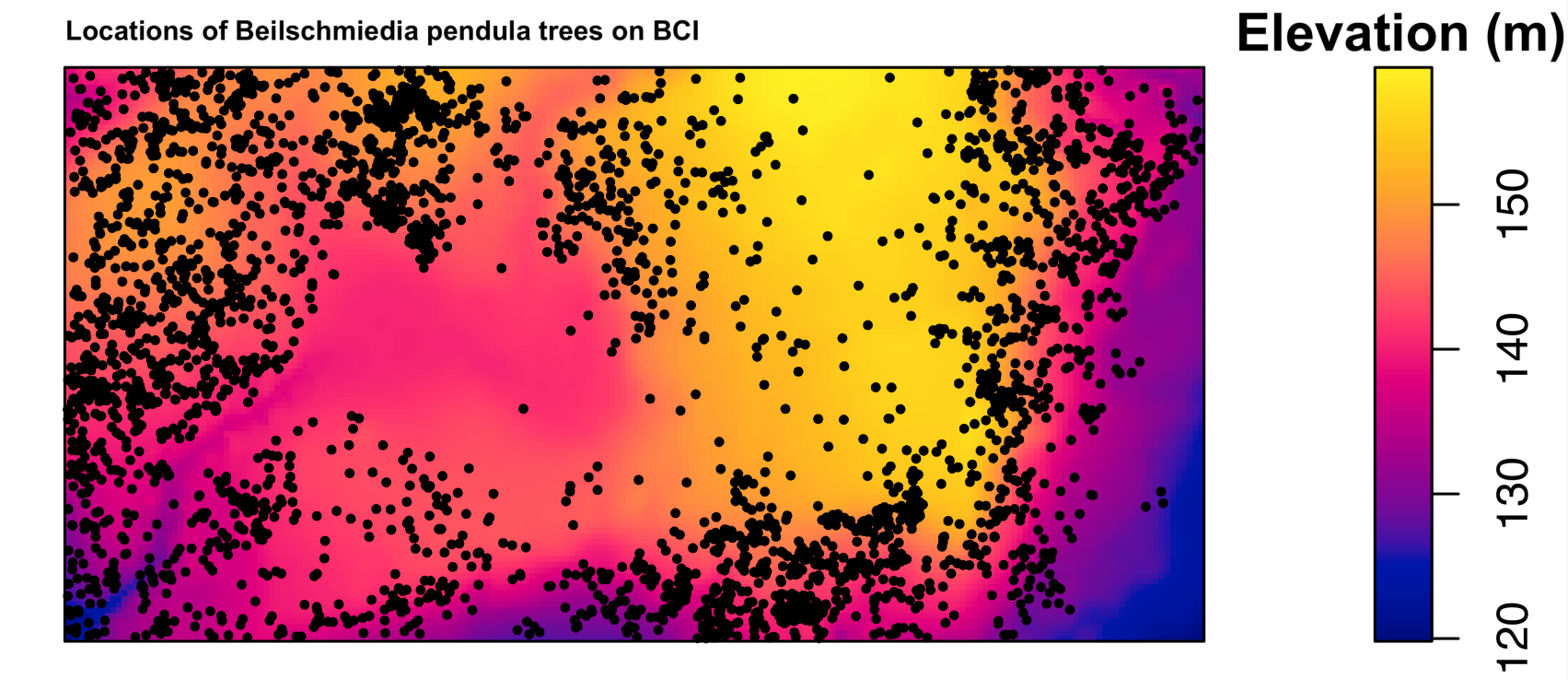
Source: freepik.com

Are these trees clustered?

Source: conservationcorridor.org/

Covariates

Often we have information on potential explanatory variables (i.e., covariates).



Source: `spatstat` package

Points pattern analysis

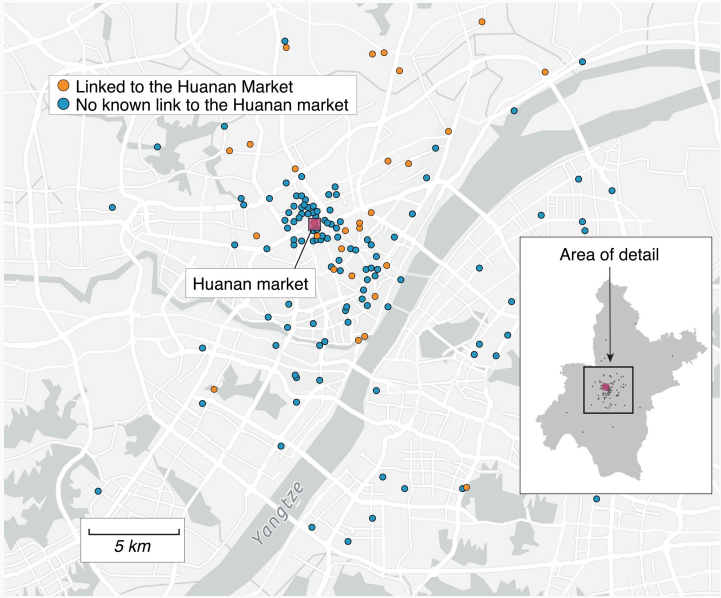
The spatial arrangement of the 'points' is the focus of investigation (e.g., spatial trends in the density of points, relationships with covariates).

The analysis of point patterns can provide key evidence in many fields of research (ecology, epidemiology, geoscience, astronomy, crime research, cell biology, econometrics, etc...)

Cholera in 1854

Source: Wikipedia

COVID-19 in 2019



Source: (Worobey *et al.*, 2022)

Points as data cont.

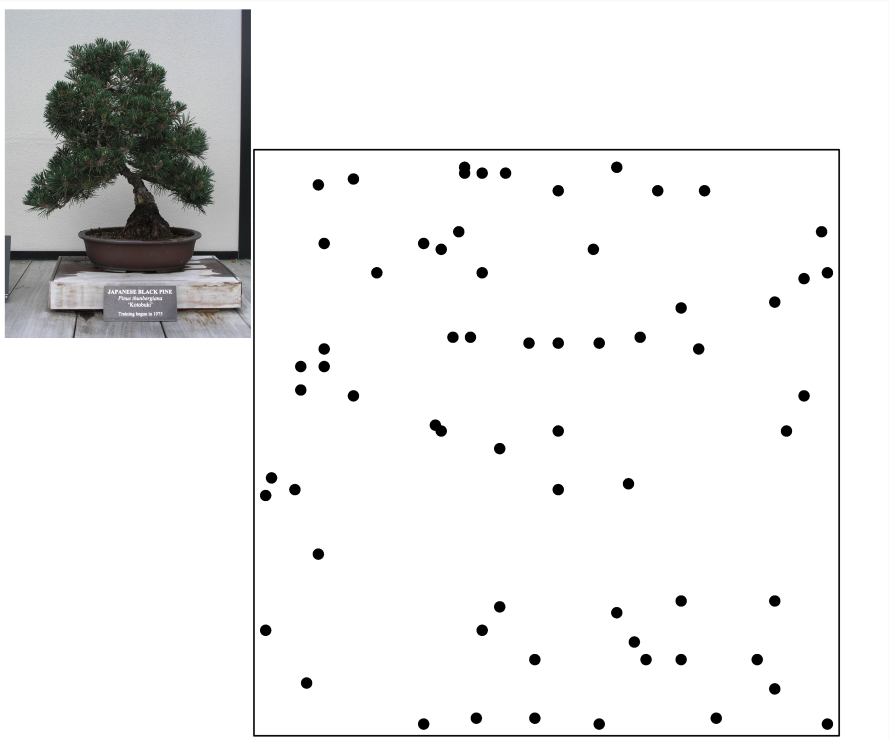
...but the human eye is often not able to objectively assess point patterns.

Are these water striders territorial, or randomly dispersed?

Source: Cleveland Museum of Natural History

Points as data cont.

Are the locations of Japanese black pine saplings clustered on the landscape? competing for light/nutrients?



