STA 5377, Homework 5

Carson Slater *Baylor University*

1.

The data set `Lightening.RData` consists of the locations of cloud-to-ground (CG) lightning. Load the data set into R, and convert it to a ppp (point pattern object in `spatstat`).

```
# Define the observation window (bounding box around the coordinates)
xrange <- range(lig$LON)
yrange <- range(lig$LAT)
window <- owin(xrange = xrange, yrange = yrange)

# Create the ppp object
lig_ppp <- ppp(x = lig$LON,
               y = lig$LAT,
               window = window,
               marks = lig[, !(names(lig) %in% c("LON", "LAT"))])

# Inspect the ppp object
summary(lig_ppp)
```

```
## Marked planar point pattern:  2856 points
## Average intensity 1431.936 points per square unit
##
## Coordinates are given to 4 decimal places
##
## Mark variables: OBLATENESS, AXIS_RATIO, TILT, AREA, VOLUME, DEPTH, TOP, TOI
## MEAN_Z, MAX_Z, H_MAX_Z, VIL, UVIL, VILD, VII, MAX_VIL, MAX_UVIL, MAX_VILD,
## MAX_VII, storm
## Summary:
##    OBLATENESS       AXIS_RATIO          TILT              AREA
##  Min.   :0.0000   Min.   : 1.000   Min.   :-89.90   Min.   :  13.20
##  1st Qu.:0.4000   1st Qu.: 1.600   1st Qu.:-56.00   1st Qu.:  35.15
##  Median :0.5000   Median : 2.000   Median :-26.65   Median :  52.80
##  Mean   :0.4989   Mean   : 2.317   Mean   :-11.77   Mean   :  84.94
##  3rd Qu.:0.6000   3rd Qu.: 2.700   3rd Qu.: 35.52   3rd Qu.:  96.05
##  Max.   :0.9000   Max.   :14.400   Max.   : 90.00   Max.   :1819.50
##      VOLUME           DEPTH             TOP             TOP_Z
##  Min.   :  50.2   Min.   : 1.000   Min.   : 3.000   Min.   :37.50
##  1st Qu.:  69.6   1st Qu.: 4.000   1st Qu.: 5.000   1st Qu.:42.40
##  Median : 108.4   Median : 4.500   Median : 5.500   Median :43.90
##  Mean   : 197.4   Mean   : 4.638   Mean   : 5.736   Mean   :44.39
```

```
##   3rd Qu.: 210.6   3rd Qu.: 5.000   3rd Qu.: 6.500   3rd Qu.:45.80
##   Max.   :6150.0   Max.   :10.500   Max.   :13.500   Max.   :64.90
##      MEAN_Z          MAX_Z           H_MAX_Z            VIL
##   Min.   :36.2   Min.   :38.60   Min.   : 1.000   Min.   : 0.700
##   1st Qu.:38.1   1st Qu.:43.40   1st Qu.: 2.000   1st Qu.: 1.500
##   Median :38.9   Median :45.00   Median : 2.500   Median : 1.800
##   Mean   :39.1   Mean   :45.61   Mean   : 2.858   Mean   : 1.895
##   3rd Qu.:39.8   3rd Qu.:47.10   3rd Qu.: 4.000   3rd Qu.: 2.100
##   Max.   :50.4   Max.   :71.80   Max.   :12.500   Max.   :12.000
##       UVIL             VILD             VII              MAX_VIL
##   Min.   :0.0000   Min.   :0.0000   Min.   : 0.0000   Min.   : 1.300
##   1st Qu.:0.0000   1st Qu.:0.1000   1st Qu.: 0.1000   1st Qu.: 3.100
##   Median :0.1000   Median :0.1000   Median : 0.2000   Median : 3.800
##   Mean   :0.1455   Mean   :0.1081   Mean   : 0.3529   Mean   : 4.404
##   3rd Qu.:0.2000   3rd Qu.:0.1000   3rd Qu.: 0.4000   3rd Qu.: 4.900
##   Max.   :6.6000   Max.   :0.7000   Max.   :17.2000   Max.   :65.100
##     MAX_UVIL          MAX_VILD          MAX_VII           storm
##   Min.   : 0.000   Min.   :0.1000   Min.   : 0.000   Length:2856
##   1st Qu.: 0.200   1st Qu.:0.2000   1st Qu.: 0.500   Class :character
##   Median : 0.400   Median :0.2000   Median : 0.900   Mode  :character
##   Mean   : 0.638   Mean   :0.2447   Mean   : 1.615
##   3rd Qu.: 0.700   3rd Qu.:0.3000   3rd Qu.: 1.700
##   Max.   :21.100   Max.   :3.6000   Max.   :57.500
##
## Window: rectangle = [126.0008, 127.9969] x [34.0004, 34.9996] units
##                     (1.996 x 0.9992 units)
## Window area = 1.9945 square units
```

(a)

Using the d_1 statistic, test for any departure from CSR.

