HW4: Random spatial index (point processes)

Fern Bromley

November 7th, 2025

Homework Guidelines

Please submit your answers on Gradescope as a PDF with pages matched to question answers.

One way to prepare your solutions to this homework is with R Markdown, which provides a way to include mathematical notation, text, code, and figures in a single document. A template `.Rmd` file is available through D2L.

Make sure all solutions are clearly labeled, and **please utilize the question pairing tool on Gradescope**. You are encouraged to work together, but your solutions, code, plots, and wording should always be your own. Come and see me during office hours or schedule an appointment when you get stuck and can't get unstuck.

I. Mathematical [5 pts]

(1) [5 pts] Consider an inhomogeneous Poisson process over a region of the real plane that includes the whole unit circle. If the process has intensity $\lambda(u) = \kappa \exp\left\{\frac{-||u||^2}{2\alpha}\right\}$, $u \in \mathbb{R}^2$, what is the expected number of points in the unit circle, $\mathrm{E}[n(\mathbf{x})]$, when $\kappa = 10$ and $\alpha = 2$? You may provide your answer as a formula or a numerical value with three significant figures. You can answer the question using techniques from calculus, or take a numerical approach, or even a Monte Carlo approach.

```
lambda_r <- function(r){
  10 * exp(-(r^2)/4) * r
}
l_int <- integrate(lambda_r, lower = 0, upper = 1)
l_int$value*pi*2 # ~ 28 points in the circle
```

```
## [1] 27.79671
```

```
library(tigris)
```

```
## To enable caching of data, set 'options(tigris_use_cache = TRUE)'
## in your R script or .Rprofile.
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.2     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
```

```
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(spatstat)
```

```
## Loading required package: spatstat.data
## Loading required package: spatstat.univar
## spatstat.univar 3.1-4
## Loading required package: spatstat.geom
## spatstat.geom 3.6-0
## Loading required package: spatstat.random
## spatstat.random 3.4-2
## Loading required package: spatstat.explore
## Loading required package: nlme
##
## Attaching package: 'nlme'
##
## The following object is masked from 'package:dplyr':
##
##     collapse
##
## spatstat.explore 3.5-3
## Loading required package: spatstat.model
## Loading required package: rpart
## spatstat.model 3.4-2
## Loading required package: spatstat.linnet
## spatstat.linnet 3.3-2
##
## spatstat 3.4-1
## For an introduction to spatstat, type 'beginner'
```

II. Exploration [17 pts]

For this lab, you will investigate a possible connection between earthquakes and oil & gas drilling in Kansas.

(2) [2 pts] Use the `tigris` package to create a simple features object named `kansas_sf` that represents the boundary of the state of Kansas. **Provide executable code** (**PEC**).

```
kansas_sf <- states() %>%
  filter(NAME == "Kansas")
```

```
## Retrieving data for the year 2024
```

```
##   |                                                                              |
```

```
library(sf)
```

```
## Linking to GEOS 3.13.0, GDAL 3.8.5, PROJ 9.5.1; sf_use_s2() is TRUE
```

```
earthquakes_df <- read.csv("Kansas_Earthquake_Database.csv")
earthquakes_df <- earthquakes_df[!duplicated(earthquakes_df[, c("Date", "Time",
                                                                "Longitude", "Latitude")]), ]
big_earthquakes <- earthquakes_df[which(earthquakes_df$Magnitude >= 3), ]
earthquakes_sf <- st_as_sf(big_earthquakes, coords = c("Longitude", "Latitude"),
                           crs = "+proj=longlat")
```

(3) [2 pts] Explain in words what lines 3–7 do. **PEC** that replaces the original character string-class Date variable in `earthquakes_sf` with a new one that is a `Date`-class object (see `?as.Date()`). What is the most recent observation?

These lines of code read in the earthquake database and then removes duplicate events (rows that have the same date, time, and spatial coordinates). We also define a dataframe of just earthquakes above a certain magnitude. Then we created an sf object of the earthquakes contained in earthquakes_df.

```
earthquakes_sf$Date <- as.Date(earthquakes_sf$Date, format = "%m/%d/%y")
max(earthquakes_sf$Date)
```

```
## [1] "2016-05-01"
```

The most recent earthquake in our dataset was on May 1st, 2016.

(4) [3 pts] Verify that there were 2 earthquakes of magnitude at least 3 in the year 1999. How many earthquakes were there in 2002? How many in 2012? 2015?

```
t
```

```
## function (x)
## UseMethod("t")
## <bytecode: 0x128857708>
## <environment: namespace:base>
```

There were 1, 0, and 56 earthquakes in 2002, 2012, and 2015 respectively.

The file `ks_wells.RData` contains a data frame called `wells` with information about actively producing oil & gas wells in Kansas, also provided by the Kansas Geological Survey (see `?load()` for how to import it to your R environment). There are several attributes recorded for each well, including an individual identifier (`KID`), the date drilling began on the well (`SPUD_DATE`), the date the drilling was completed and production began (`COMPLETION` or `COMP_Date`), and the location of the well (`LONGITUDE`, `LATITUDE`).

```
load(file = "ks_wells.Rdata", envir = .GlobalEnv)
```

(5) [2 pts] When was the most recent well completed? Which year had the largest number of wells completed? How many wells were completed that year? **PEC** that create a new variable called `wells_sf` that is a simple features object containing the information about wells in Kansas with an appropriate coordinate reference system.

```r
max(wells$COMP_Date)
```

```
## [1] "2024-09-23"
```

```r
ann_wells <- wells %>%
  mutate(comp_year = year(COMP_Date)) %>%
  group_by(comp_year) %>%
  unique() %>%
  summarise(wells = n())

wells_sf <- st_as_sf(wells,
                     coords = c("LONGITUDE", "LATITUDE"),
                           crs = "+proj=longlat")
```

The most recent well was completed in September 2024. 2012 had the most wells completed, with a total of 6,027 wells.