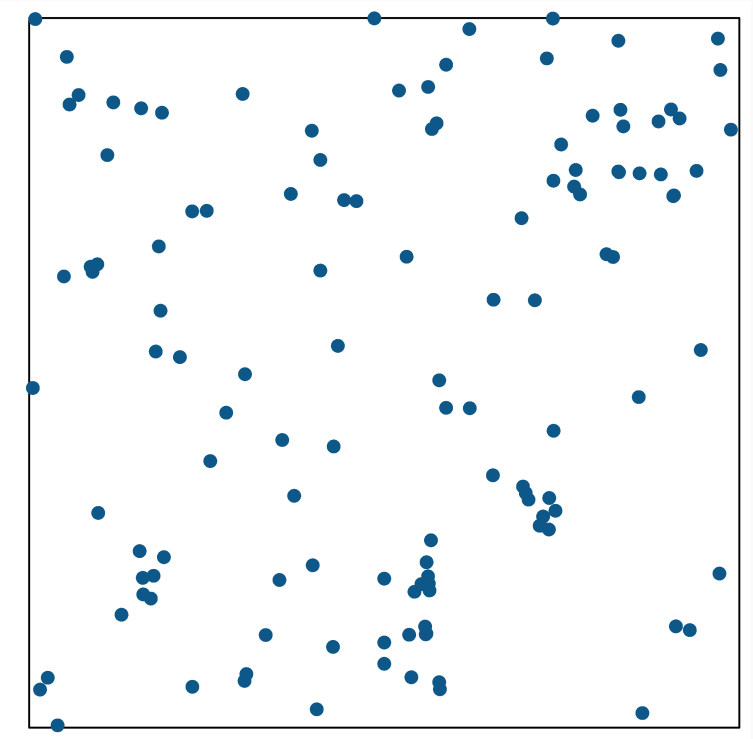
Source: `spatstat` package

Clearly we need a way to carry out formal, objective analyses.

Point Intensity

Finnish Pine Dataset

Today we will work with a dataset describing the locations of 126 pine saplings, their heights (in m) and their diameters (in cm), in a Finnish forest.



Source: `spatstat` package

Descriptive Statistics

With some point data in hand, the first thing we usually want to do is visualise our data and calculate some summary statistics

...and the first summary statistics we want to calculate is the average number of points per unit area (i.e., our 'expectation', or 'first moment').

In point pattern analysis, this quantity is called the 'intensity', denoted λ.

Note: estimating the intensity generally requires few assumptions.

Intensity

Under an assumption of homogeneity, the expected number of points falling within B is proportional to the area of B:

$$\mathbb{E}[n\mathbf{X} \cap B] = \lambda |B|$$

where $\mathbb{E}[n\mathbf{X} \cap B]$ is the expected number of points in B

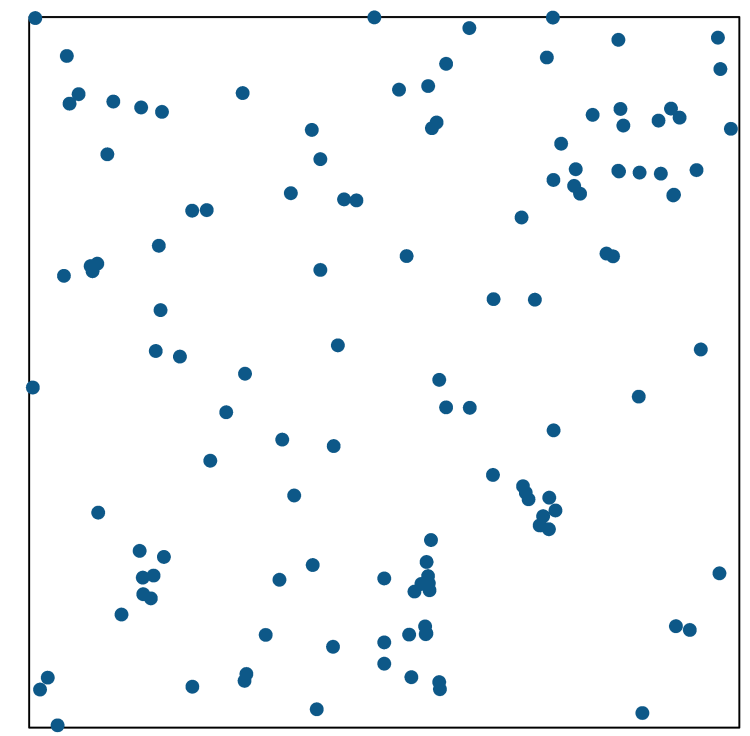
λ is the intensity (in points per unit area)

$|B|$ is the area of B (in units of area)

Intensity cont.

The simplest estimator of λ is just the number of points in our window B, divided by the area of B

$$\hat{\lambda} = \frac{n(\mathbf{x})}{|B|}$$



```
#Load in the necessary packages
library(spatstat)

#Load in and plot the Finnish pines dataset
data(finpines); plot(finpines, use.marks = F)

#Estimate intensity by hand
npoints(finpines)/area(Window(finpines))
[1] 1.26

#Get units
unitname(finpines)
metre / metres

#Estimate intensity automatically
intensity(finpines)
[1] 1.26
```

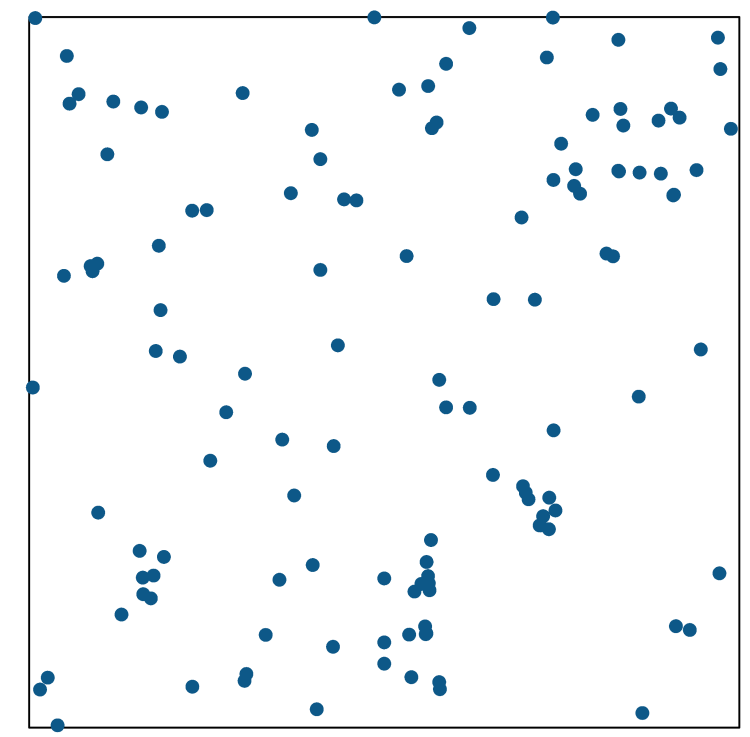
i.e., 1.26 trees per m^2

Weighted intensity

For marked datasets, we might also be interested in a weighted intensity.

E.g., the Finnish pines dataset has information on the heights (in m) and diameters (in cm) of the saplings.

Assuming the trees are cones, we can estimate their volumes as $\frac{\pi h d^2}{12}$.



```
head(marks(finpines),3)
  diameter height
1        1    1.7
2        1    1.7
3        1    1.6

#Calculate the volume of each tree
height <- marks(finpines)$height
diameter <- marks(finpines)$diameter/100
volume <- (pi * height * diameter^2)/12

#Estimate the weighted intensity
intensity(finpines, weights = volume)
[1] 0.001273693
```