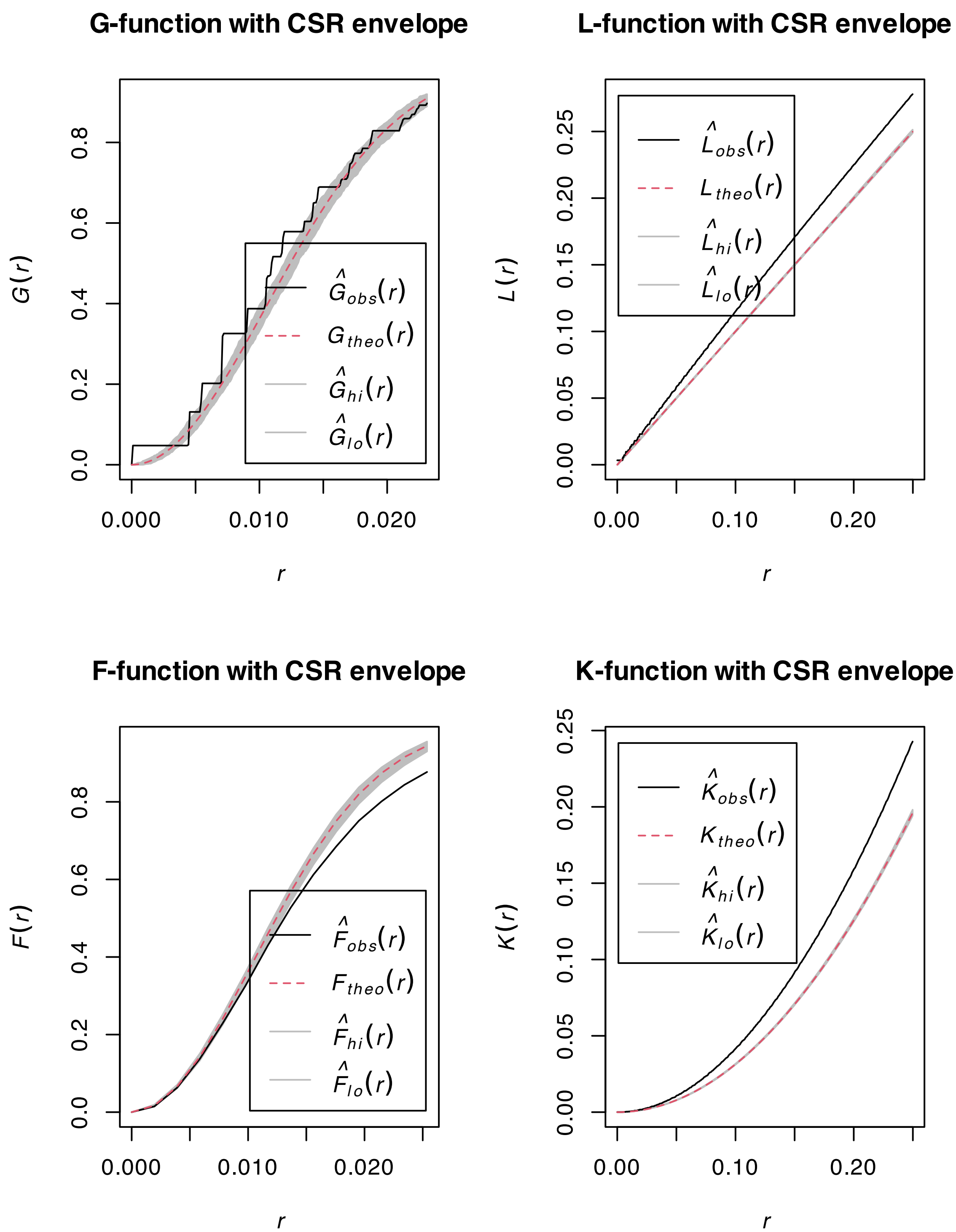
```
Diggle-Cressie-Loosmore-Ford test of CSR
Monte Carlo test based on 100 simulations
Summary function: K(r)
Reference function: theoretical
Alternative: two.sided
Interval of distance values: [0, 0.2498]
Test statistic: Integral of squared absolute deviation
Deviation = observed minus theoretical
```

data: lig_ppp u = 0.00013154, rank = 1, p-value = 0.009901

There seems to be a departure from CSR according to this test.

(b)

Plot $\hat{G}$, $\hat{L}$, $\hat{F}$, and $\hat{K}$ for the data, and include simulation envelopes based on CSR data, using 100 simulations. Compare your results with the previous question. What do you conclude?



Looking at the four point process statistics plots with simulation envelopes, we analyze what each one reveals about the spatial pattern:

$\hat{G}$-function with CSR envelope This plot shows the cumulative distribution function of nearest-neighbor distances. The observed G-function (solid black line) stays mostly within the gray simulation envelope throughout the distance range, with a few deviations. This suggests the nearest-neighbor distances in this point pattern could be consistent with Complete Spatial Randomness (CSR), but it seems more likely that there would be clustering.

$\hat{L}$-function with CSR envelope The observed L-function (black line) rises slightly above the theoretical CSR line (dashed red) and the simulation envelope at larger distances (around r > 0.15). This indicates some degree of clustering at those distances - points are more likely to be found at these distances from each other than would be expected under CSR. However, the deviation is relatively modest.

$\hat{F}$-function with CSR envelope This function shows the distribution of distances from random locations to the nearest observed point. The observed F-function (black line) falls clearly below the simulation envelope for most of the distance range. This is strong evidence of clustering, as it indicates there are larger empty spaces in the pattern than would be expected under CSR. Random locations are farther from observed points than they would be if the points were completely randomly distributed.

