Ville Väänänen
63527M

Lansing woods

S-114.4202 Special Course in Computational Engineering II

February 13, 2012

1 Data description

It's an important question in forest ecology wether certain tree species are spatially associated with each other and how they respond to competition. The Lansing Woods dataset [3] contains the location and botanical classification of 2251 trees. The data was collected in Lansing Woods, Clinton County, Michigan USA by D.J. Gerrard in 1969 from a square area of 282×282 metres.

The dataset is available in the *R* package *spatstat* [5, 1]. It's a categorically marked dataset, where the mark can have one of the values

- blackoak

  - *Quercus velutina*
  - known associates: whiteoak, redoak, hickory, maple

- redoak

  - *Quercus rubra*
  - known associates: whiteoak, blackoak

- whiteoak

  - *Quercus alba*
  - known associates: whiteoak, redoak

- hickory

  - *Carya*
  - known associates: Look it up

- maple

  - *Acer*
  - known associates: Look it up

- misc   find the original article

The interesting questions will be:

- do the associations known *a priori* show in the data

- do some species avoid some other species

- clustering behavior inside and between the species

The dataset is plotted in figure 1.

(a) Blackoak   (b) Redoak   (c) Whiteoak

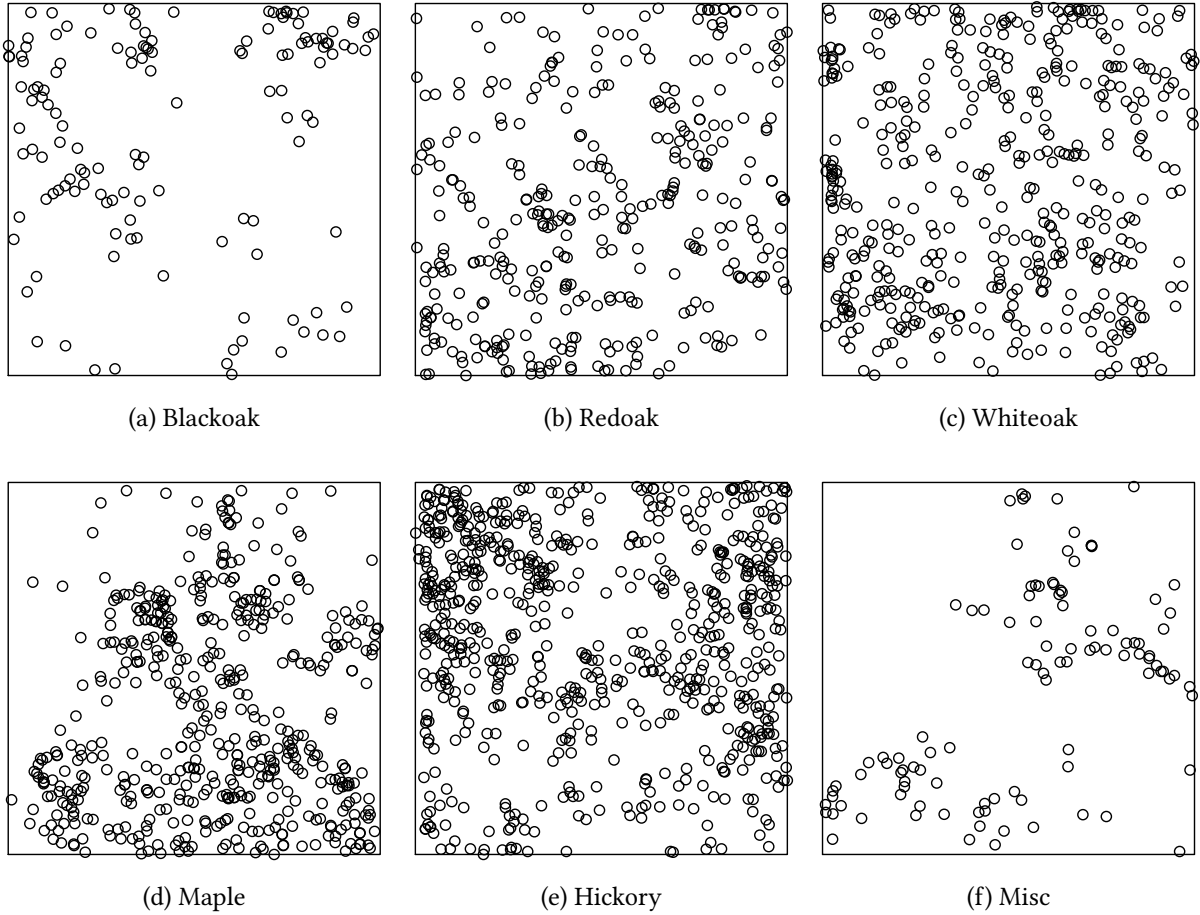(d) Maple   (e) Hickory   (f) Misc

Figure 1: The Lansing Woods dataset

2   Preprocessing

The different oaks, namely the black, the white and the redoaks seem to display rather homogenous intensities judging from the point patterns. Taking into account the prior information, that these oaks tend associate with each other, it seems reasonable to combine the different oaks into a single category. Also since there is no information regarding the constitution of the "misc" category, it is discarded from further analysis. The point patterns resulting from these preprocessing steps are displayed in figure 2.