(6) [2 pts] Make new versions of `kansas_sf`, `earthquakes_sf`, and `wells_sf` called `kansas_utm`, and `earthquakes_utm`, `wells_utm` that are projected to the Universal Transverse Mercator coordinate system, zone 13. **PEC**. Compare the bounding box for `kansas_utm` and the first few coordinate values in `earthquakes_utm` and `wells_utm` to the ones below to make sure you've transformed correctly (you might also want to try plotting all three objects on the same map).

```r
kansas_utm <- st_transform(kansas_sf, crs = "epsg:32613")
earthquakes_utm <- st_transform(earthquakes_sf, crs = "epsg:32613")
wells_utm <- st_transform(wells_sf, crs = "epsg:32613")
```

The following code plots the locations of all earthquakes detected in a given year (red asterisks) and all the locations of newly completed wells from the previous year (black circles). As you've probably already noticed in your explorations of these data, there has been a recent and dramatic increase in the number of earthquakes detected in Kansas.

```r
layout(matrix(1:20, 5, 4))
par(mar = c(1, 1, 1, 1))
for(year in 1996:2015){
  plot(kansas_utm$geometry, main = year + 1)
  plot(wells_utm$geometry[format(wells_utm$COMP_Date, "%Y") == year],
         col = scales::alpha("black", 0.1), add = T)
  plot(earthquakes_utm$geometry[format(earthquakes_utm$Date, "%Y") == year + 1],
         col = "darkred", pch = 8, add = T)
}
```

(7) [2 pts] From the explorations so far, do you anticipate any connection between the locations/numbers of new oil & gas wells and the location/numbers of earthquakes? Why or why not?

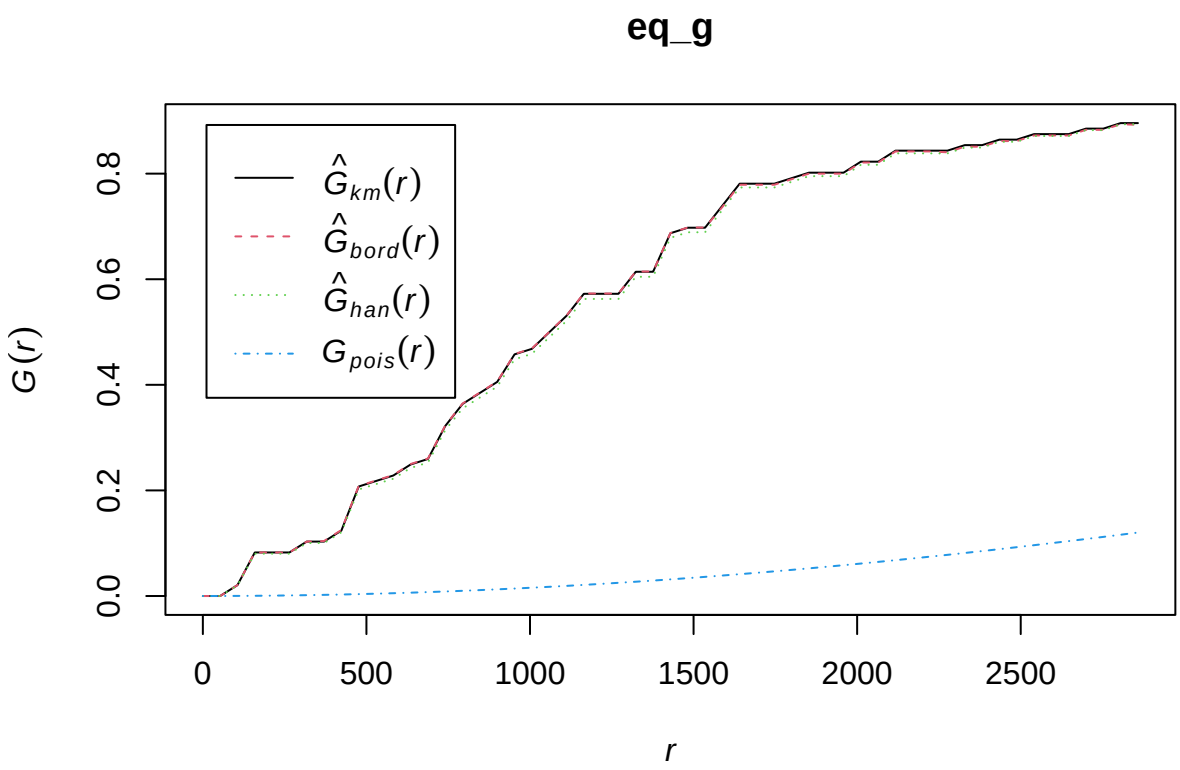
2012, 2013, and 2014 were the most productive years for new well completions, and 2013-2016 had the most large (>= 3 magnitude) earthquakes in these datasets. Considering this and the spatial distribution of earthquakes and wells plotted above, I'm already pretty convinced that these processes are linked. But this was also the first ever research topic I worked on when I was in high school!!