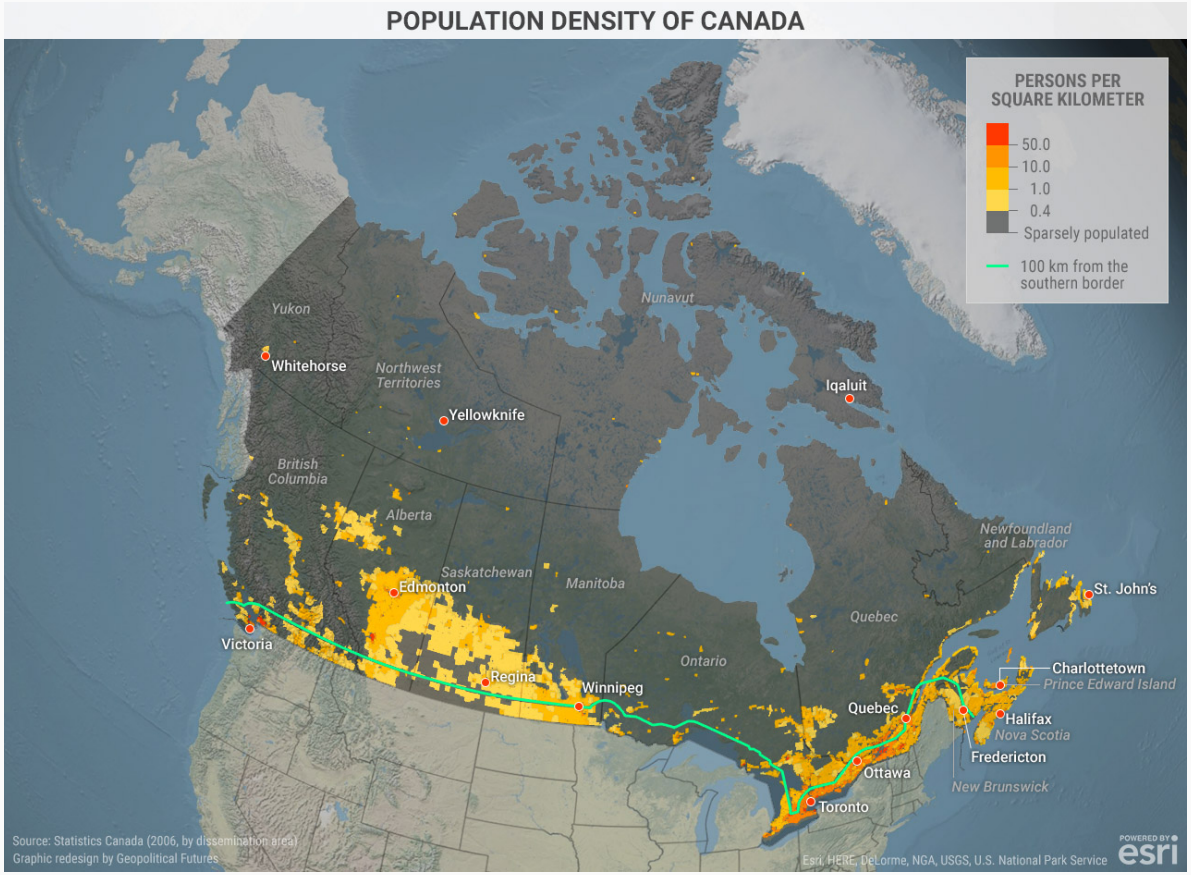
i.e., $\sim$0.0013 m^3 of wood per m^2

Bias

Area of Canada: $\sim$ 8,788,702.8 km^2

Population of Canada: $\sim$ 36,991,981

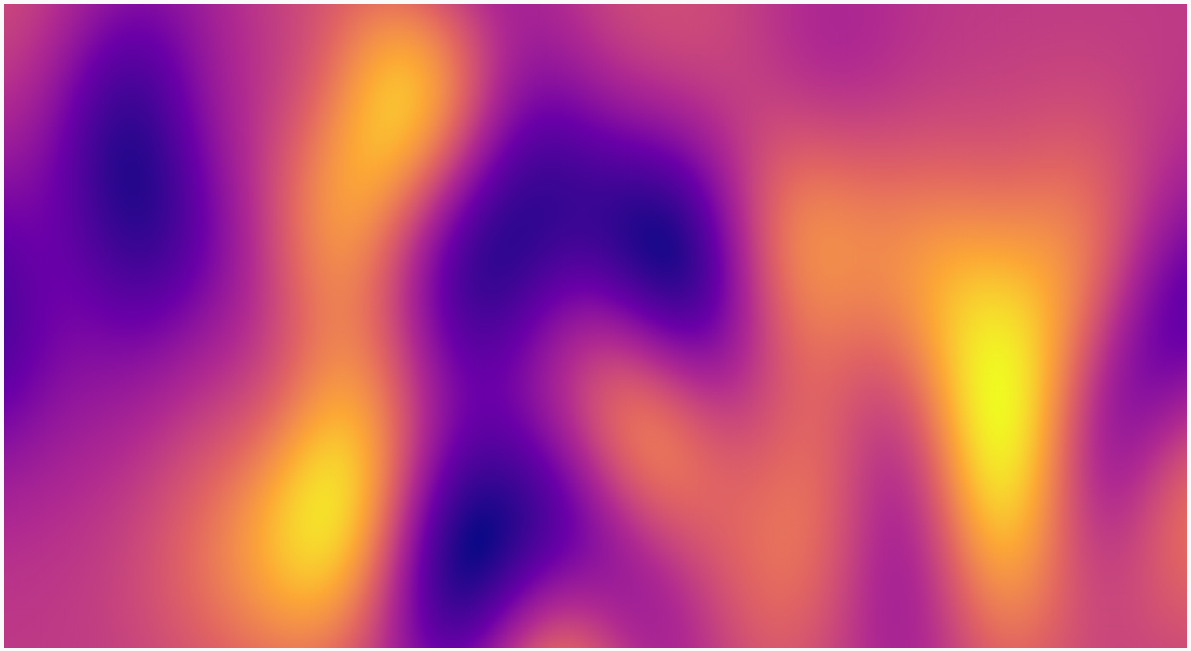
$$\hat{\lambda} = \frac{36{,}991{,}981}{8{,}788{,}702.8} = 4.2 \text{ people/km}^2$$



Generalised intensity

Estimating the intensity in this way assumes homogeneity (i.e., λ is constant in space).

In most real scenarios, λ is likely to be spatially varying.



... which means our estimate would be biased and misrepresentative.

Generalised intensity cont.

When λ is spatially varying, the intensity at any location u is $\lambda(u)$.

The number of points falling in B is thus given by the integral of the intensity function within B

$$\mathbb{E}[n(\mathbf{X} \cap B)] = \int_B \lambda(u) du$$

...which means we now need an estimator of $\lambda(u)$.

Quadrat counting

When λ is spatially varying, $\lambda(u)$ can be estimated nonparametrically by dividing the window into sub-regions (i.e., quadrats) and using our simple points/area estimator.

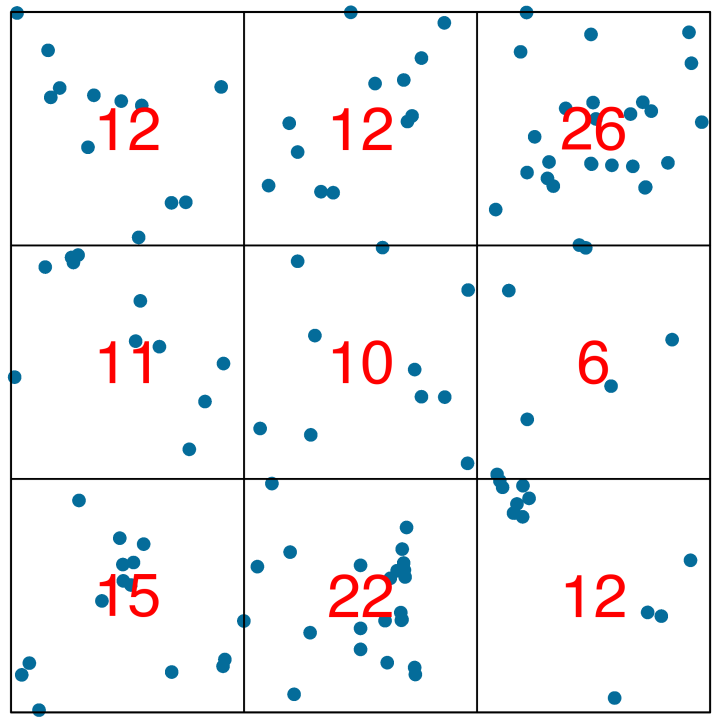
The number of points n falling in each quadrat j, is $n_j = n(\mathbf{x} \cap B_j)$ for $j = 1 \ldots, m$, which is an unbiased estimate of $\mathbb{E}[n(\mathbf{X} \cap B_j)]$.

We can therefore estimate the intensity in each quadrat by counting the number of points in each quadrat divided by the quadrat's area

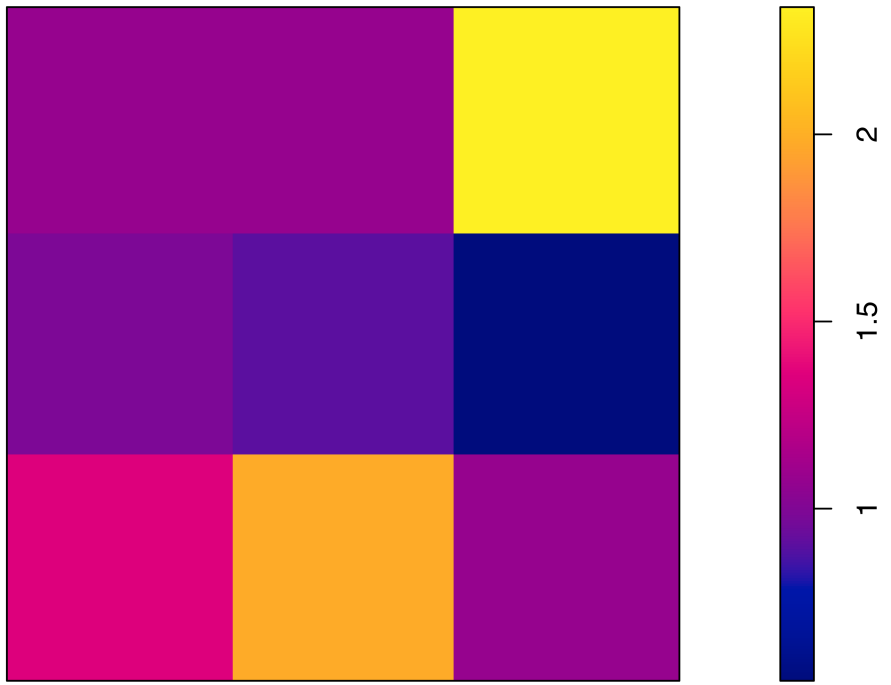
$$\hat{\lambda(j)} = \frac{n(\mathbf{x} \cap B_j)}{|B_j|} \text{ for } j = 1 \ldots, m$$

Quadrat counting in R


```
#Split into a 3 by 3 quadrat and count points
Q3by3 <- quadratcount(finpines,
                      nx = 3,
                      ny = 3)

#Plot the output
plot(finpines, pch = 16,
     use.marks = F, cols = "#046C9A")

plot(Q3by3, cex = 2, col = "red", add = T)
```




```
#Estimate intensity in each quadrat
intensity(Q3by3)
                x
y                [-5,-1.67) [-1.67,1.67) [1.67,5]
  [-1.33,2]            1.08          1.08     2.34
  [-4.67,-1.33)        0.99          0.90     0.54
  [-8,-4.67)           1.35          1.98     1.08

#Plot the output
plot(intensity(Q3by3, image = T))
```



Test of homogeneity

Quadrat counting suggests a spatially varying $\lambda(u)$, but point processes are stochastic and some variation is expected, so how can we objectively test for spatial homogeneity?