3   Intensity analysis

It's obvious just by looking at figure 1 that the intensity profiles exhibit significant interspecies variability. For example oaks seems to have almost homogenous intensity whereas maples displays a much more inhomogenous pattern. A Gaussian kernel smoothed intensity estimate is displayed in figure 3, where
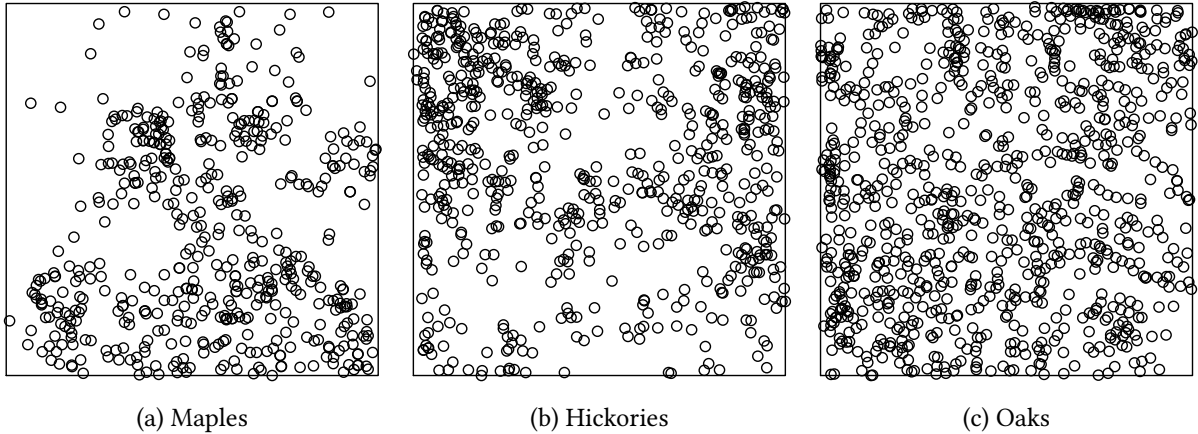
(a) Maples   (b) Hickories   (c) Oaks

Figure 2: The dataset after preprocessing

the intensities are comparable between species.

The most striking conclusion from figure 3 is that the patterns for hickories and maples are almost complementary. The intensity of the oaks varies somewhat, but it seems that there are some oaks pretty much everywhere in the window.

More conclusions can be drawn by plotting some combined point patterns. In figure 4 there are all the trees plotted together, then the oaks and finally the maples and the hickories combined. Indeed, it seems that when plotted in these combinations, it would be reasonable to assume constant intensities. We can already come to the following conclusions

- discarding the marks, the intensity of trees is homogenous

- oaks are independent of other species

- hickories and maples show strong segregation

4   Randomization tests

The homogenous Poisson process is an important null model for point processes, since in that case the intensity is constant and the locations of the points are i.i.d given the number of points. It is then common to test for deviance of the pattern in question to this null model, the model of *complete spatial randomness* (CSR).

A marked point process can depart from complete randomness in multiple ways:

- random labeling

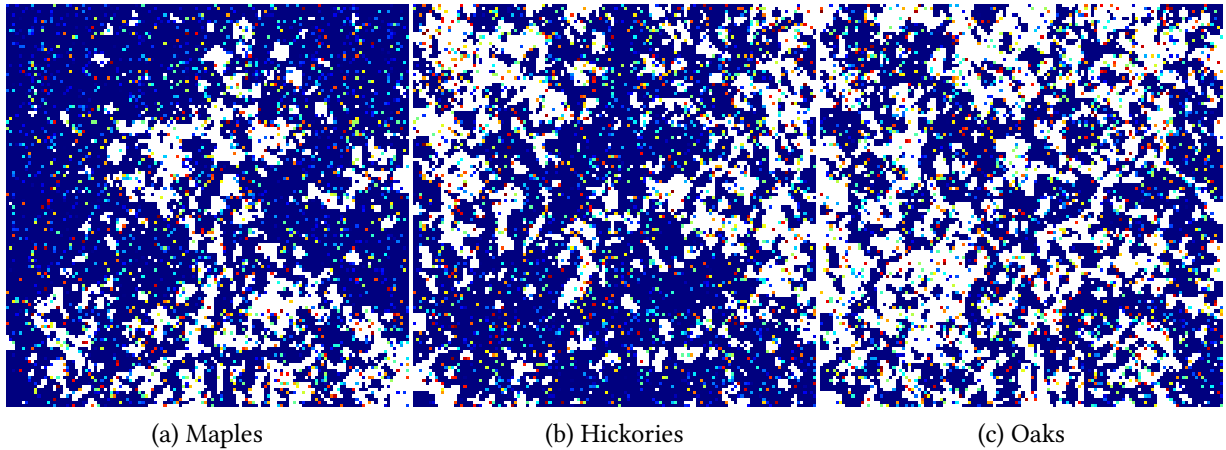   - The marks are i.i.d random variables given the locations