Moving forward, attention will focus on earthquakes recorded from 2014 onward and for wells completed between 2010–2013.

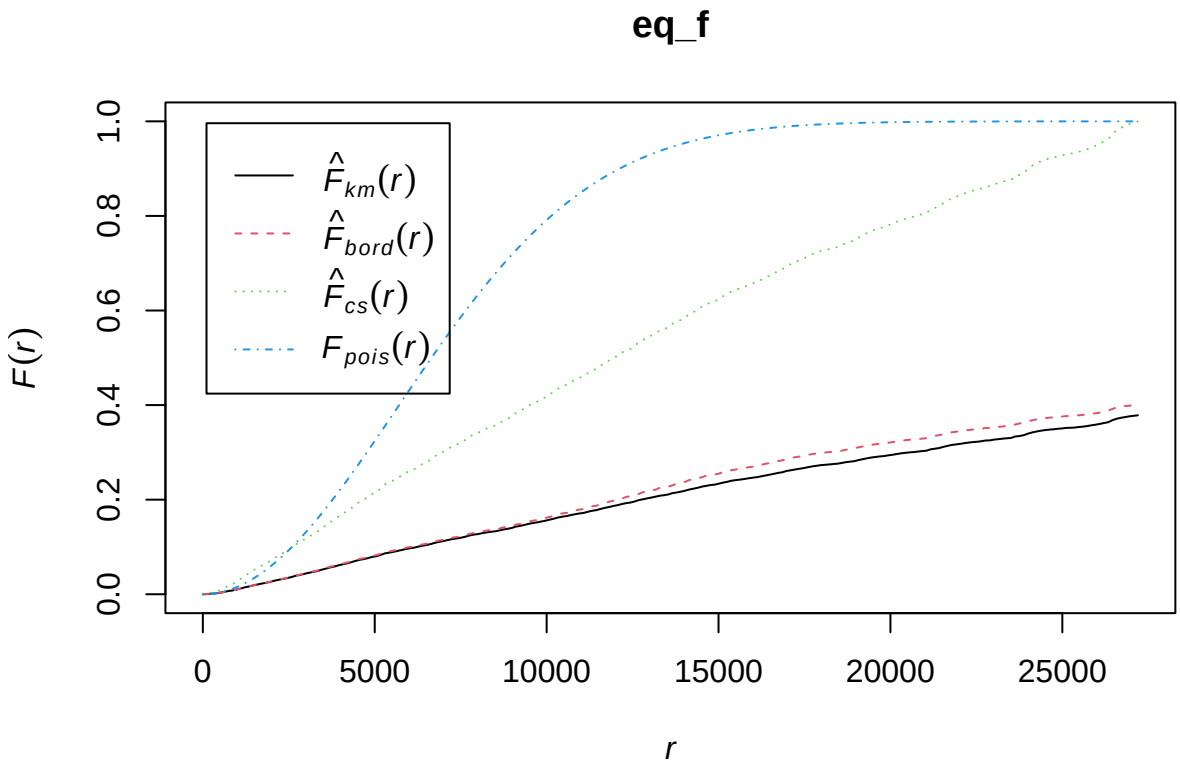
(8) [3 pts] **PEC** that creates an object called `earthquakes_ppp` using only earthquake records beginning in 2014. Verify your point pattern has 100 points. Compute the empirical $G(r)$ and $F(r)$ functions for the collection of all earthquake locations. Is it reasonable to conclude that the earthquakes arise according to a homogeneous Poisson process (HPP)? Why or why not?

```
earthquakes_ppp <- earthquakes_utm %>%
  filter(year(Date) >= 2014) %>%
  unique() %>%
  as.ppp()

library(spatstat)
eq_g <- Gest(earthquakes_ppp)
plot(eq_g)
```



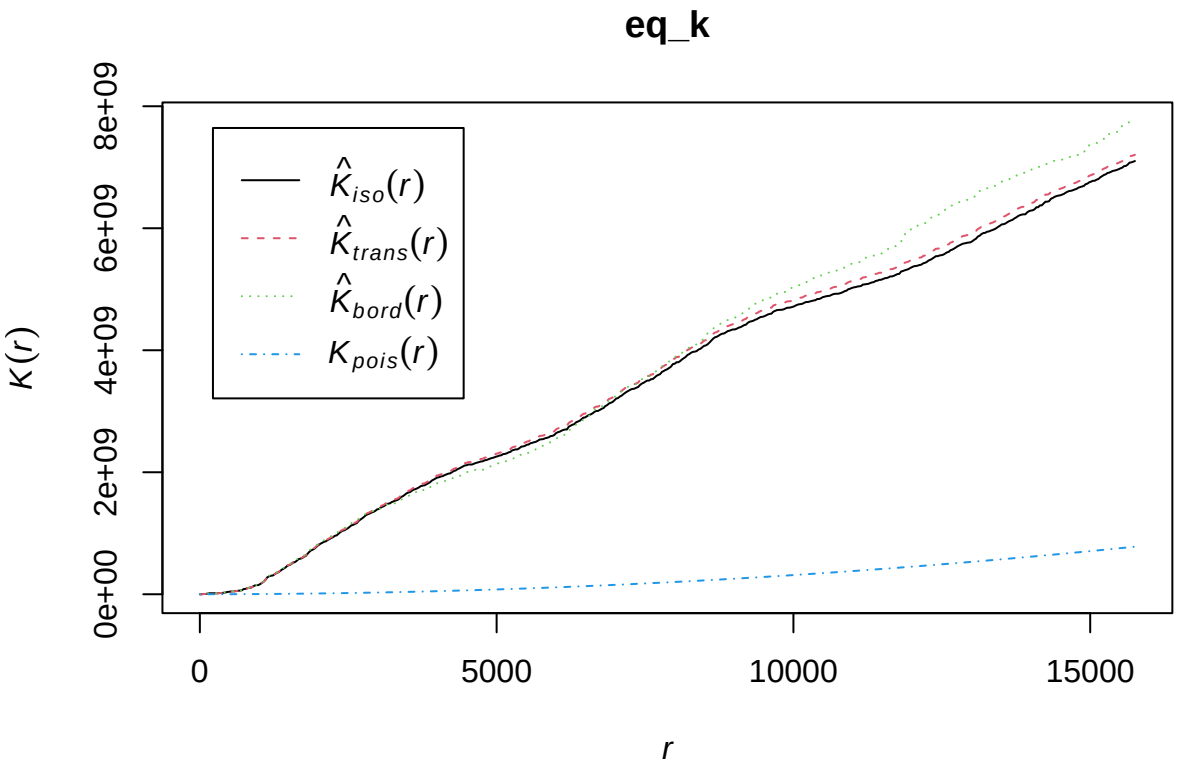

```
eq_f <- Fest(earthquakes_ppp)
plot(eq_f)
```