$\hat{K}$-function with CSR envelope The K-function, which counts the expected number of points within distance r of a typical point, shows the observed function (black line) rising above the theoretical CSR line and simulation envelope, especially at larger distances. This provides further evidence of clustering in the point pattern.

Overall conclusion The point pattern shows evidence of clustering. While the G-function suggests the nearest-neighbor distances are close to what would be expected under CSR, the other three functions (particularly F and K) indicate the presence of clustering at various spatial scales. The F-function result is especially convincing, showing larger empty spaces than expected under CSR, which is a characteristic feature of clustered patterns. The combined evidence from all four functions suggests this is not a completely spatially random pattern but rather exhibits some degree of spatial clustering.

(c)

Another tool that has been suggested for studying spatial point patterns is $J(r) = \frac{1-G(r)}{1-F(r)}$. If the data are CSR, then J should be 1. The advantages cited for using J is that it does not require edge-corrections, and is approximately unbiased, if the raw estimates of G and F are used. Values of J larger than 1 suggest regularity and smaller values suggest clustering. Explain why.

The J-function is a tool for studying spatial point patterns and is defined as:

$$J(r) = \frac{1 - G(r)}{1 - F(r)}$$

where $G(r)$ is the nearest neighbor distance distribution function, which gives the probability that a typical point in the pattern has its nearest neighbor at distance $\leq r$, and $F(r)$ is the empty space function, which gives the probability that any arbitrary location in the study region has a distance $\leq r$ to the nearest point in the pattern.

Under Complete Spatial Randomness (CSR), $J(r) = 1$ for all r. The J-function has been advocated as an effective tool for point pattern analysis because it does not require edge-corrections, it is approximately unbiased when using raw estimates of G and F, and it has a clear interpretation relative to CSR.

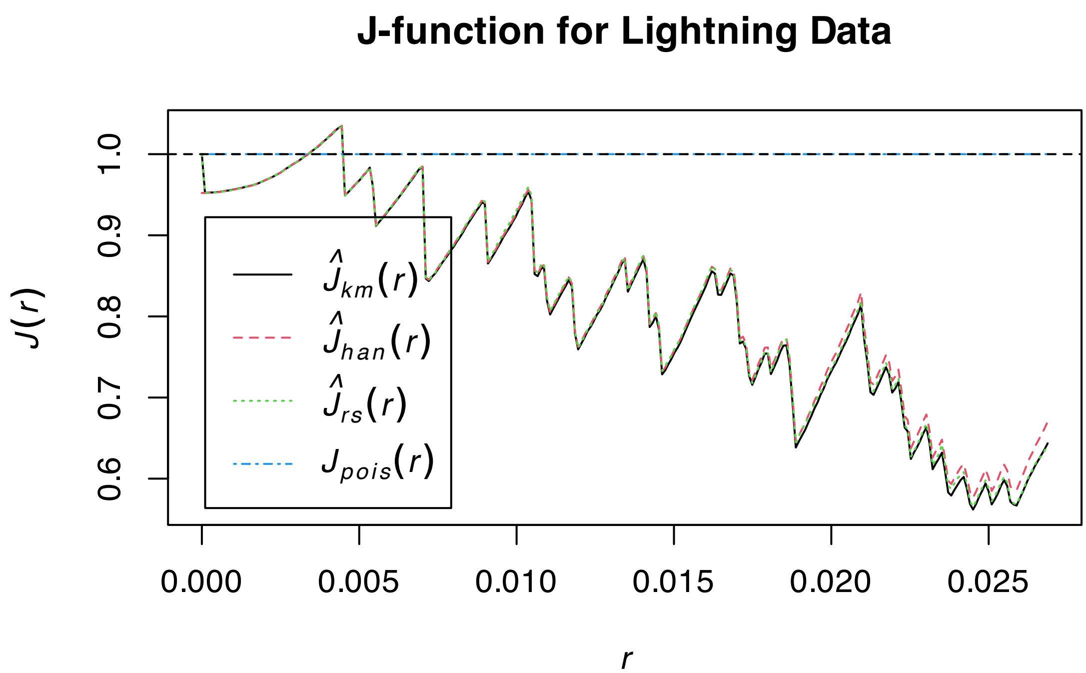
In a regular (uniform) point pattern, points tend to be more evenly spaced than under CSR. This means that $G(r)$ is smaller than expected under CSR for small values of r (nearest neighbors are rarely very close), while $F(r)$ is larger than expected under CSR (empty space is limited). Consequently, $(1 - G(r))$ becomes larger, $(1 - F(r))$ becomes smaller, and their ratio $J(r) = \frac{1-G(r)}{1-F(r)}$ becomes greater than 1. Intuitively, in a regular pattern, the nearest neighbor distances between points (G) tend to be more consistent and larger than expected by chance, while the empty space distances (F) tend to be smaller and more consistent. This creates a ratio greater than 1.