(a) Maples   (b) Hickories   (c) Oaks

Figure 3: Gaussian Kernel smoothed intensity estimates

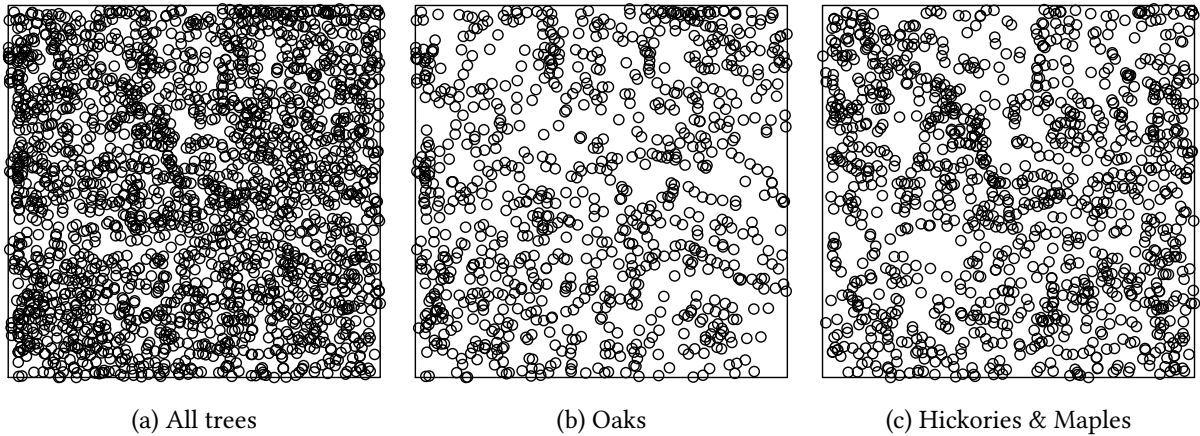(a) All trees   (b) Oaks   (c) Hickories & Maples

Figure 4: Point patterns with different combinations of the marks in the dataset. The patterns display homogenous intensity.

- independence of components
  - The points and marks in each *subprocess* or *component*, i.e. the process that consists of the points having the same categorical marks, exhibit whatever distributional characteristics, but the subprocesses are independent of each other.

- complete spatial randomness and independence (CSRI)
  - the locations are distributed like in a homogenous Poisson process and the marks are i.i.d

Testing for these different null hypotheses is done by considering the "cross" versions of the second order summary functions of the process. In our case it is quite obvious even without testing, that the subprocesses show interaction and

the labeling is not random. For the sake of completeness, tests were carried for each of the null hypotheses.

In figure 5 the CSRI assumption was tested. If the null hypothesis was correct, the L_{ij} function should stay between the envelopes obtained by simulating a homogenous Poisson process.

The indepence of components null hypothesis can be tested by comparing the empirical L_{ij} function to the envelopes obtained by splitting the data into subprocesses by mark, and then randomly shifting them independently of each other. This case is presented in figure 6.

Finally the random labeling property can be accounted for by testing for the deviance of the one-to-any type L-function from the L-function obtained by discarding the marks. These whould be equal under the null hypothesis. The envelopes can be constructed by calculating this deviance for datasets obtained by randomly relabeling the marked point process. The results are presented in figure 7.
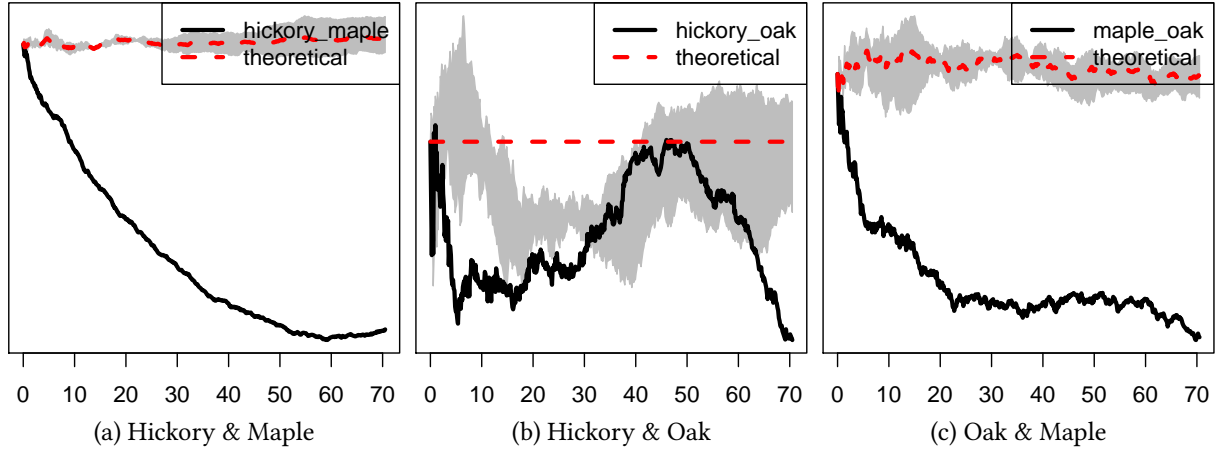