Based on our empirical G(r) and F(r) estimates, it would appear that our data are not arising from a HPP. G(r) shows that our points are more clustered than expected, and F(r) shows that there are fewer small empty spaces, also indicating clustering.

(9) [2 pts] Compute the empirical $K(r)$ function for the collection of all earthquake locations. How does it compare to the theoretical function for a completely spatially random (CSR) process?

```
eq_k <- Kest(earthquakes_ppp)
plot(eq_k)
```



Our empirical K function shows that there are more points within small radii around earthquakes than expected from a CSR process, indicating clustering of points and further confirming that our earthquake points did not arise from a HPP.

II. Models for earthquakes [18 pts]

To investigate whether there may be a connection between the locations of recently developed wells and the locations of earthquakes, one approach is to construct a model for the intensity function for the earthquake process as a function of the distance to the nearest well. The documentation for `?ppm()` explains that predictor variables in the trend formula can be specified in a few different ways, including as functions of spatial locations. You will create a function that represents a possible predictor of earthquakes and use it in a log-linear model for the intensity surface.

(10) [3 pts] Create a ppp object called `wells_ppp` that includes all wells completed anytime during 2010–2013. Verify there are 20099 records in the point pattern.

```
wells_ppp <- wells_utm %>%
  mutate(comp_year = year(COMP_Date)) %>%
  filter(comp_year %in% seq(2010, 2013, 1)) %>%
  as.ppp()
```

(11) [3 pts] Explain in your own words what the function `min_dist()` created in line 20 does. What value would `min_dist(wells_ppp[1])` return? Why?

```
20 min_dist <- distfun(X = wells_ppp)
```

min_dist represents the minimum distance to a well point from all points in the domain of wells_ppp. Running min_dist(wells_ppp[x]) for any x in 1:n gives you zero, since the minimum distance from any well to a well is 0 (i.e., the distance to itself).

(12) [3 pts] Fit a log-linear inhomogeneous Poisson process (IPP) model with `min_dist` as the sole predictor. Provide a 95% confidence interval for the effect of `min_dist` on the log-intensity surface. Do the values in the interval make sense to you? Why/why not? (Hint: if you get a warning about model fitting failing to converge, try using `method = "logi"` to use an alternative fitting algorithm.)

```
eq <- unmark(earthquakes_ppp)
well_mod1 <- ppm(eq ~ min_dist, method = "logi")
summary(well_mod1)
```

```
## Point process model
## Fitted to data: eq
## Fitting method: maximum likelihood (logistic regression approximation)
## Model was fitted using glm()
## Algorithm converged
## Call:
## ppm.formula(Q = eq ~ min_dist, method = "logi")
## Edge correction: "border"
##  [border correction distance r = 0 ]
## --------------------------------------------------------------------------------
## Quadrature scheme (logistic) = data + dummy
## Data pattern:
## Planar point pattern:  100 points
## Average intensity 4.99e-09 points per square unit
## Window: rectangle = [1034304.9, 1352363.6] x [4119010, 4182025] units
##                     (318100 x 63020 units)
## Window area = 20042500000 square units
##
##
## Dummy pattern:
## (Stratified random dummy points, 32 x 32 grid of cells)
## Planar point pattern:  1024 points
## Average intensity 5.11e-08 points per square unit
## Window: rectangle = [1034304.9, 1352363.6] x [4119010, 4182025] units
##                     (318100 x 63020 units)
## Window area = 20042500000 square units
## --------------------------------------------------------------------------------
## FITTED :
##
## Nonstationary Poisson process
##
## ---- Intensity: ----
##
## Log intensity: ~min_dist
## Model depends on external covariate 'min_dist'
## Covariates provided:
##  min_dist: distfun
```

```
##
## Fitted trend coefficients:
##   (Intercept)      min_dist
## -1.868326e+01 -1.415386e-04
##
##                  Estimate         S.E.       CI95.lo       CI95.hi Ztest
## (Intercept) -1.868326e+01 0.1658666932 -1.900835e+01 -1.835817e+01   ***
## min_dist    -1.415386e-04 0.0000489407 -2.374606e-04 -4.561659e-05    **
##                    Zval
## (Intercept) -112.640207
## min_dist      -2.892043
##
## ----------- gory details -----
##
## Fitted regular parameters (theta):
##   (Intercept)      min_dist
## -1.868326e+01 -1.415386e-04
##
## Fitted exp(theta):
##  (Intercept)     min_dist
## 7.690664e-09 9.998585e-01
```