Under a null hypothesis that the intensity is homogeneous, and if all quadrats have equal area, then the expected number of points falling in each quadrat, j, is just λa_j, where a_j is the area of each quadrat.

We can therefore test for significant deviations from complete spatial randomness (CSR) using a χ^2 test

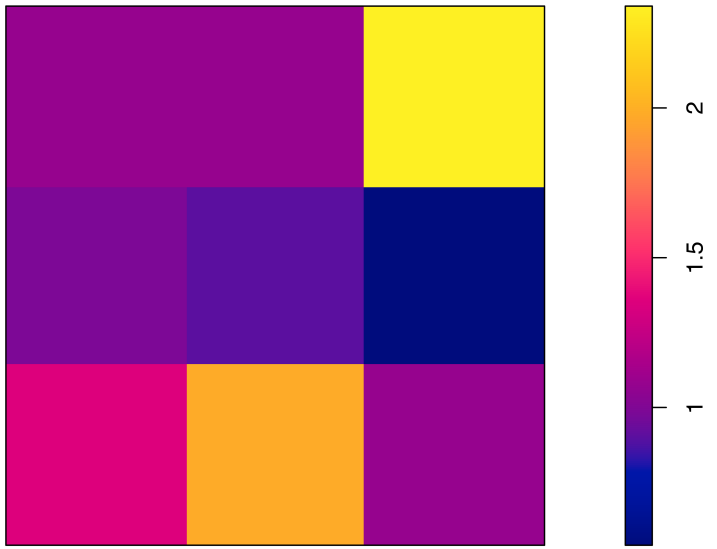
$$\chi^2 = \sum_j \frac{(\text{observed} - \text{expected})^2}{\text{expected}} = \sum_j \frac{(n_j - \hat{\lambda} a_j)^2}{\hat{\lambda} a_j},$$

where $\hat{\lambda}$ is estimated using the points/area estimator.

Test of homogeneity in R

This test can be performed using the `quadrat.test()` function from the `spatstat` package.

```
#Quadrat test
quadrat.test(Q3by3)


  Chi-squared test of CSR using quadrat counts

data:
X2 = 22.143, df = 8, p-value = 0.009316
alternative hypothesis: two.sided

Quadrats: 3 by 3 grid of tiles
```



...which suggests that there's a significant deviation from homogeneity.

Test of homogeneity caveats

This test shows up regularly, but the p-value doesn't provide any information on the cause of inhomogeneity.

Significant deviations can be due to the processes being inhomogenous, but also due to a lack of independence.

Result is sensitive to the size of the quadrats (recommended to compute multiple times and plot the test statistics vs. quadrat size).

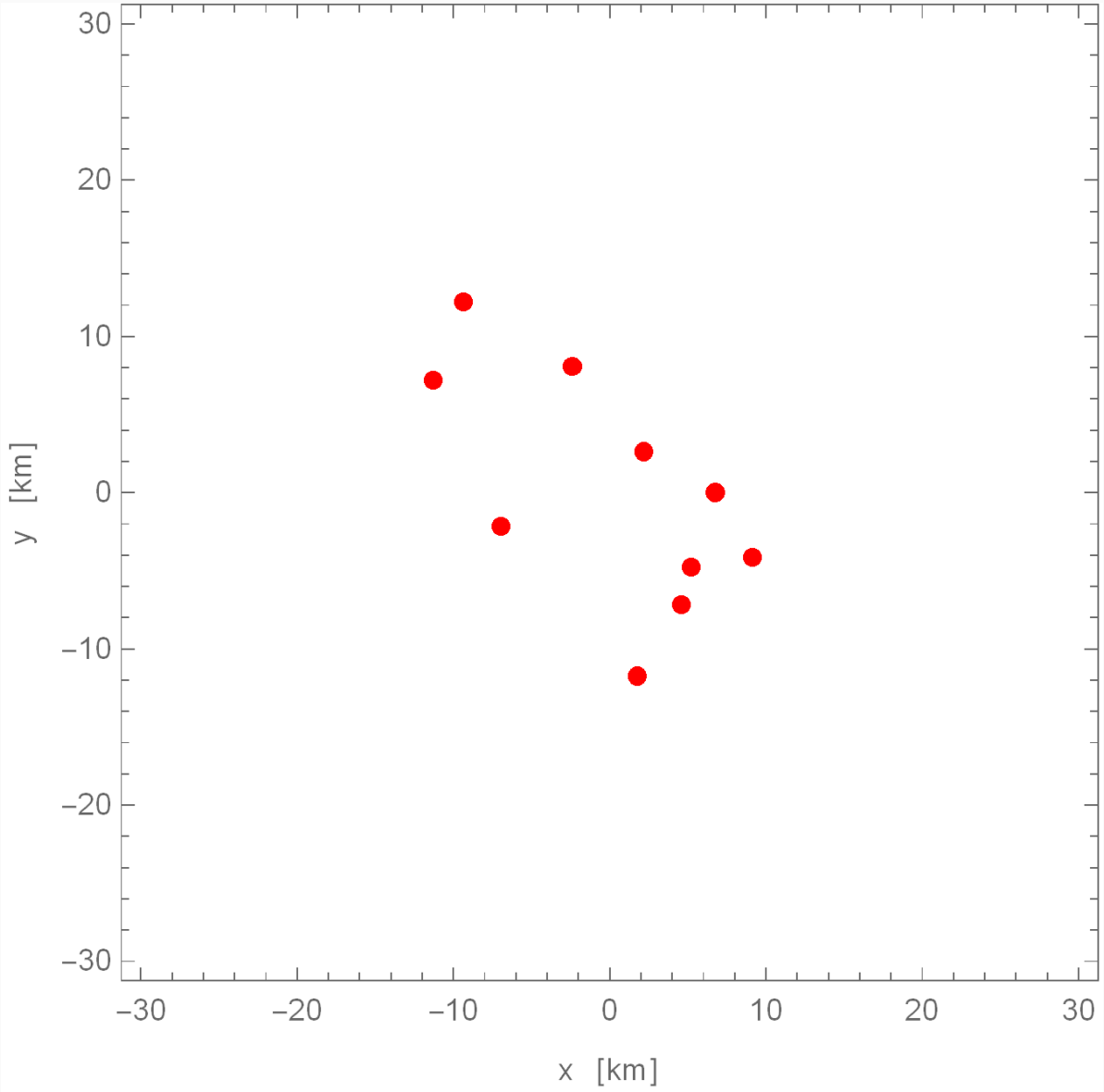
Kernel density estimation

A spatially varying, $\lambda(u)$ can also be estimated nonparametrically by kernel density estimation.