Figure 5: The empirical differences between the L_{ij} functions from the general L function

5   Interaction analysis

Next we will attempt to characterize the interactions within a single species and amongst different species.

5.1   Intra-species interaction

I have plotted the inhomogenous L-functions [4][2] for the maples and hickories and the ordinary L-function for the oaks. These are displayed in figure 8.
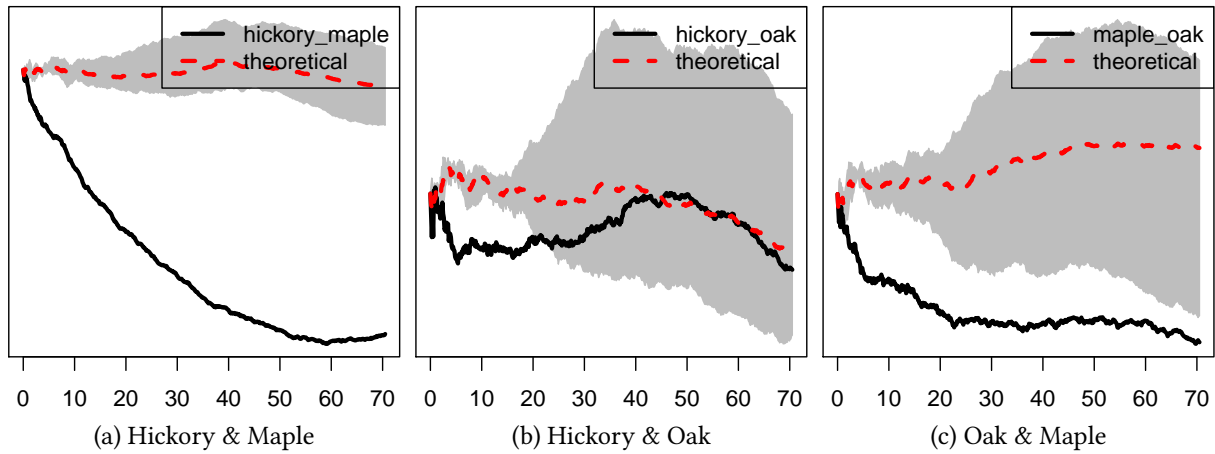
Figure 6: The empirical L_{ij} functions with envelopes from random shiftings of the subprocesses
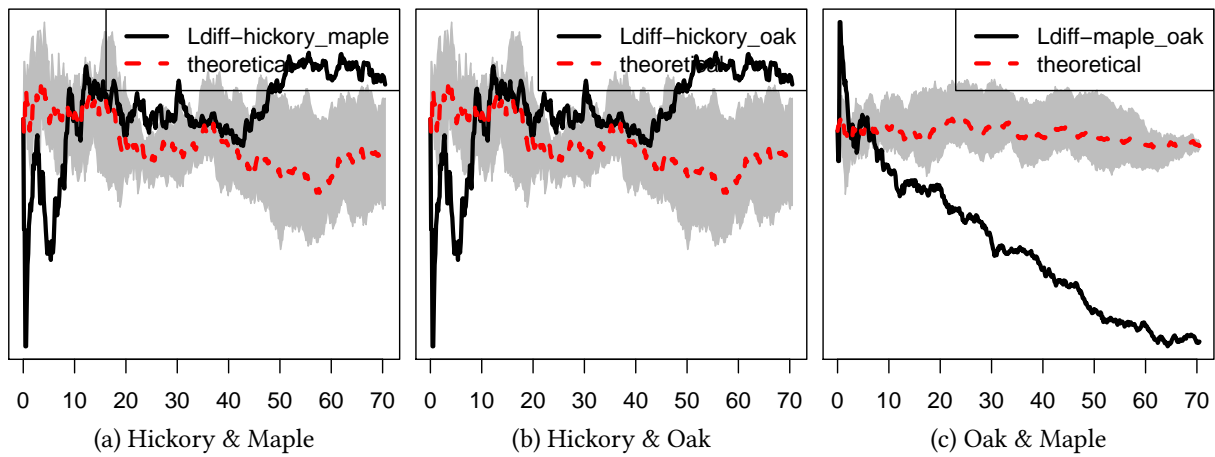