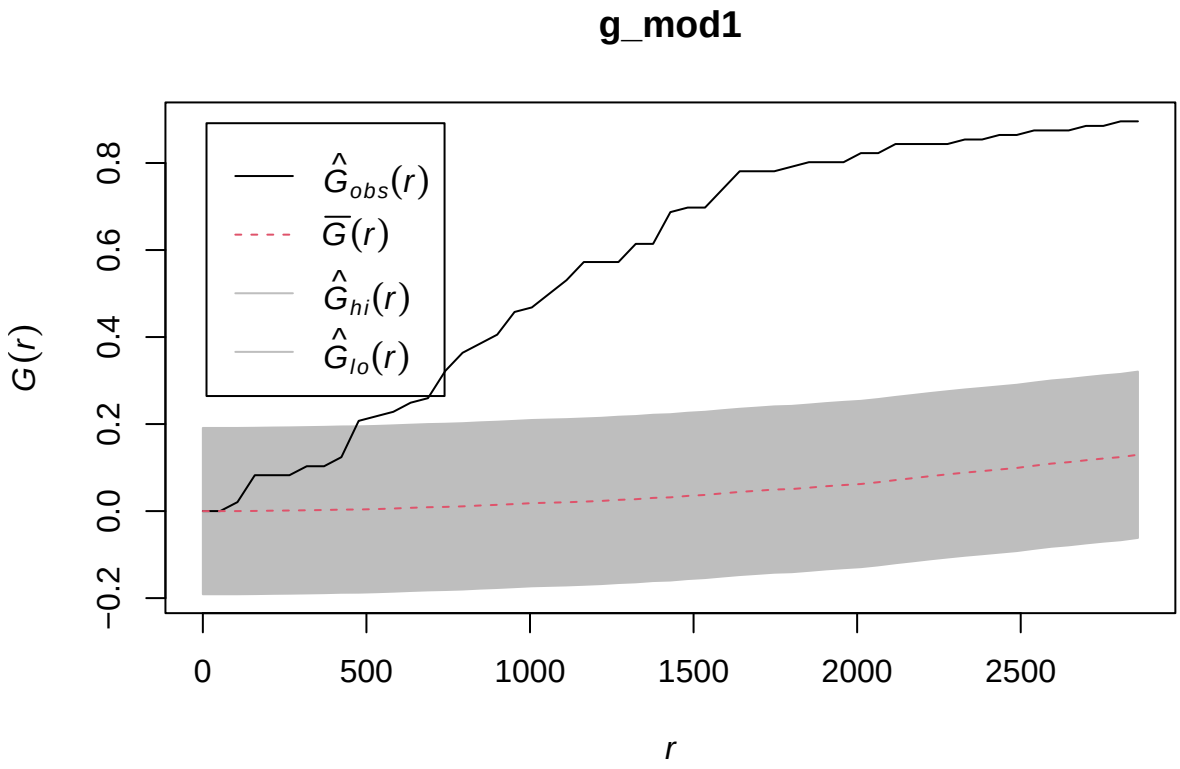
The effect of minimum distance to a well on the log intensity is negative, such that the log intensity decreases as minimum distance increases. The estimated CI ranges from -2.362 x 10^-4 to -4.317 x 10^-5. This makes sense if earthquakes are associated with well locations, because we would expect a higher intensity (i.e., potential for earthquakes) nearer to wells.

(13) [3 pts] Create simulation-based envelopes for the $G(r)$ and $F(r)$ functions implied by your fitted IPP from question (12) and compare them to the corresponding empirical functions for the observed point pattern in `earthquakes_ppp`. What do your figures suggest about the quality of the model fit?

```
g_mod1 <- envelope(well_mod1, fun = Gest, global = T)
```

```
## Generating 198 simulated realisations of fitted Poisson model (99 to estimate
## the mean and 99 to calculate envelopes) ...
## 1, 2, 3, 4.6.8.10.12.14.16.18.20.22.24.26.28.30.32.34
## .36.38.40.42.44.46.48.50.52.54.56.58.60.62.64.66.68.70.72.74
## .76.78.80.82.84.86.88.90.92.94.96.98.100.102.104.106.108.110.112.114
## .116.118.120.122.124.126.128.130.132.134.136.138.140.142.144.146.148.150.152.154
## .156.158.160.162.164.166.168.170.172.174.176.178.180.182.184.186.188.190.192.194
## .196.
## 198.
##
## Done.
```