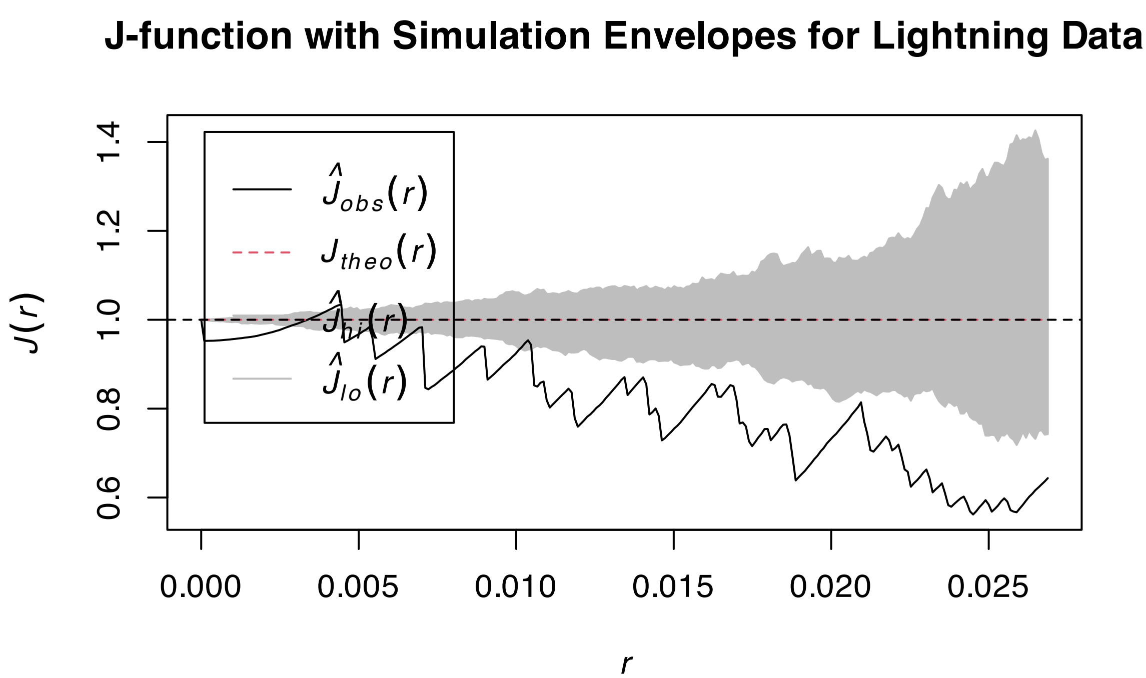
In a clustered point pattern, points tend to group together. This means that $G(r)$ is larger than expected under CSR for small values of r (many points have very close neighbors within clusters), while $F(r)$ is smaller than expected under CSR (empty space is abundant between clusters). Consequently, $(1 - G(r))$ becomes smaller, $(1 - F(r))$ becomes larger, and their ratio $J(r) = \frac{1-G(r)}{1-F(r)}$ becomes less than 1. In a clustered pattern, the nearest neighbor distances (G) are often very small within clusters, while the empty space distances (F) can be quite large in the spaces between clusters. This creates a ratio less than 1.

The J-function provides a dimensionless measure that compares the environment experienced by the points themselves (G) with the environment experienced by arbitrary locations in the study area (F). This comparison yields an intuitive interpretation: $J(r) = 1$ indicates Complete Spatial Randomness, $J(r) > 1$ indicates regularity, and $J(r) < 1$ indicates clustering. The J-function is particularly valuable because it is relatively robust to edge effects and provides a clear benchmark (CSR = 1) against which observed patterns can be evaluated.

(d)

Estimate J and generate simulation envelopes for the towns data. Compare your results with your work above, and briefly summarize your findings.