Figure 7: The empirical differences between the L_{ij} functions from the general L function with envelopes from random labelings of the marked point pattern

5.2 Inter-species interaction

The interspecies interaction was quantified by using the two different summary statistics: the *partial pair correlation function* (PPCF) and the *mark connection function*. These can both be interpreted as displaying the probability, that there is a point of species i an r distance away from a point of species j. The plots have been made for all the pairings i, j from the three species, resulting in 6 different plots (pairings of type i, i are also included). The PPCF is displayed in figure 9b and the mark-connection function in figure 9a. As can be seen, both of them show quite similar results.
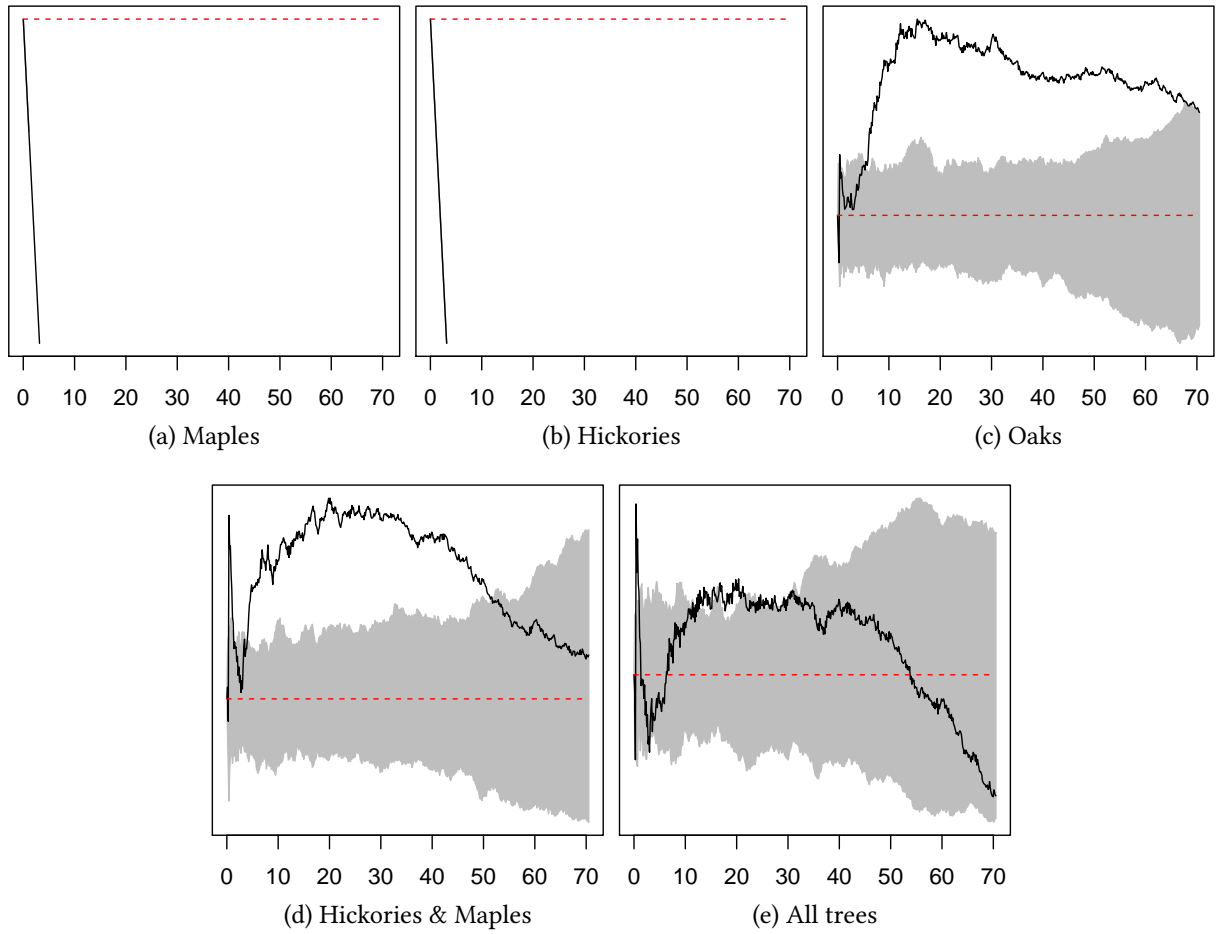
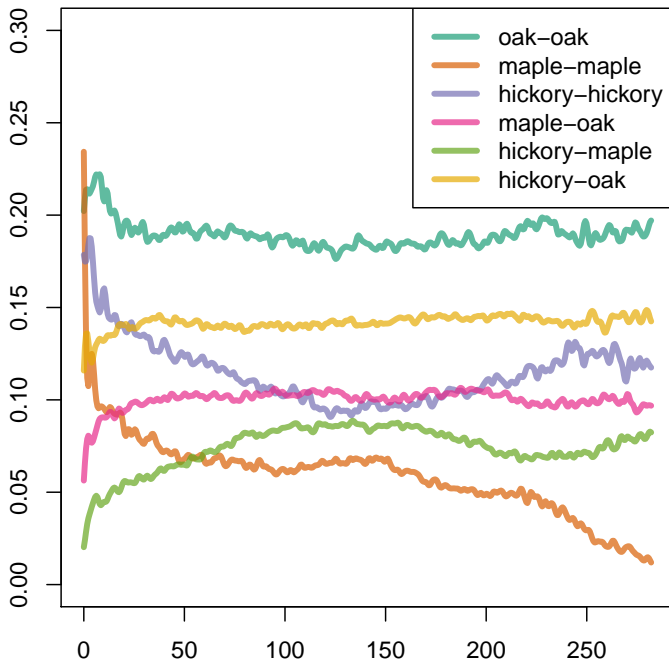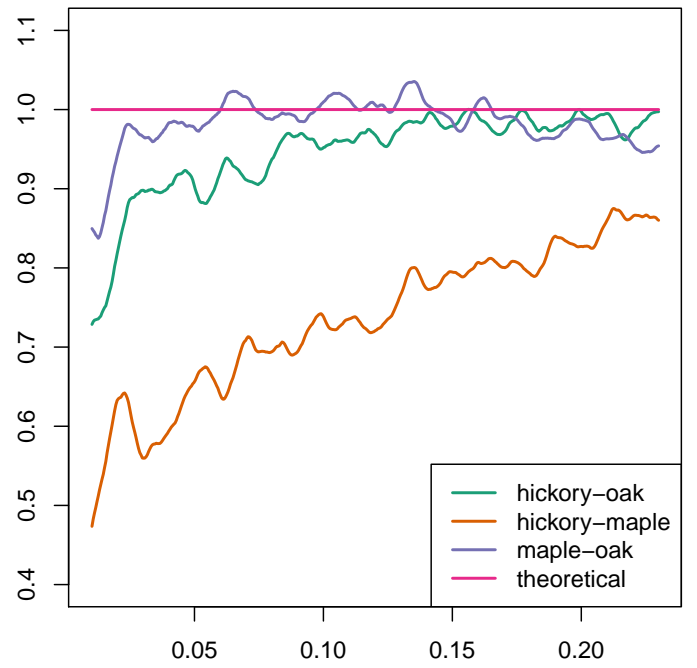Figure 8: Besag's L-function for the different species. For maples and hickories the inhomogenous version was used. In all of the figures there are the envelopes after 20 monte carlo tests assuming complete spatial randomness.