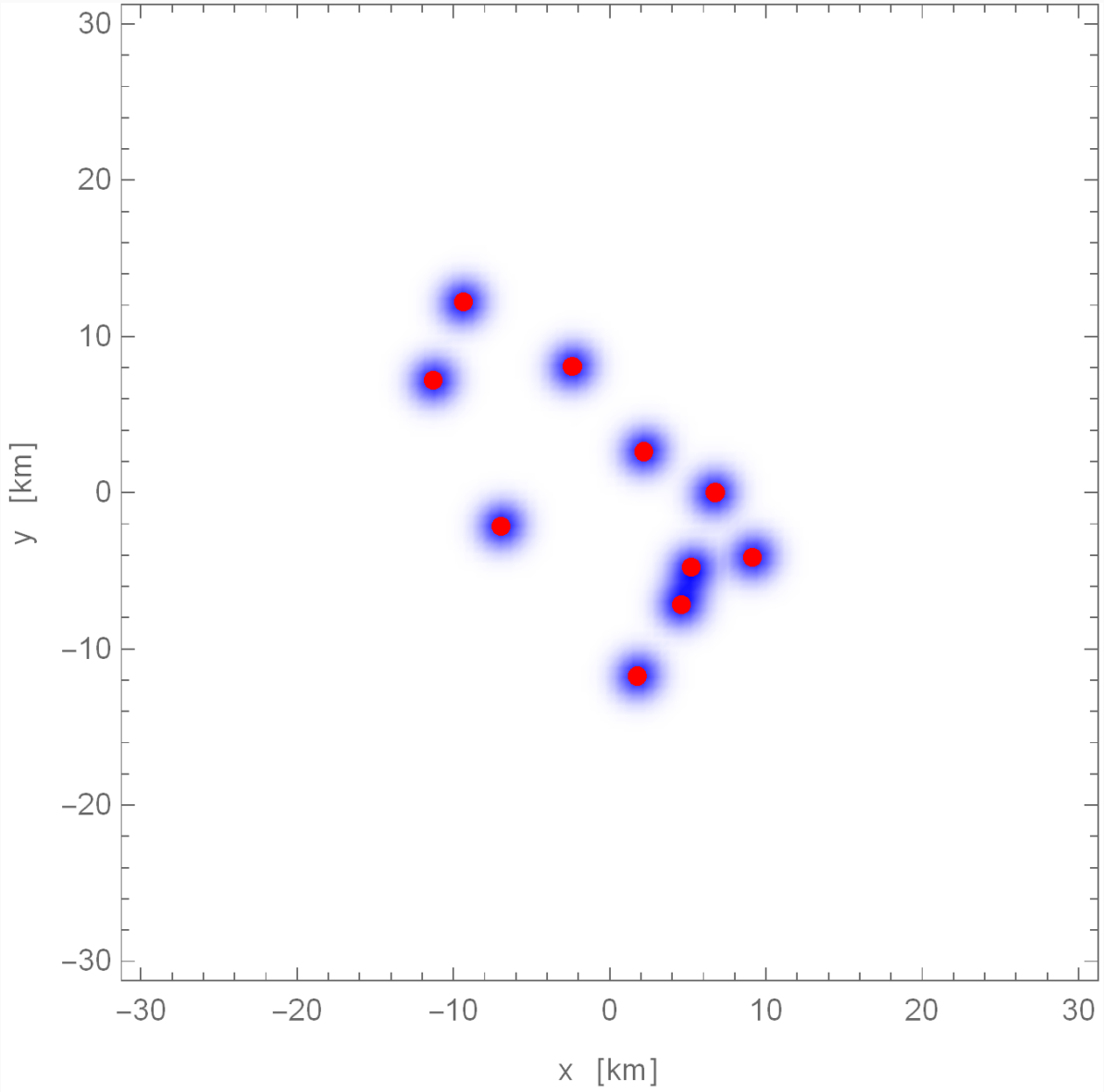
Kernels estimate $\lambda(u)$ by placing 'kernels' on each datapoint (often bi-variate Gaussian) and optimising the 'bandwidth' (i.e., the standard deviation of the kernel).

In practice, there are many different bandwidth optimisers, kernel shapes, and bias corrections for estimating $\hat{\lambda}(u)$ (beyond the scope of this course).