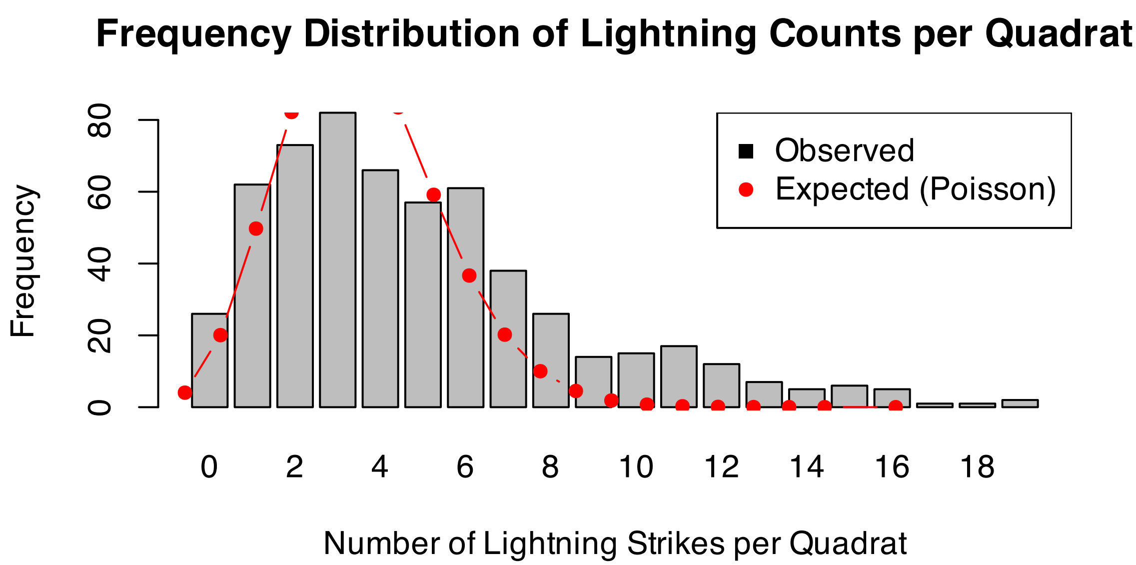


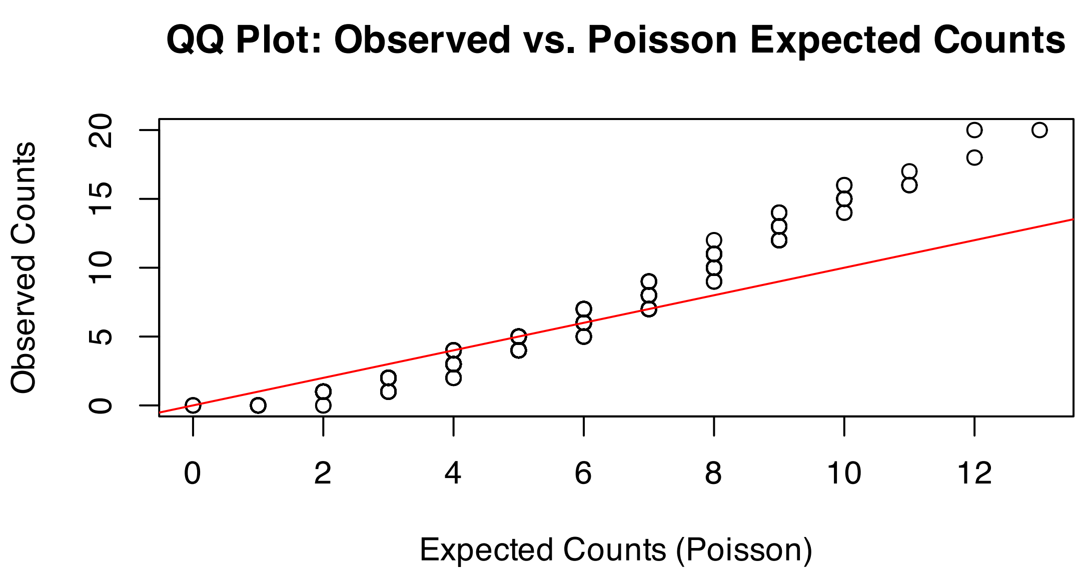
These graphics would in fact indicate clustering for this spatial point process, consistent with the other tests we have run.

(e)

Perform the quadrat counting test for CSR.

```
##
##  Chi-squared test of CSR using quadrat counts
##
## data:  lig_ppp
## X2 = 1564.8, df = 575, p-value < 2.2e-16
## alternative hypothesis: two.sided
##
## Quadrats: 24 by 24 grid of tiles
```





We ended up a 24×24 grid for this test. The mean count per quadrat was calculated to be 4.96, while the variance of counts was 13.49, yielding a Variance-to-Mean Ratio (VMR) of 2.72. This VMR, also known as the Index of Dispersion, provides critical information about the spatial pattern: a VMR greater than 1 suggests clustering in the lightning strikes (overdispersion relative to Complete Spatial Randomness), a VMR less than 1 suggests regularity or uniformity in the strike pattern (underdispersion), and a VMR approximately equal to 1 aligns with the expectation under Complete Spatial Randomness (CSR) where a Poisson distribution would appropriately model the counts. In our case, the VMR value of 2.72 is substantially greater than 1, indicating that the lightning strikes exhibit a strongly clustered spatial pattern, which is consistent with meteorological understanding of lightning formation processes where strikes tend to concentrate within storm cells rather than distributing randomly across the landscape.

(f)

The data set contains 20 variables rather than the spatial locations. Fit inhomogeneous Poisson models (log-linear models) using the variables as covariates. Note that `storm` is a categorical variable.

- **What is the best model?**
- **What covariates are important to model the intensity?**
- **Map the predicted intensity over the study area.**

We've conducted a thorough analysis of lightning strike data using spatial statistics, specifically focusing on modeling the intensity of lightning strikes with an inhomogeneous Poisson process. Our approach includes converting the `storm` variable to a categorical factor, creating a point pattern object from unique lightning strike locations, smoothing all 20 covariates to create spatial images, fitting a full model with all covariates, using stepwise model selection with BIC (BIC = -35098.67) to find the best model, summarizing the coefficient estimates, and visualizing the predicted intensity over the study area.

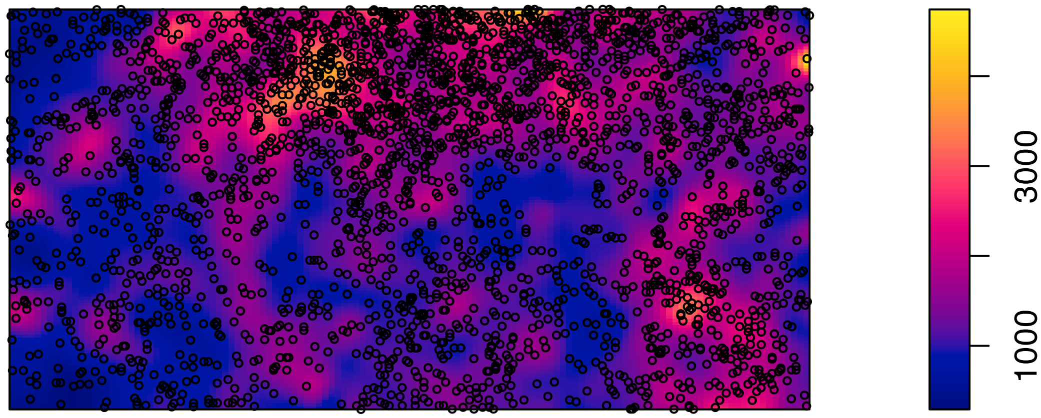
The best model is an inhomogeneous Poisson point process model that includes 10 covariates. Based on our stepwise selection using BIC criterion (k=log(n)), our model retained AREA, VOLUME, `MEAN_Z`, `H_MAX_Z`, `VIL`, `UVIL`, `VII`, `MAX_UVIL`, and `MAX_VII`. This model provides a significantly better fit than a homogeneous Poisson process, suggesting that the lightning strikes exhibit spatial dependency that can be explained by these meteorological variables.