(a) Mark-connection function

(b) PPCF

Figure 9: The mark-connection function and the partial pair-correlation function that measure intra- and interspecies interactions.

6 Methods

6.1 Intensity

6.1.1 Kernel smoother

6.2 Intra-species interaction

6.2.1 K & L functions

6.3 Inter-species interaction

6.3.1 L_{ij} function

6.3.2 $L_{i\bullet}$ function

6.3.3 mark-connection function

7 Results

7.1 Intensity

7.2 Intra-species interaction

7.3 Inter-species interaction

8 Conclusion

References

[1] Adrian Baddeley and Rolf Turner. "Spatstat: an R package for analyzing spatial point patterns". In: *Journal of Statistical Software* 12.6 (2005). ISSN 1548-7660, pp. 1–42. URL: www.jstatsoft.org (cit. on p. 1).

[2] A.E. Gelfand, P. Diggle, and P. Guttorp. *Handbook of spatial statistics*. Chapman & Hall/CRC handbooks of modern statistical methods. Taylor & Francis Group, 2010. ISBN: 9781420072877. URL: http://books.google.fi/books?id=Xf4leslPDzsC (cit. on p. 5).

[3] D.J. Gerrard. "Competition quotient: a new measure of the competition affecting individual forest trees". In: *Research Bulletin 20, Agricultural Experiment Station* (1969) (cit. on p. 1).

[4] J. Illian. *Statistical analysis and modelling of spatial point patterns*. Statistics in practice. John Wiley, 2008. ISBN: 9780470014912. URL: http://books.google.fi/books?id=_U6BER2stYsC (cit. on p. 5).

[5] R Development Core Team. *R: A Language and Environment for Statistical Computing*. ISBN 3-900051-07-0. R Foundation for Statistical Computing. Vienna, Austria, 2011. URL: http://www.R-project.org/ (cit. on p. 1).