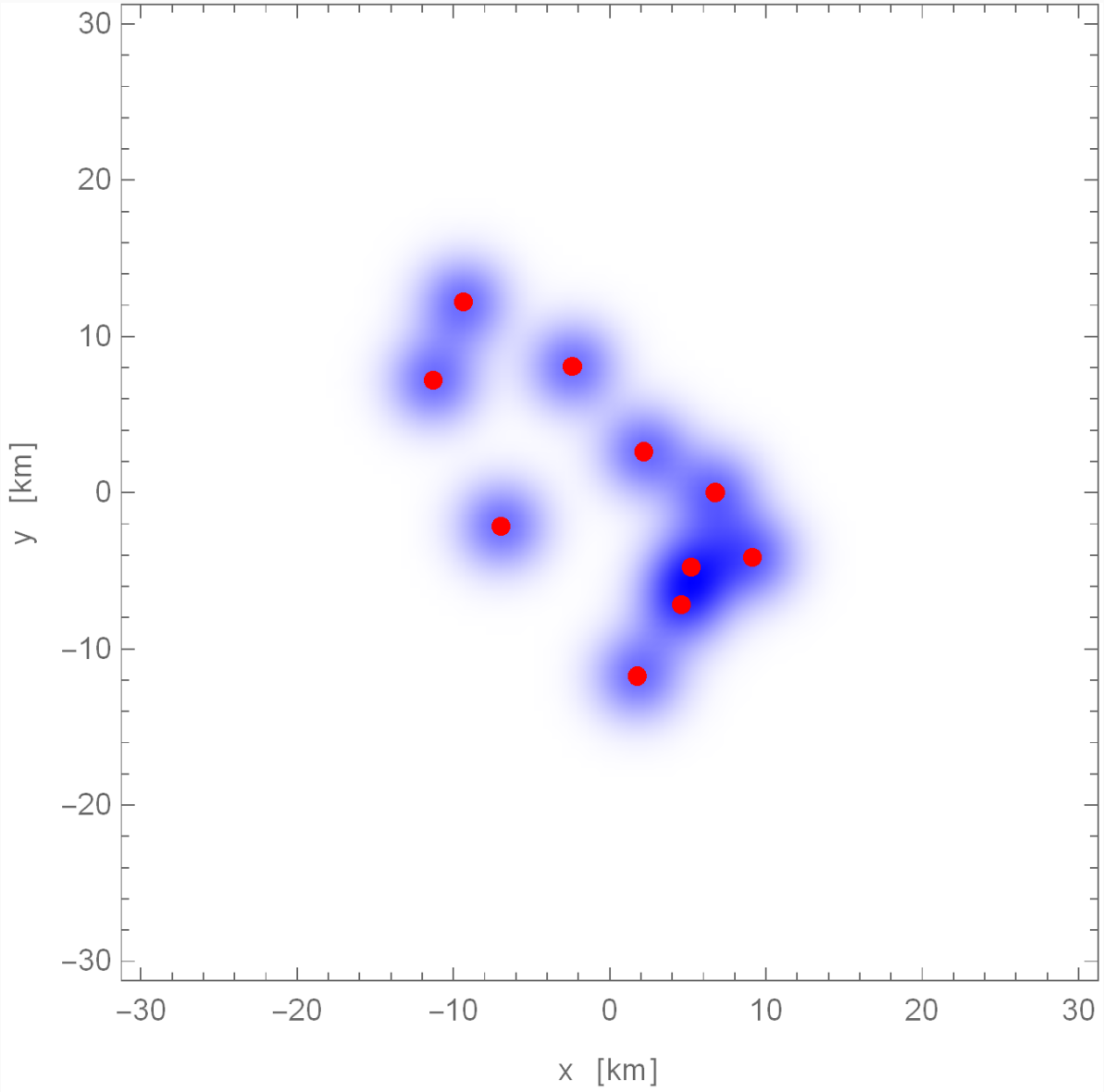
KDE: Start with data



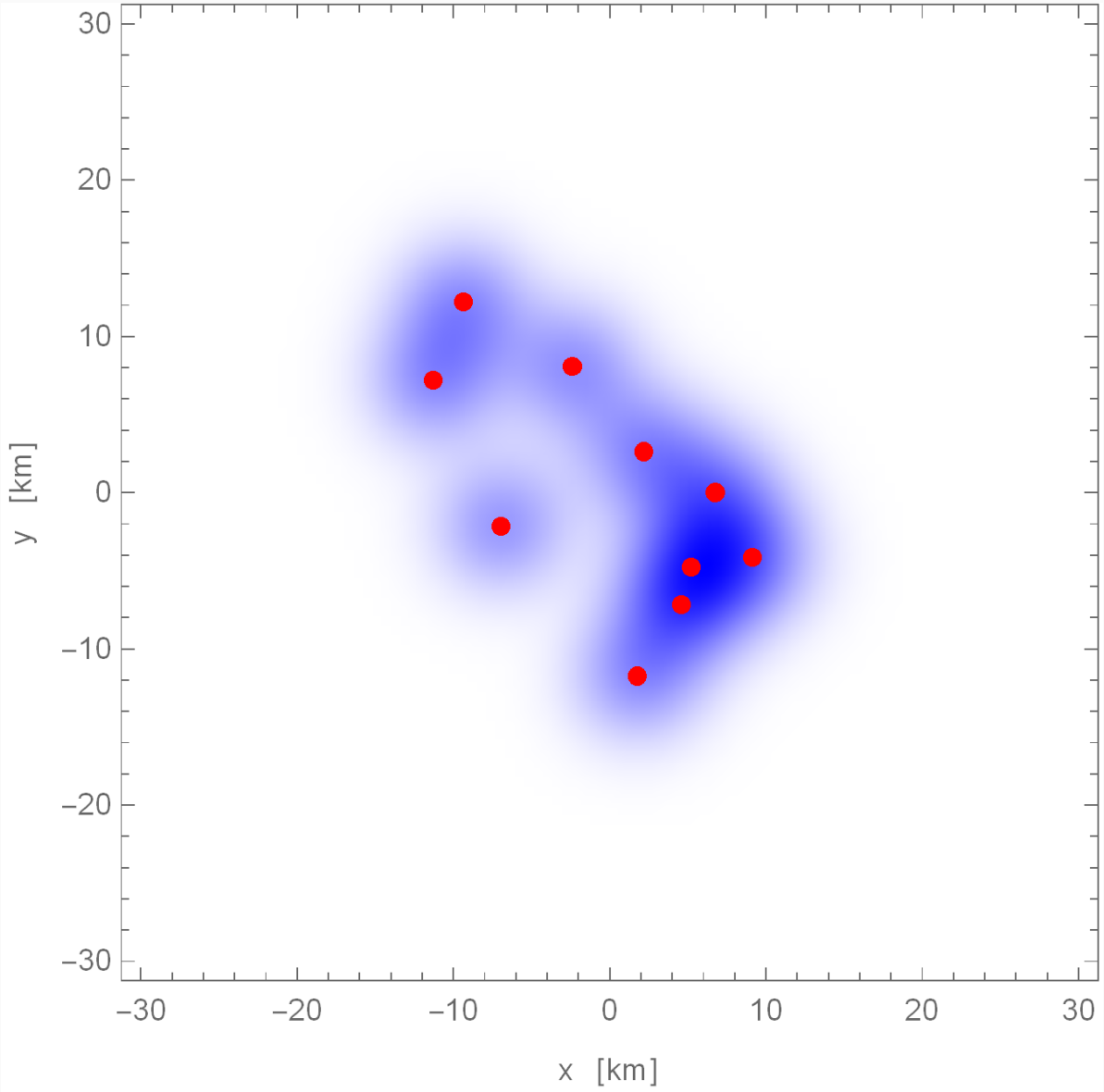
KDE: Place kernels on data



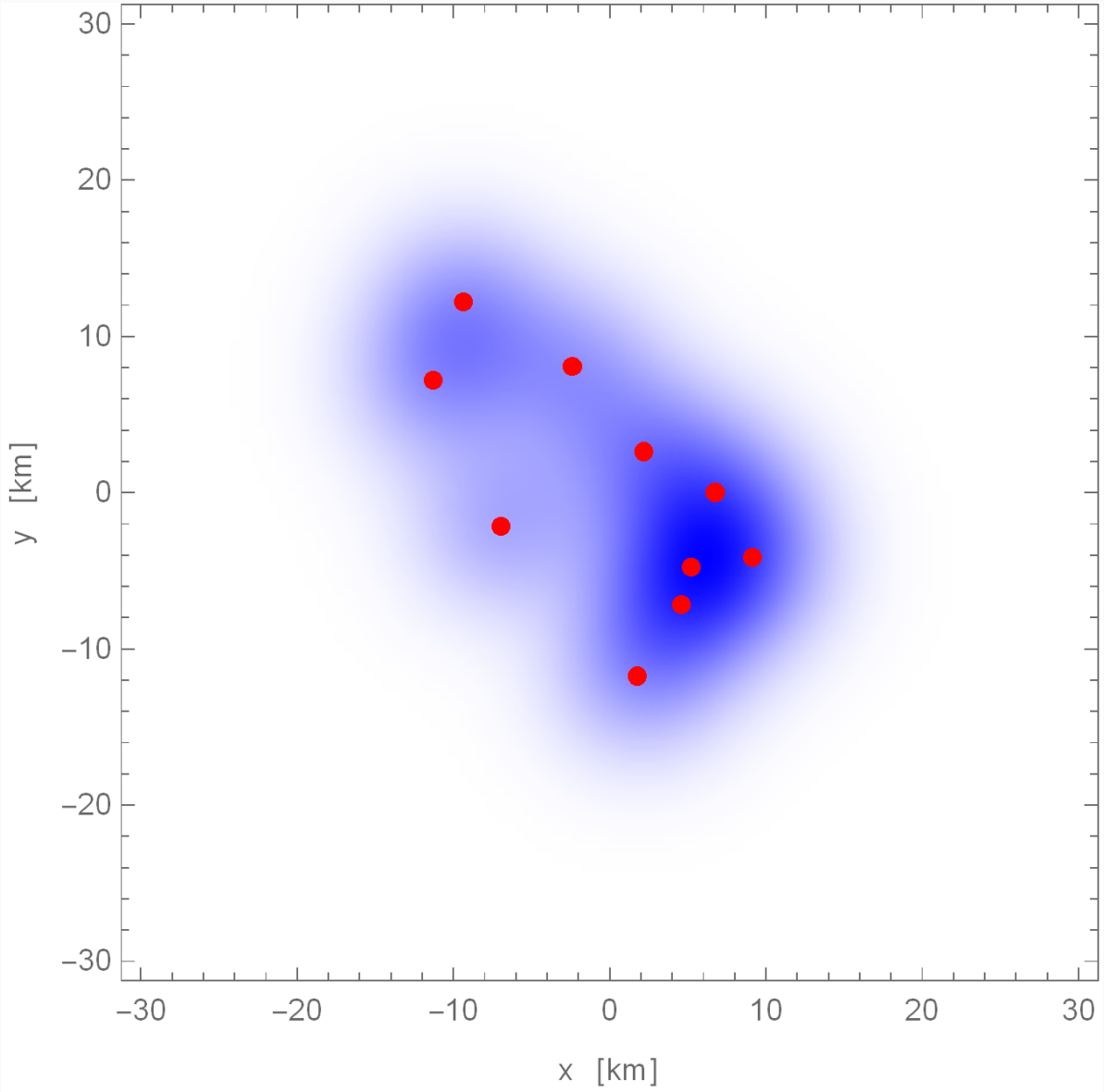
KDE: Optimize bandwidth.



KDE: Optimize bandwidth..



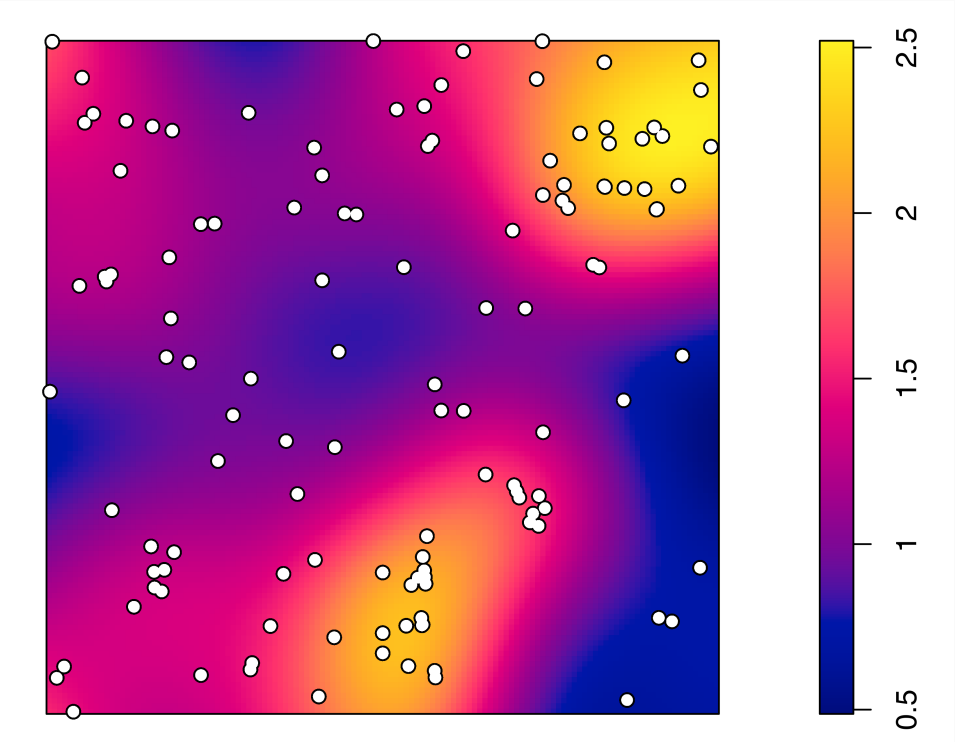
KDE: Optimize bandwidth...