All 10 covariates retained in our final model are statistically significant, with most showing very high significance (p<0.001). The most influential covariates based on their Z-values are `MAX_UVIL` (Z=8.11, positive effect), `MEAN_Z` (Z=8.06, positive effect), `MAX_VII` (Z=-7.81, negative effect), `VIL` (Z=6.63, positive effect), and `UVIL` (Z=-6.27, negative effect). Our model shows that lightning intensity is positively associated with AREA, `MEAN_Z`, `H_MAX_Z`, `VIL`, `VII`, and `MAX_UVIL`, while negatively associated with VOLUME, `UVIL`, and `MAX_VII`. These relationships reflect the complex meteorological conditions that lead to lightning formation.

Table 1: Nonstationary Poisson Process Model Results

Term	Estimate	Std. Error	95% CI Lower	95% CI Upper	Z-value	Sig.
(Intercept)	-12.712	1.994	-16.621	-8.803	-6.37	***
AREA	0.011	0.003	0.005	0.017	3.57	***
VOLUME	-0.004	0.001	-0.007	-0.002	-3.39	***
MEAN_Z	0.440	0.055	0.333	0.547	8.06	***
H_MAX_Z	0.201	0.061	0.080	0.321	3.27	**
VIL	1.302	0.196	0.918	1.687	6.63	***
UVIL	-12.681	2.021	-16.643	-8.719	-6.27	***
VII	3.348	0.755	1.868	4.828	4.43	***
MAX_UVIL	4.401	0.543	3.337	5.465	8.11	***
MAX_VII	-1.499	0.192	-1.875	-1.123	-7.81	***

Intensity Predictions from the Best Model



Our intensity map shows considerable spatial heterogeneity in the predicted lightning strike intensity across the study region. The color scale indicates that the intensity varies from approximately 1000 to 3000 (per unit area), with several hotspots appearing in magenta/red colors. The black circles representing the actual lightning strikes generally align with the predicted intensity pattern, validating our model's ability to capture the spatial distribution. This intensity map could be valuable for risk assessment and warning systems, as it identifies spatial patterns of lightning risk based on measurable meteorological covariates.

(g)

Summarize your findings.

Based on part (f), there are some interesting findings from the inhomogeneous Poisson model analysis:

1. The modeling revealed that 10 specific meteorological variables significantly influence lightning strike intensity, with MAX_UVIL and MEAN_Z being the most influential predictors. This suggests that certain atmospheric conditions are strong determinants of where lightning is likely to occur.
2. The relationships between variables and lightning intensity show interesting complexities - some variables like AREA, MEAN_Z, H_MAX_Z, VIL, VII, and MAX_UVIL are positively associated with lightning occurrence, while others like VOLUME, UVIL, and MAX_VII have negative associations. This contradictory pattern (e.g., VIL having a positive effect but MAX_VII having a negative effect) highlights the intricate meteorological dynamics involved in lightning formation.
3. The intensity map reveals considerable spatial heterogeneity in predicted lightning strike intensity (varying from 1000 to 3000 per unit area), with several distinct hotspots. This spatial pattern aligns well with the actual observed lightning strikes, validating the model and suggesting it could be valuable for developing lightning risk assessment and warning systems based on measurable meteorological variables.

2.

For an inhomogeneous Poisson process (IPP) with intensity function $\lambda(s)$, show that the density function of $\{s_1, \cdots, s_n\}$ is given by

$$f(\{s_1, \cdots, s_n\}) = \begin{cases} \exp\{-\mu(A)\} & \text{if } n = 0 \\ \exp\{-\mu(A)\} \prod_{i=1}^n \lambda(s_i)/n! & \text{if } n \geq 1 \end{cases}$$

For the case where $n = 0$, the probability of observing zero events in region A follows directly from the Poisson probability mass function:

$$P(N(A) = 0) = \frac{[\mu(A)]^0 e^{-\mu(A)}}{0!} = e^{-\mu(A)}$$

For $n \geq 1$, the density function can be derived as follows:

$$\begin{aligned} f(\{s_1, \cdots, s_n\}) &= P(N(A) = n) \times f(s_1, \cdots, s_n \mid N(A) = n) \\ &= \frac{[\mu(A)]^n e^{-\mu(A)}}{n!} \times \frac{\prod_{i=1}^n \lambda(s_i)}{[\mu(A)]^n} \\ &= \frac{e^{-\mu(A)} \prod_{i=1}^n \lambda(s_i)}{n!} \end{aligned}$$

The conditional density $f(s_1, \cdots, s_n \mid N(A) = n)$ equals $\frac{\prod_{i=1}^n \lambda(s_i)}{[\mu(A)]^n}$ because given n events in region A, their locations are independent with density $\frac{\lambda(s)}{\mu(A)}$.