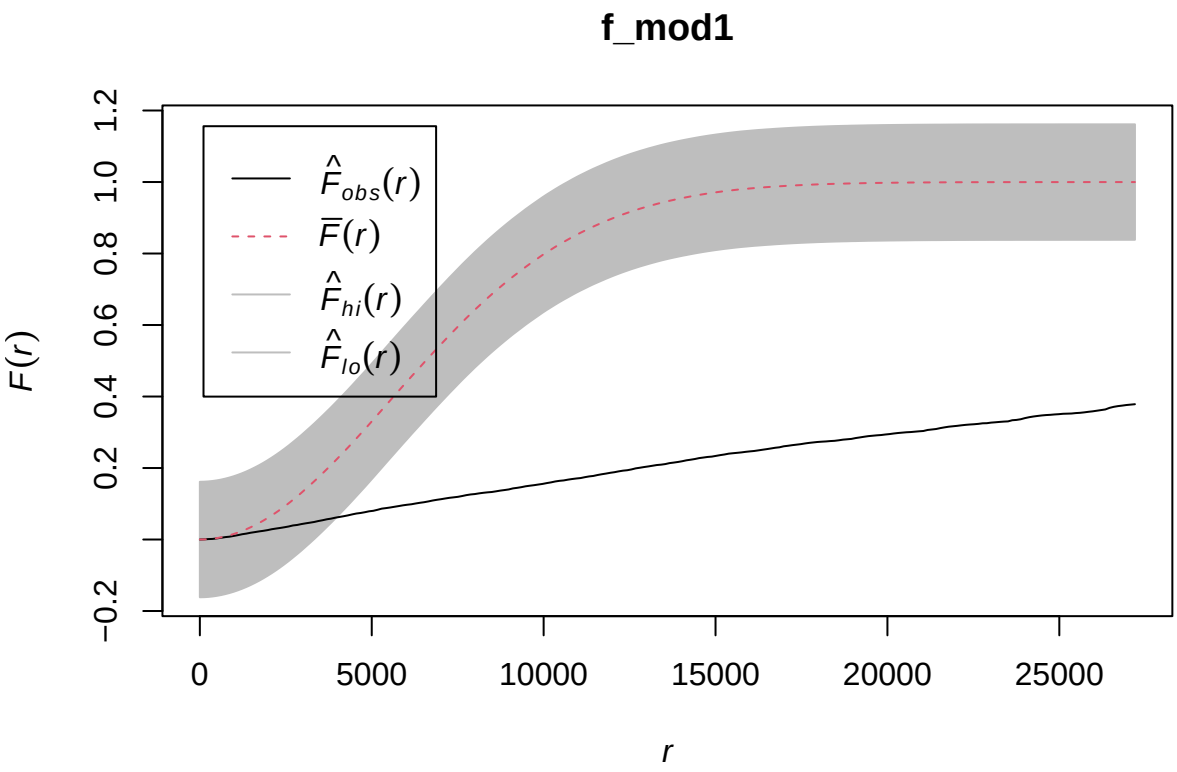
```
plot(g_mod1)
```




```
f_mod1 <- envelope(well_mod1, fun = Fest, global = T)
```

```
## Generating 198 simulated realisations of fitted Poisson model (99 to estimate
## the mean and 99 to calculate envelopes) ...
## 1, 2, 3, 4.6.8.10.12.14.16.18.20.22.24.26.28.30.32.34
## .36.38.40.42.44.46.48.50.52.54.56.58.60.62.64.66.68.70.72.74
## .76.78.80.82.84.86.88.90.92.94.96.98.100.102.104.106.108.110.112.114
## .116.118.120.122.124.126.128.130.132.134.136.138.140.142.144.146.148.150.152.154
## .156.158.160.162.164.166.168.170.172.174.176.178.180.182.184.186.188.190.192.194
## .196.
## 198.
##
## Done.
```

```
plot(f_mod1)
```



These figures suggest that our fitted model predicts fewer close neighbors than were observed in the data, and that our model is overestimating the amount of empty spaces. Our model is still probably underestimating the spatial clustering of these earthquakes.