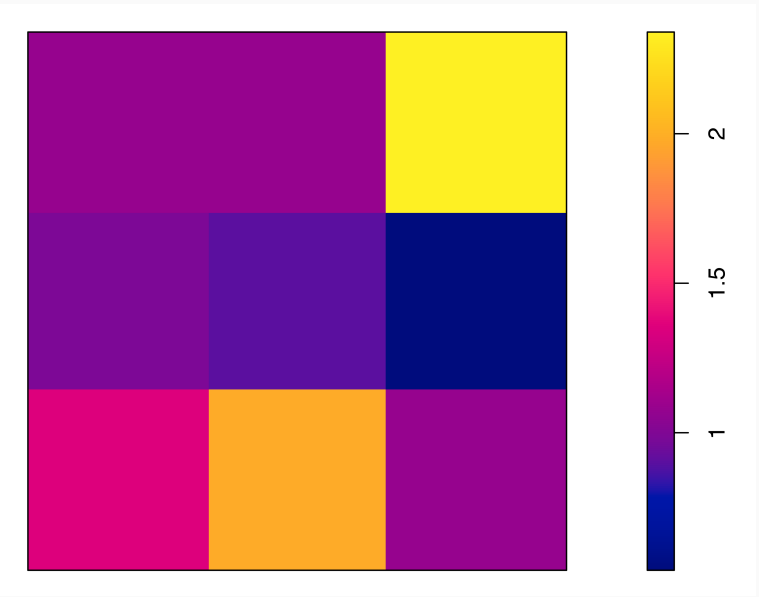


Kernel estimation in R


```
#Density estimation of lambda(u)
lambda_u_hat <- density(finpines)

#Plot the output
plot(lambda_u_hat)
plot(finpines, pch = 16,cex = 1.2,
     use.marks = F, add = T)
plot(finpines, pch = 16, cex = 0.9,
     use.marks = F, cols = "white", add = T)
```

Comparable results to quadrat estimation, but with finer-scale resolution.





Kernel estimation considerations

Kernel estimation is the most efficient non-parametric density estimation technique, but can be sensitive to data features and the chosen estimation technique.

Weighted kernel estimation can be carried out via the `weights` argument of the `density()` function.

The default uses a single bandwidth across the whole dataset, but this can be relaxed by using adaptive smoothing via the `adaptive.density()` function.

This estimate of $\hat{\lambda}(u)$ (points/area) assumes the window is perfectly flat, so topography can bias this estimate.

Hot spots

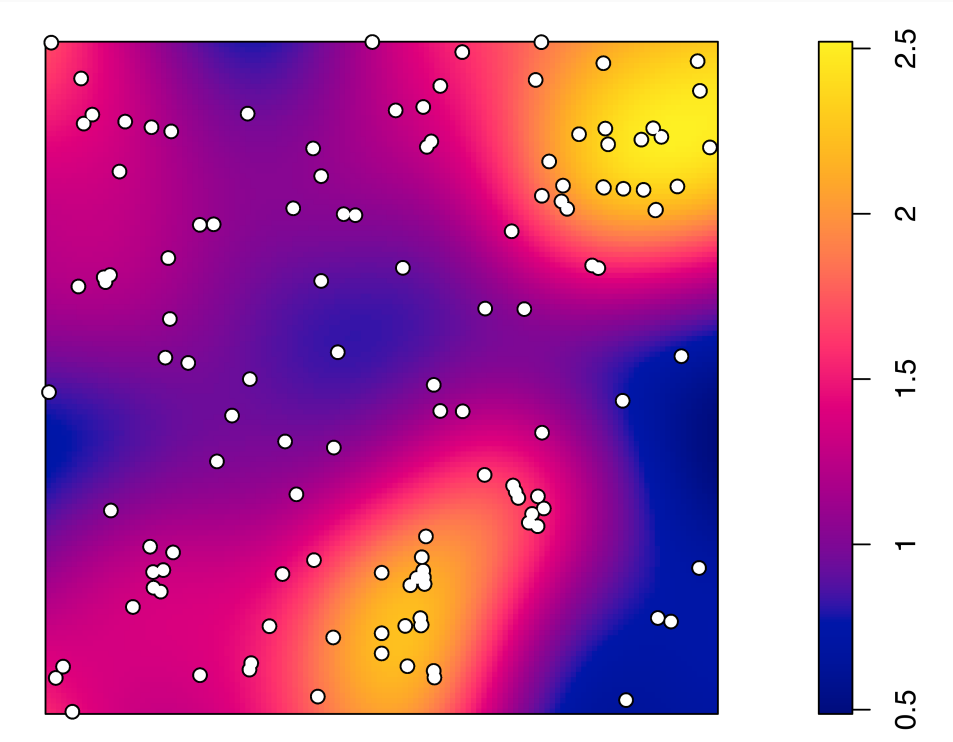
If the intensity is inhomogeneous, we often want to identify areas of elevated intensity (i.e., hotspots).

Identifying hotspots can provide valuable information on a spatial processes (e.g., high crime areas, a high density of artifacts at an archeological dig, dense clusters of galaxies in the universe).

Which tools we should use for detecting hot spots is still an open-ended question.

Hot spot analysis

The best place to start is usually with the kernel estimate (zones of elevated intensity are usually clearly visible).



Sometimes a visual assessment is sufficient (depends on goals).

Hot spot analysis cont.

If we need something more objective, one option is a 'scan test':

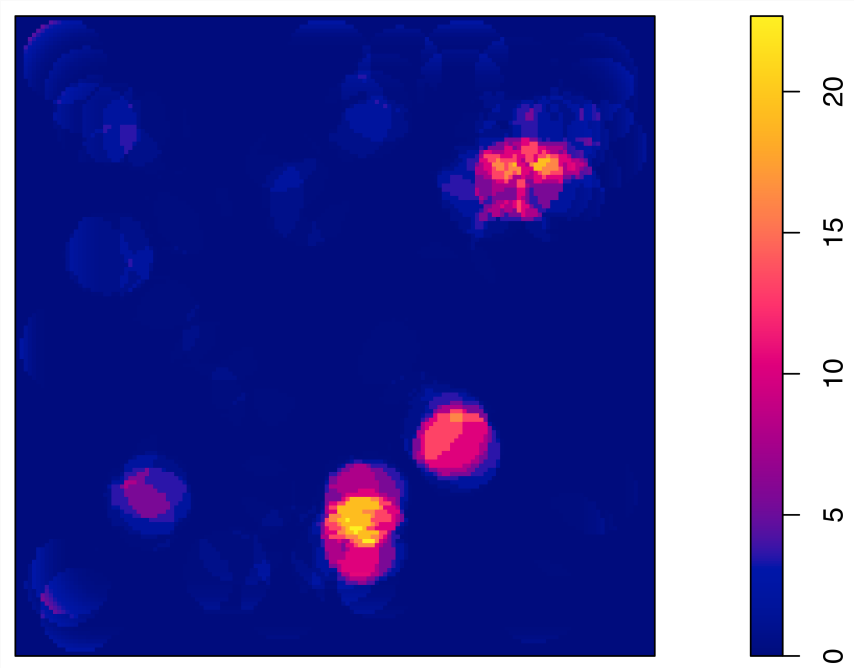
- At each location u, we can draw a circle of radius r.
- We can then count the number of points in $n_{\text{in}} = n(\mathbf{x} \cap b(u, r))$ and out $n_{\text{out}} = n(\mathbf{x} \cap W \notin b(u, r))$ of the circle.
- Under an assumption that the process is Poisson distributed, we can calculate a likelihood ratio test statistics for the number of points inside vs. outside of the circle (details in spatstat textbook).
- The null distribution is $\sim \chi^2$ with 1 degree of freedom (allowing us to calculate p-values).

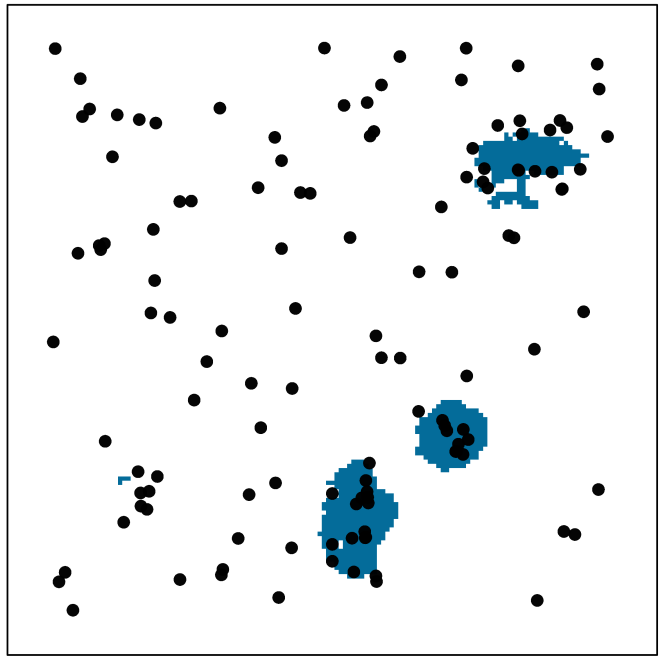
Hot spot analysis in R

A likelihood ratio test can be undertaken via the `scanLRTS()` function from the `spatstat.explore` package.