R code

```r
# Lansing wood data analysis

exportFigs <- 1
displayFigs <- 0
interaction <- 0
speciesinteraction <- 1
intensity <- 0
dimyx <- ifelse(exportFigs,c(500,500),c(100,100))
nsim <- 3

require("spatstat");
require("RColorBrewer")
#require("playwith");
data(lansing)
lansingm <- lansing

#unitname(lansingm) <- c("metre","metres",round(924/3.2808399))
unitname(lansingm) <- list("metre","metres",1)
ft2m <- round(lansing$window$units$multiplier/3.2808399)
lansingm <- affine(lansingm,diag(c(ft2m,ft2m)))
range <- lansingm$window$xrange

jet.colors <-
  colorRampPalette(c("#00007F", "blue", "#007FFF", "cyan",
                     "#7FFF7F", "yellow", "#FF7F00", "red", "#7F0000"))

mar_lab <- c(2.5,2.5,1.5,1.0)
mar_tight <- c(0.1,0.1,0.1,0.1)
myplot <- function(...,width=6,height=6,mar=mar_lab,file=FALSE,nodevoff=FALSE,
    afterfn=NULL,k=NULL) {
  if(displayFigs) {
    quartz()
    par(mar=mar)
    p <- plot(...)
    if(is.function(afterfn)) {
      afterfn(p,k)
    }
    if(!exportFigs) {
      return(p)
    }
  }
  if(exportFigs && file != FALSE) {
    pdf(file=file,width=width,height=height)
    par(mar=mar)
    p <- plot(...)
    if(is.function(afterfn)) {
      afterfn(p,k)
    }
    if(!nodevoff) {
      dev.off()
    }
```

```
      return(p)
  }
}
listplot <- function(k,v,file=FALSE,formula=FALSE,...) {
  if(formula != FALSE) {
    p <- myplot(v,formula,file=sprintf(file,k),k=k,...)
  } else {
    p <- myplot(v,file=sprintf(file,k),k=k,...)
  }
  return(p)
}

sigma <- bw.relrisk(lansing);

nlansing <- lansingm[lansingm$marks!="misc"];
levels(nlansing$marks) <- c("oak","hickory","maple",NA,"oak","oak")

hm <- lansingm[lansingm$marks=="maple" | lansingm$marks=="hickory"];
levels(hm$marks) <- c(NA,"hickory","maple",NA,NA,NA)

oaks <- lansing[grep("oak",lansing$marks)]
levels(oaks$marks) <- c("blackoak",NA,NA,NA,"redoak","whiteoak")

oakhm <- nlansing
levels(oakhm$marks) <- c("oak","hm","hm")

bw <- bw.diggle(nlansing)

if(intensity) {
  dens1 <- density(split(nlansing),
    sigma=2.5*sigma,
    dimyx=dimyx)

  rl=relrisk(nlansing,sigma=2.5*sigma)

  # smoothed intensities
  mapply(listplot,names(rl),rl,
    MoreArgs=list(
      zlim = c(0, 0.7),
      col=jet.colors(512),
      main="",
      mar=mar_tight,
      ribbon=FALSE,
      file="intensity_relative_%s.pdf",
      width=3,
      height=3
    ))

  # original point patterns
  mapply(listplot,names(split(lansing)),split(lansing),
    MoreArgs=list(
      main="",
      mar=mar_tight,
      file="lansing_%s.pdf",
      width=3,
```

```r
      height=3
    ))

  # combined point patterns
  mapply(listplot,list("oaks","hm","all"),list(oaks,hm,lansing),
    MoreArgs=list(
      use.marks=FALSE,
      pch=21,
      main="",
      mar=mar_tight,
      file="lansing_%s_combined.pdf",
      width=3,
      height=3
    ))
}

if(interaction) {

  patrns <- c(split(nlansing),list(hm=hm,all=nlansing))
  Ls <- mapply(envelope,patrns,list(Lest,Linhom,Linhom,Lest,Lest),
    MoreArgs=list(
      nsim=nsim,
      correction="Ripley",
      normpower=2,
      sigma=2.5*sigma
    ),SIMPLIFY=FALSE)
  mapply(listplot,names(patrns),Ls,
    MoreArgs=list(
      main="",
      formula=.-r~r,
      file="l_%s.pdf",
      legend=FALSE,
      width=3,
      height=3,
      mar=c(2.0,0.3,0.1,0.3),
      yaxt="n"
    ))
}

if(speciesinteraction) {

  # CSRI
  i <- c("hickory","hickory","maple")
  j <- c("oak","maple","oak")
  fns <- mapply(function(i,j){
      return(sprintf("%s_%s",i,j))
    },i,j,USE.NAMES=FALSE)

  Ls1 <- envelope(
      nlansing,
      Lcross,
      i=i[1],
      j=j[1],
      nsim=nsim,
      correction="Ripley",
```

```r
        savepatterns=TRUE)

Ls <- mapply(
    envelope,
    rep(list(nlansing),2),
    rep(list(Lcross),2),