(14) [3 pts] Fit a log-Gaussian Cox Process (LGCP) model to the earthquake point pattern with the same single predictor and an exponential covariance function. Report a 95% confidence interval for the effect of `min_dist`. How does your interval compare to the one from question (12)?

```
well_mod2 <- kppm(eq ~ min_dist, clusters = "LGCP", model = "exponential")
```

```
## Warning: data contain duplicated points
```

```
summary(well_mod2)
```

```
## Inhomogeneous Cox point process model
## Fitted to point pattern dataset 'eq'
## Fitted by minimum contrast
##  Summary statistic: inhomogeneous K-function
## Minimum contrast fit (object of class "minconfit")
## Model: Log-Gaussian Cox process
##   Covariance model: exponential
## Fitted by matching theoretical K function to eq
##
## Internal parameters fitted by minimum contrast ($par):
##       sigma2        alpha
```

```
##     4.419574 10646.235922
##
## Fitted covariance parameters:
##          var        scale
##     4.419574 10646.235922
## Fitted mean of log of random intensity:  [pixel image]
##
## Converged successfully after 61 function evaluations
##
## Starting values of parameters:
##   sigma2    alpha
##    1.000 6870.746
## Domain of integration: [ 0 , 15750 ]
## Exponents: p= 2, q= 0.25
##
## ----------- TREND  -----
## Point process model
## Fitted to data: X
## Fitting method: maximum likelihood (Berman-Turner approximation)
## Model was fitted using glm()
## Algorithm converged
## Call:
## ppm.ppp(Q = X, trend = trend, rename.intercept = FALSE, covariates = covariates,
##     covfunargs = covfunargs, use.gam = use.gam, forcefit = TRUE,
##     improve.type = ppm.improve.type, improve.args = ppm.improve.args,
##     nd = nd, eps = eps)
## Edge correction: "border"
##  [border correction distance r = 0 ]
## --------------------------------------------------------------------------------
## Quadrature scheme (Berman-Turner) = data + dummy + weights
##
## Data pattern:
## Planar point pattern:  100 points
## Average intensity 4.99e-09 points per square unit
## Window: rectangle = [1034304.9, 1352363.6] x [4119010, 4182025] units
##                     (318100 x 63020 units)
## Window area = 20042500000 square units
##
## Dummy quadrature points:
##      32 x 32 grid of dummy points, plus 4 corner points
##      dummy spacing: 9939.333 x 1969.223 units
##
## Original dummy parameters: =
## Planar point pattern:  1028 points
## Average intensity 5.13e-08 points per square unit
## Window: rectangle = [1034304.9, 1352363.6] x [4119010, 4182025] units
##                     (318100 x 63020 units)
## Window area = 20042500000 square units
## Quadrature weights:
##      (counting weights based on 32 x 32 array of rectangular tiles)
## All weights:
##  range: [1960000, 19600000]  total: 2e+10
## Weights on data points:
##  range: [1960000, 9790000]   total: 5.15e+08
```

```
## Weights on dummy points:
##  range: [1960000, 19600000]  total: 1.95e+10
## --------------------------------------------------------------------------------
## FITTED :
##
## Nonstationary Poisson process
##
## ---- Intensity: ----
##
## Log intensity: ~min_dist
## Model depends on external covariate 'min_dist'
## Covariates provided:
##  min_dist: distfun
##
## Fitted trend coefficients:
##   (Intercept)      min_dist
## -1.871558e+01 -1.316879e-04
##
##                  Estimate         S.E.       CI95.lo       CI95.hi Ztest
## (Intercept) -1.871558e+01 1.620078e-01 -1.903311e+01 -1.839805e+01   ***
## min_dist    -1.316879e-04 4.746359e-05 -2.247149e-04 -3.866099e-05    **
##                    Zval
## (Intercept) -115.522688
## min_dist      -2.774504
##
## ----------- gory details -----
##
## Fitted regular parameters (theta):
##   (Intercept)      min_dist
## -1.871558e+01 -1.316879e-04
##
## Fitted exp(theta):
##  (Intercept)     min_dist
## 7.446082e-09 9.998683e-01
##
## ----------- COX  -----------
## Model: log-Gaussian Cox process
##
##  Covariance model: exponential
## Fitted covariance parameters:
##          var        scale
##     4.419574 10646.235922
## Fitted mean of log of random intensity: [pixel image]
##
## Final standard error and CI
## (allowing for correlation of Cox process):
##                  Estimate        S.E.       CI95.lo       CI95.hi Ztest
## (Intercept) -1.871558e+01 0.895355111 -2.047044e+01 -1.696071e+01   ***
## min_dist    -1.316879e-04 0.000185444 -4.951514e-04  2.317755e-04
##                    Zval
## (Intercept) -20.9029665
## min_dist     -0.7101225
```

Here, our confidence interval overlaps with zero, indicating that there was no evidence for an effect of minimum

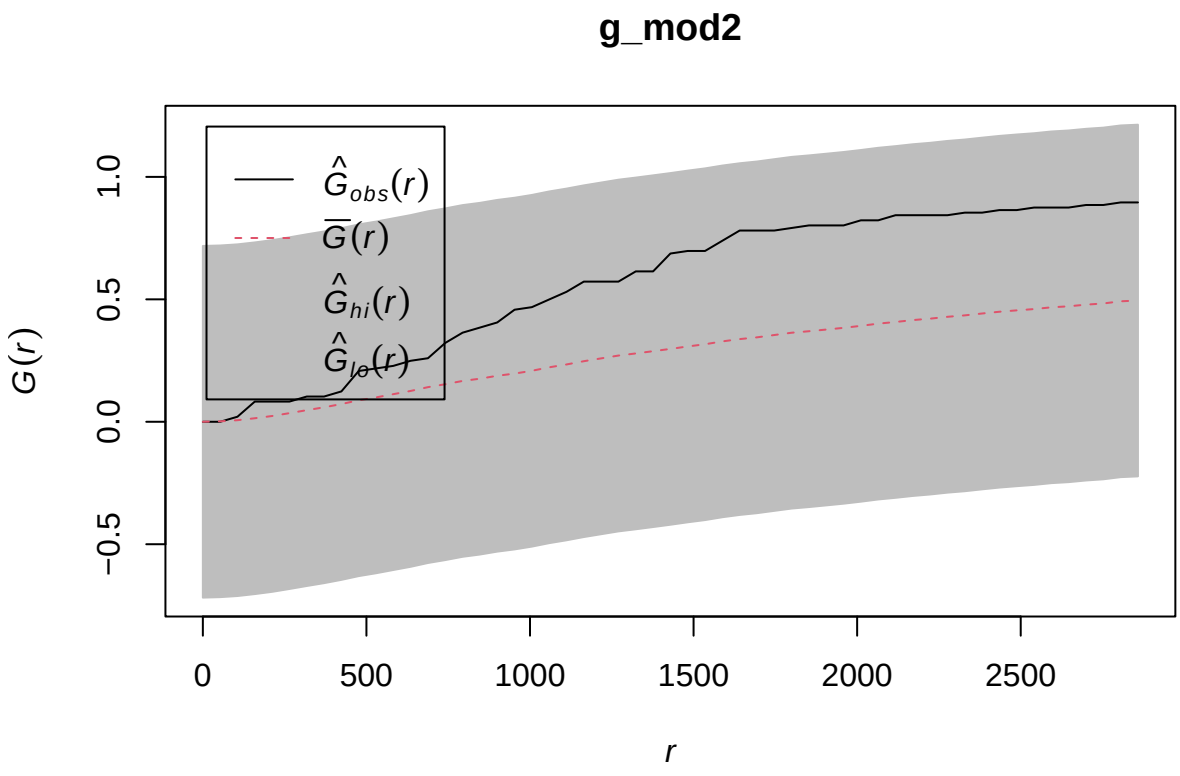
distance after accounting for spatial dependence in the LGCP model.