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

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

#Plot the output
plot(LR)
```

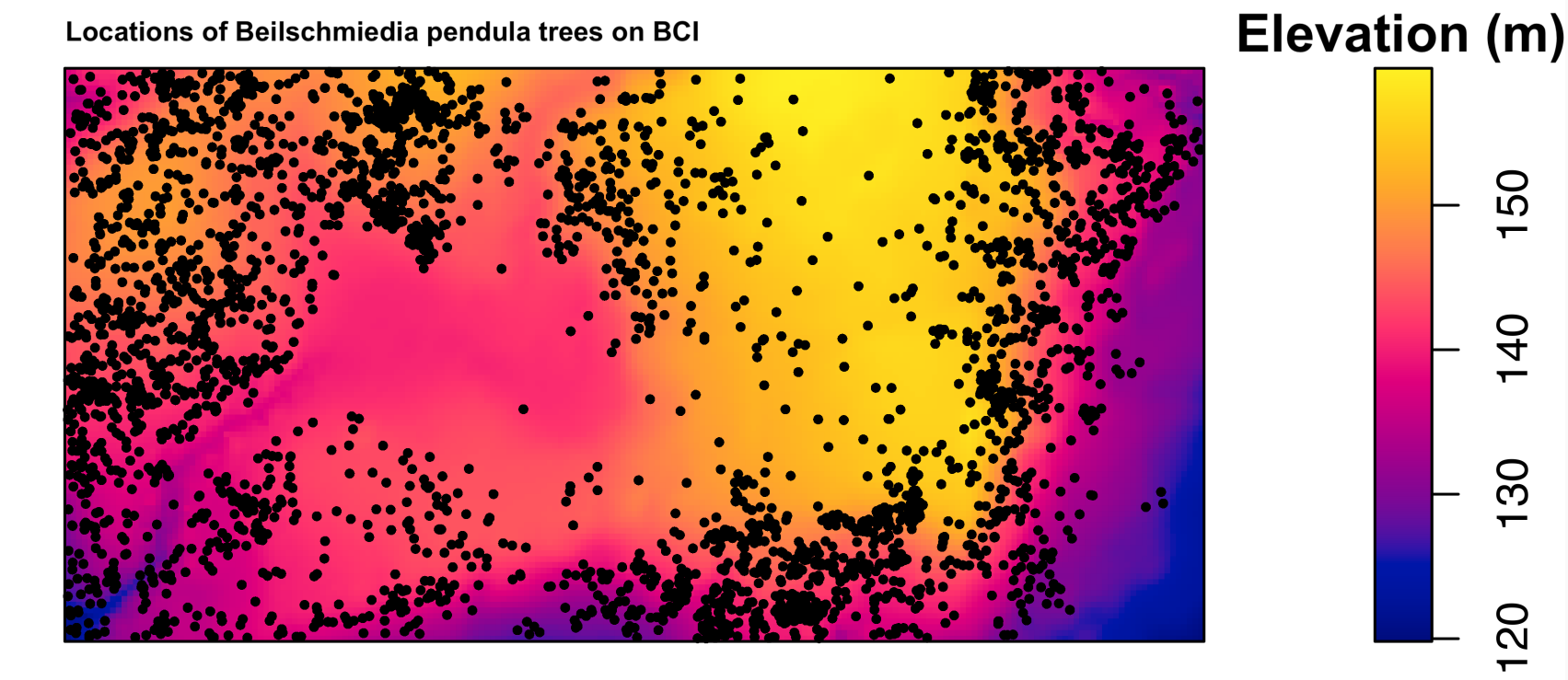
```
#Compute p-values
pvals <- eval.im(pchisq(LR,
                        df = 1,
                        lower.tail = FALSE))

#Plot the output (filtered for p < 0.01)
plot(pvals)
```





Relationships with covariates

We are usually interested in determining whether the intensity depends on a covariate(s).



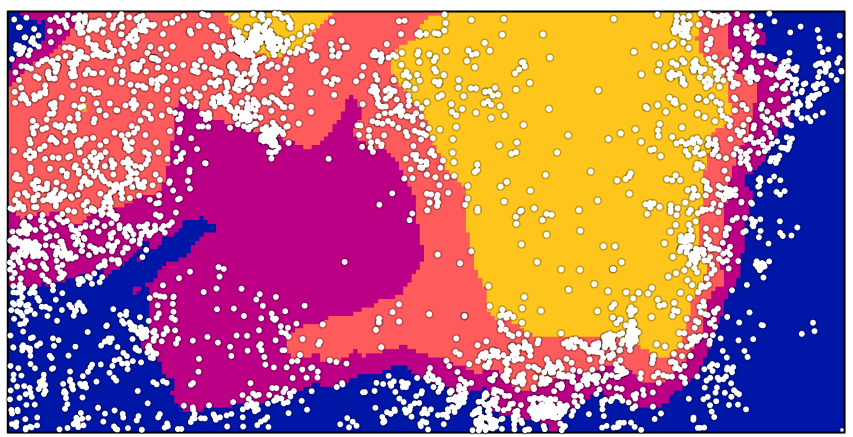
Source: `spatstat` package

A visual assessment may be informative, but is unlikely to be sufficient.

Quadrate counting

The simplest approach to check for a relationship between $\lambda(u)$ and a spatial covariate $Z(u)$ is via quadrat counting.

```
#Extract elevation information
elev <- bei.extra$elev

#define quartiles
b <- quantile(elev, probs = (0:4)/4, type = 2)

#Split image into 4 equal-area quadrats based on elevation values
Zcut <- cut(elev, breaks = b)
V <- tess(image = Zcut)

#Count points in each quadrate
quadratcount(bei, tess = V)

tile
(120,140] (140,144] (144,150] (150,159]
      714       883      1344       663
```



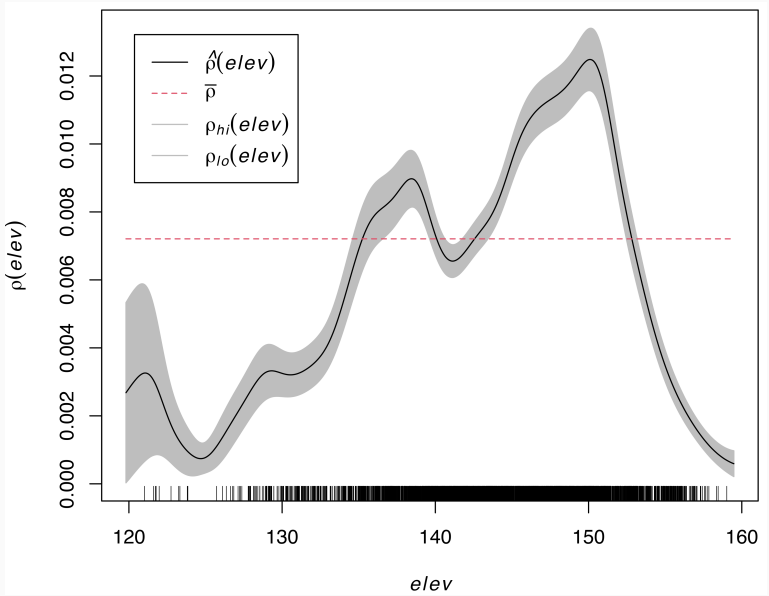
Relative distribution estimate

More formally, in testing for relationships with covariates we are assuming that λ is a function of Z, such that

$$\lambda(u) = \rho(Z(u))$$

A non-parametric estimate of ρ can be obtained via kernel estimation, available via the `rhohat()` function.

```
#Estimate Rho
rho <- rhohat(bei, elev)

plot(rho)
```



Take home messages

A spatial point pattern is a dataset comprised of the locations of 'things' or 'events', and the spatial arrangement of the 'points' is the focus of investigation.

The formal analysis of a point process usually begins with a descriptive analysis of the intensity (many tools and few assumptions).

Spatial patterns in intensity can provide valuable information on a point process, but many of the tests are sensitive to they way they are setup, so results should be interpreted with care.

Next lecture we will focus on how to describe the relationships between points (second moment descriptive statistics).

References

Morales, A.B. & Laurini, M.P. (2021). Firm location: A spatial point process approach. *Applied Spatial Analysis and Policy*, pp. 1–33.

Worobey, M., Levy, J.I., Malpica Serrano, L., Crits-Christoph, A., Pekar, J.E., Goldstein, S.A., Rasmussen, A.L., Kraemer, M.U., Newman, C., Koopmans, M.P. *et al.* (2022). The huanan seafood wholesale market in wuhan was the early epicenter of the covid-19 pandemic. *Science*, 377, 951–959.