    i=i[2:3],
    j=j[2:3],
    MoreArgs=list(
      nsim=nsim,
      simulate=Ls1
    ),SIMPLIFY=FALSE)
csrd <- c(list(Ls1),Ls)

csrp <- mapply(listplot,fns,csrd,
  MoreArgs=list(
    lwd=2,
    lty=1,
    main="",
    formula=.-r~r,
    file="csri_%s.pdf",
    legend=FALSE,
    width=3,
    height=3,
    mar=c(2.0,0.3,0.1,0.3),
    yaxt="n",
    afterfn=function(p,k) {
      legend(
        'topright',
        c(k,"theoretical"),
        col=p$col[1:2],
        lty=1,
        lwd=2
      )
    }
  ),SIMPLIFY=FALSE)

# independence of components
i <- c("hickory","hickory","maple")
j <- c("oak","maple","oak")
fns <- mapply(function(i,j){
    return(sprintf("%s_%s",i,j))
  },i,j,USE.NAMES=FALSE)

Ls1 <- envelope(
    nlansing,
    Lcross,
    i=i[1],
    j=j[1],
    nsim=nsim,
    correction="Ripley",
    simulate = expression(rshift(nlansing)),
    savepatterns=TRUE)

Ls <- mapply(
```

```r
      envelope,
      rep(list(nlansing),2),
      rep(list(Lcross),2),
      i=i[2:3],
      j=j[2:3],
      MoreArgs=list(
        simulate = Ls1,
        nsim=nsim
      ),SIMPLIFY=FALSE)

iocd <- c(list(Ls1),Ls)

iocp <- mapply(listplot,fns,iocd,
    MoreArgs=list(
      lwd=2,
      lty=1,
      main="",
      formula=.-r~r,
      file="ioc_%s.pdf",
      legend=FALSE,
      width=3,
      height=3,
      mar=c(2.0,0.3,0.1,0.3),
      yaxt="n",
      afterfn=function(p,k) {
        legend(
          'topright',
          c(k,"theoretical"),
          col=p$col[1:2],
          lty=1,
          lwd=2
        )
      }
    ),SIMPLIFY=FALSE)

# random labeling
Ldif <- function(X, ..., i) {
  Lidot <- Ldot(X, ..., i = i)
  L <- Lest(X, ...)
  return(eval.fv(Lidot - L))
}

Ls1 <- envelope(
    nlansing,
    Ldif,
    i="hickory",
    nsim=nsim,
    correction="Ripley",
    simulate = expression(rlabel(nlansing)),
    savepatterns=TRUE)

Ls <- mapply(
    envelope,
    rep(list(nlansing),length(i)),
    rep(list(Ldif),length(i)),
```

```r
      i=c("oak","maple"),
      MoreArgs=list(
        simulate = Ls1,
        nsim=nsim
      ),SIMPLIFY=FALSE)

rld <- c(list(Ls1),Ls)

rlp <- mapply(listplot,fns,rld,
  MoreArgs=list(
    main="",
    formula=.~r,
    file="rl_%s.pdf",
    legend=FALSE,
    width=3,
    height=3,
    mar=c(2.0,0.3,0.1,0.3),
    yaxt="n",
    lwd=2,
    lty=1,
    afterfn=function(p,k) {
      legend(
        'topright',
        c(sprintf("Ldiff-%s",k),"theoretical"),
        col=p$col[1:2],
        lty=1,
        lwd=2
      )
    }
  ),SIMPLIFY=FALSE)

# mark connection functions, pairwise
# bw <- 2*bw.stoyan(nlansing)
# i <- c("hickory","hickory","maple","hickory","maple","oak")
# j <- c("oak","maple","oak","hickory","maple","oak")
# markcs <- mapply(
#     markconnect,
#     rep(list(nlansing),length(i)),
#     i,
#     j,
#     MoreArgs=list(
#       r=seq.int(range[1],range[2],(range[2]-range[1])/500),
#       correction="Ripley",
#       bw=bw,
#       normalise=FALSE
#     ),SIMPLIFY=FALSE)
# markc <- markcs[[1]]
# markc <- markc[,c("r","iso")]
# for(m in markcs[2:length(markcs)]) {
#   markc <- cbind(markc,m[,c("r","iso")])
# }
# col <- sapply(brewer.pal(length(markc)-1,"Dark2"),function(c) {
#     return(paste(c,as.hexmode(round(0.7*255)),sep=''))
#   },USE.NAMES=FALSE)
```

```r
  # v <- myplot(
  #     markc,
  #     legend=FALSE,
  #     col=col,
  #     lty=1,
  #     lwd=4,
  #     ylab="mark-connection",
  #     main="",
  #     file="markc.pdf",nodevoff=TRUE,
  #     xlim=c(range[1],range[2]),
  #     ylim=c(0,0.3),
  #     width=5,
  #     height=5)
  # legend('topright',
  #   mapply(function(i,j){
  #     return(sprintf("%s-%s",i,j))
  #   },rev(i),rev(j)),
  #   col=col,
  #   lwd=4,
  #   lty=v$lty)
  # if(exportFigs) {
  #   dev.off()
  # }

}
```