(15) [3 pts] Create simulation-based envelopes for the $G(r)$ and $F(r)$ functions implied by your fitted LGCP from question (14) and compare them to the corresponding empirical functions for the observed point pattern in `earthquakes_ppp`. What do your figures suggest about the quality of the model fit?

```
g_mod2 <- envelope(well_mod2, fun = Gest, global = T)
```

```
## Generating 198 simulated realisations of fitted cluster model (99 to estimate
## the mean and 99 to calculate envelopes) ...
## 1, 2, 3, 4.6.8.10.12.14.16.18.20.22.24.26.28.30.32.34
## .36.38.40.42.44.46.48.50.52.54.56.58.60.62.64.66.68.70.72.74
## .76.78.80.82.84.86.88.90.92.94.96.98.100.102.104.106.108.110.112.114
## .116.118.120.122.124.126.128.130.132.134.136.138.140.142.144.146.148.150.152.154
## .156.158.160.162.164.166.168.170.172.174.176.178.180.182.184.186.188.190.192.194
## .196.
## 198.
##
## Done.
```

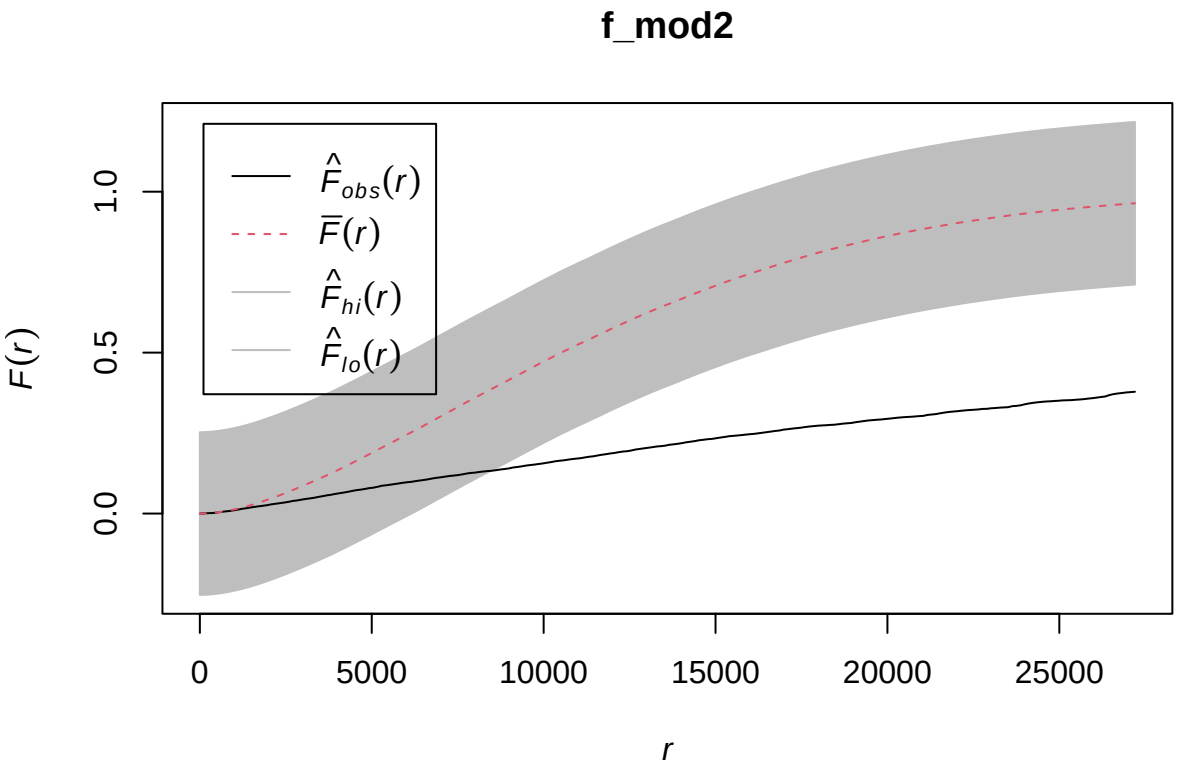
```
plot(g_mod2)
```



```
f_mod2 <- envelope(well_mod2, fun = Fest, global = T)
```

```
## Generating 198 simulated realisations of fitted cluster model (99 to estimate
## the mean and 99 to calculate envelopes) ...
## 1, 2, 3, 4.6.8.10.12.14.16.18.20.22.24.26.28.30.32.34
## .36.38.40.42.44.46.48.50.52.54.56.58.60.62.64.66.68.70.72.74
## .76.78.80.82.84.86.88.90.92.94.96.98.100.102.104.106.108.110.112.114
## .116.118.120.122.124.126.128.130.132.134.136.138.140.142.144.146.148.150.152.154
## .156.158.160.162.164.166.168.170.172.174.176.178.180.182.184.186.188.190.192.194
## .196.
## 198.
##
## Done.
```

```
plot(f_mod2)
```



These figures show that our new LGCP model more accurately captures the density of neighbors in the observations. This model is still not performing well in estimating empty spaces, predicting more small empty spaces than expected from a Poisson process.