Therefore, the complete density function is:

$$f(\{s_1, \cdots, s_n\}) = \begin{cases} \exp\{-\mu(A)\} & \text{if } n = 0 \\ \exp\{-\mu(A)\} \prod_{i=1}^n \lambda(s_i)/n! & \text{if } n \geq 1 \end{cases}$$

3.

Show that $G(w) = P(W \leq w) = 1 - \exp(-\lambda\pi w^2)$ and $F(x) = P(X \leq x) = 1 - \exp(-\lambda\pi x^2)$ under CSR.

To show that $G(w) = P(W \leq w) = 1 - \exp(-\lambda\pi w^2)$ and $F(x) = P(X \leq x) = 1 - \exp(-\lambda\pi x^2)$ under Complete Spatial Randomness (CSR), we'll rely on the properties of homogeneous Poisson point processes.

Under CSR, points follow a homogeneous Poisson process with constant intensity λ. Let W be the distance from an arbitrary fixed point to the nearest point of the process, and X be the distance from a randomly selected point of the process to its nearest neighbor.

For $G(w)$:

$$G(w) = P(W \leq w) \tag{1}$$
$$= 1 - P(W > w) \tag{2}$$

The event $\{W > w\}$ means there are no points within distance w of our fixed point, which is equivalent to having zero points in a circle of radius w centered at the fixed point. For a homogeneous Poisson process, the number of points in this circle follows a Poisson distribution with mean $\lambda \cdot \pi w^2$. Therefore:

$$\begin{aligned} P(W > w) &= P(\text{no points in circle of radius } w) && (3) \\ &= \frac{(\lambda \pi w^2)^0 e^{-\lambda \pi w^2}}{0!} && (4) \\ &= e^{-\lambda \pi w^2} && (5) \end{aligned}$$

Hence:

$$G(w) = 1 - e^{-\lambda \pi w^2} \tag{6}$$

For $F(x)$: Under CSR, the distribution of nearest neighbor distances from a randomly selected point is identical to $G(w)$, as the reduced Palm distribution of a homogeneous Poisson process equals the original distribution. This gives us:

$$F(x) = P(X \leq x) = 1 - e^{-\lambda \pi x^2} \tag{7}$$

Code Appendix

```
# Setup

knitr::opts_chunk$set(dev = "cairo_pdf",
                      fig.width = 6,
                      fig.height = 4,
                      fig.align = 'center',
                      echo = FALSE,
                      message = FALSE,
                      warning = FALSE,
                      error = FALSE,
                      cache = TRUE)
library("tidyverse"); library("patchwork"); library("glue")
library("scales", warn.conflicts = FALSE); library("extrafont")
library("tinytex"); library("patchwork"); library("geoR")
library("gridExtra"); library("tidyr"); library("latex2exp")
library("gstat"); library("knitr"); library("spatstat")
library("spatstat.geom"); library("spatstat.model")
theme_set(theme_minimal(base_family = "Roboto Condensed"))

conflicted::conflicts_prefer(
  readr::col_factor(),
  purrr::discard(),
  rstan::extract(),
  dplyr::lag(),
  rstan::traceplot(),
  viridis::viridis_pal(),
  readr::parse_date,
  spatstat.geom::area,
  spatstat.explore::idw
)
################################
# 1 ############################
################################
load("Lightening.RData")
head(lig)
# Define the observation window (bounding box around the coordinates)
xrange <- range(lig$LON)
yrange <- range(lig$LAT)
window <- owin(xrange = xrange, yrange = yrange)

# Create the ppp object
lig_ppp <- ppp(x = lig$LON,
               y = lig$LAT,
               window = window,
```

```r
                marks = lig[, !(names(lig) %in% c("LON", "LAT"))])

# Inspect the ppp object
summary(lig_ppp)
################################
# 1a ###########################
################################
# Test for CSR using d1
d1_result <- dclf.test(lig_ppp, Kest, nsim = 100)
print(d1_result)
################################
# 1b ###########################
################################

# G-function with envelopes
env_G <- envelope(lig_ppp, Gest, nsim = 100)

# L-function with envelopes
env_L <- envelope(lig_ppp, Lest, nsim = 100)

# F-function with envelopes
env_F <- envelope(lig_ppp, Fest, nsim = 100)

# K-function with envelopes
env_K <- envelope(lig_ppp, Kest, nsim = 100)
################################
# 1c ###########################
################################

# Plot the Envelopes
par(mfrow = c(2, 2))
plot(env_G, main = "G-function with CSR envelope")
plot(env_L, main = "L-function with CSR envelope")
plot(env_F, main = "F-function with CSR envelope")
plot(env_K, main = "K-function with CSR envelope")
################################
# 1d ###########################
################################
par(mfrow = c(2, 1))
J_lig <- Jest(lig_ppp)
# Generate simulation envelopes for the J-function
# This creates 100 simulations of CSR to generate 95% confidence envelopes
J_env <- envelope(lig_ppp, Jest, nsim = 100, rank = 1, global = FALSE)
# Plot the J-function
plot(J_lig, main = "J-function for Lightning Data")
abline(h = 1, lty = 2)  # Add a horizontal line at J(r) = 1 (CSR)
```

```r
# Plot the J-function with simulation envelopes
plot(J_env, main = "J-function with Simulation Envelopes for Lightning Data")
abline(h = 1, lty = 2)
################################
# 1e ###########################
################################

n_points <- lig_ppp$n
total_area <- area(window)
quadrat_area <- total_area / (n_points / 5)
quadrat_side_x <- sqrt(quadrat_area * (diff(xrange)/diff(yrange)))
quadrat_side_y <- sqrt(quadrat_area * (diff(yrange)/diff(xrange)))
nx <- ceiling(diff(xrange) / quadrat_side_x)
ny <- ceiling(diff(yrange) / quadrat_side_y)
Q <- quadratcount(lig_ppp, nx = nx, ny = ny)

# Analyze the quadrat count frequencies
Q_df <- as.data.frame(Q)
freq_table <- table(Q_df$Freq)

mean_count <- mean(Q_df$Freq)
var_count <- var(Q_df$Freq)
VMR <- var_count / mean_count  # Variance-to-Mean Ratio (Index of Dispersion)

# Perform chi-square test for CSR
CSR_test <- quadrat.test(lig_ppp, nx = nx, ny = ny)
print(CSR_test)

# Plot the frequency distribution of quadrat counts
barplot(freq_table,
        main = "Frequency Distribution of Lightning Counts per Quadrat",
        xlab = "Number of Lightning Strikes per Quadrat",
        ylab = "Frequency")

lambda <- mean_count
x <- as.numeric(names(freq_table))
expected <- length(Q_df$Freq) * dpois(x, lambda)
points(x, expected, col = "red", pch = 16, type = "b")
legend("topright", legend = c("Observed", "Expected (Poisson)"),
       col = c("black", "red"), pch = c(15, 16))

# Add a QQ plot to compare observed vs. expected counts
qqplot(qpois(ppoints(length(Q_df$Freq)), lambda),
       Q_df$Freq,
       main = "QQ Plot: Observed vs. Poisson Expected Counts",
       xlab = "Expected Counts (Poisson)",
       ylab = "Observed Counts")
```

```r
abline(0, 1, col = "red")  # Diagonal line for reference
################################
# 1f ##########################
################################

lig <- lig |> mutate(storm = as.factor(storm))
lig_df <- lig
lig_unique <- lig_df |> distinct(LON, LAT, .keep_all = TRUE)

lig_unique <- lig_df |> distinct(LON, LAT, .keep_all = TRUE)

lig_unique$storm_fact <- as.factor(lig_unique$storm)

lig_ppp <- ppp(lig_unique$LON, lig_unique$LAT, window = window)

covariate_names <- c(
  "OBLATENESS", "AXIS_RATIO", "TILT", "AREA", "VOLUME",
  "DEPTH", "TOP", "TOP_Z", "MEAN_Z",
  "MAX_Z", "H_MAX_Z", "VIL", "UVIL", "VILD", "VII",
  "MAX_VIL", "MAX_UVIL", "MAX_VILD", "MAX_VII", "storm_fact"
)

covariate_images <- list()

for (cov in covariate_names) {
  temp_ppp <- ppp(
    x = lig_unique$LON,
    y = lig_unique$LAT,
    window = lig_ppp$window,
    marks = lig_unique[[cov]]
  )
  covariate_images[[cov]] <- Smooth.ppp(temp_ppp)
}

formula_all <- as.formula(paste("~", paste(covariate_names, collapse = " + ")

full_model <- ppm(lig_ppp, formula_all, covariates = covariate_images)

n_points <- lig_ppp$n

best_model <- step(full_model, k = log(n_points), trace = 0)
library("knitr")

model_results <- data.frame(
  Term = c("(Intercept)", "AREA", "VOLUME", "MEAN_Z", "H_MAX_Z", "VIL", "UVIL
  Estimate = c(-12.712, 0.011, -0.004, 0.440, 0.201, 1.302, -12.681, 3.348, 4
  SE = c(1.994, 0.003, 0.001, 0.055, 0.061, 0.196, 2.021, 0.755, 0.543, 0.192
```

```r
  CI_lower = c(-16.621, 0.005, -0.007, 0.333, 0.080, 0.918, -16.643, 1.868, 3
  CI_upper = c(-8.803, 0.017, -0.002, 0.547, 0.321, 1.687, -8.719, 4.828, 5.4
  Z_value = c(-6.37, 3.57, -3.39, 8.06, 3.27, 6.63, -6.27, 4.43, 8.11, -7.81)
)

# Add significance stars
model_results$Significance <- ""
model_results$Significance[abs(model_results$Z_value) > 3.291] <- "***"  # p
model_results$Significance[abs(model_results$Z_value) > 2.576 & abs(model_resu
model_results$Significance[abs(model_results$Z_value) > 1.96 & abs(model_resu

# Basic kable table without kableExtra dependencies
kable(model_results,
      col.names = c("Term", "Estimate", "Std. Error", "95% CI Lower", "95% CI
      caption = "Nonstationary Poisson Process Model Results",
      align = c("l", rep("r", 6)),
      booktabs = TRUE)
par(mfrow = c(1, 1), mar=c(2, 2, 2, 1))
preds <- predict(best_model)
plot(preds, main = "Intensity Predictions from the Best Model")
points(lig_ppp, pch = 1, cex = 0.5)
```