

HW5: Non-Gaussian spatial data (generalized responses)

Fern Bromley

November 21, 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 [6 pts]

The goal of the following exercise is to show that sometimes log-linear inhomogeneous Poisson processes and Poisson GLMs are equivalent probabilistic models. The equivalence occurs when the areal units used to aggregate a point pattern into counts (e.g., the central park squirrels example) match up with constant values of the predictors.

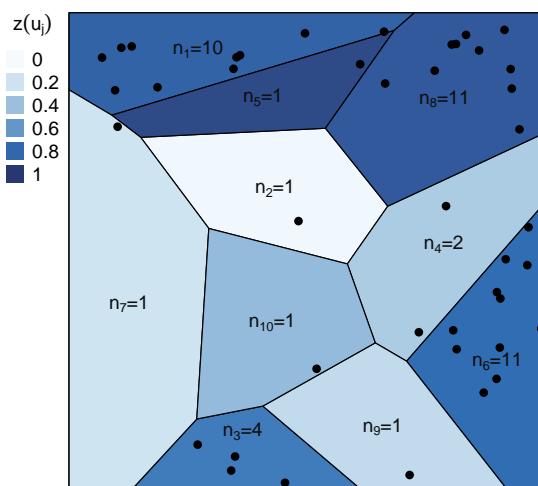


Figure 1: Example domain and partition. Points in the pattern \mathbf{X} are marked by solid dots and counts of points in each subset, $n_j = n(u_j)$, are written at the center of each component of the partition. The color of each partition corresponds to the level of a predictor variable, $z(u_j)$.

(1) [6 pts] Consider a point process, $X(u)$, which takes place in a finite domain W that is partitioned into J compact sets. Let u_j , $j = 1, \dots, J$ denote the j th set of the partition such that $\cup_{j=1}^J u_j = W$ and $u_j \cap u_{j^*} = \emptyset$ for $j \neq j^*$. Consider also the aggregated process $n(u_j) = \sum_{i=1}^n \mathbf{1}_{x_i \in u_j}$ that is the sum of points in any finite realization of $\mathbf{X} = \{x_1, \dots, x_n\}$ occurring in each of the J subsets of the partition. Suppose a predictor of interest, $Z(u)$, is piece-wise constant with value z_j across each component of the partition such that $Z(u) = z_j \forall u \in u_j$. Figure 1 shows an example.

(Option 1) Show mathematically that the likelihoods for the log-linear inhomogeneous Poisson process for \mathbf{X} given by $\log(\lambda(u)) = \beta_0 + \beta_1 Z(u)$ and the GLM for \mathbf{n} with a Poisson response distribution, log link, and log-expected value $\log(\lambda(u_j)) = \beta_0 + z(u_j)\beta_1 + \log(|u_j|)$ are proportional to each other up to a constant that does not depend on $\beta = (\beta_0 \ \beta_1)^\top$, which therefore implies that the MLE estimators of β are the same for each model (*hint: an expression for the likelihood of a log-linear IPP is provided in BRT Ch. 9.7.4*). Note, the extra term in the GLM, $\log(|u|)$ is not a predictor and has no coefficient. It is a known quantity sometimes called an “offset”.

(Option 2) The file HW5_1.RData contains the realized point pattern \mathbf{X} from Figure 1 above and a pixel-image Z_{pix} that provides the value of the predictor $z(u)$ at a fine grid of locations in the plane $u = (x, y)$. Use the objects to fit a log-linear model for the point pattern and estimate the coefficients β_0 and β_1 . HW5_1.RData also contains the data frame named aggregated with columns z and area corresponding to $z(u_j)$ and $|u_j|$, respectively. Use the data frame to fit a GLM for the counts, \mathbf{n} , with a Poisson response distribution, log link function, and offset $\log(|u_j|)$. Verify these two approaches give approximately the same estimates for the coefficients β_0 and β_1 .

```

1 load("HW5_1.RData")
2 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

## 
## Attaching package: 'spatstat.model'

## The following object is masked from 'package:spmodel':
## 
##     pseudoR2

## Loading required package: spatstat.linnet

## spatstat.linnet 3.3-2

## 
## spatstat 3.4-1
## For an introduction to spatstat, type 'beginner'

1 logmod1 <- ppm(X ~ Z_pix)
2 summary(logmod1)

## Point process model
## Fitted to data: X
## Fitting method: maximum likelihood (Berman-Turner approximation)
## Model was fitted using glm()
## Algorithm converged
## Call:
## ppm.formula(Q = X ~ Z_pix)
## Edge correction: "border"
## [border correction distance r = 0 ]
##
## Quadrature scheme (Berman-Turner) = data + dummy + weights
##
## Data pattern:
## Planar point pattern: 43 points
## Average intensity 43 points per square unit
## Window: rectangle = [0, 1] x [0, 1] units
## Window area = 1 square unit
##
## Dummy quadrature points:
##     32 x 32 grid of dummy points, plus 4 corner points
##     dummy spacing: 0.03125 units
##
## Original dummy parameters: =
## Planar point pattern: 1028 points
## Average intensity 1030 points per square unit
## Window: rectangle = [0, 1] x [0, 1] units
## Window area = 1 square unit

```

```

## Quadrature weights:
##   (counting weights based on 32 x 32 array of rectangular tiles)
## All weights:
##   range: [0.000326, 0.000977] total: 1
## Weights on data points:
##   range: [0.000326, 0.000488] total: 0.0207
## Weights on dummy points:
##   range: [0.000326, 0.000977] total: 0.979
## -----
## FITTED :
##
## Nonstationary Poisson process
##
## ---- Intensity: ----
##
## Log intensity: ~Z_pix
## Model depends on external covariate 'Z_pix'
## Covariates provided:
##   Z_pix: im
##
## Fitted trend coefficients:
## (Intercept)      Z_pix
## 1.772969     3.180419
##
##           Estimate      S.E.  CI95.lo  CI95.hi Ztest      Zval
## (Intercept) 1.772969 0.5281679 0.7377788 2.808159 *** 3.356828
## Z_pix       3.180419 0.6902209 1.8276110 4.533227 *** 4.607828
##
## ----- gory details -----
##
## Fitted regular parameters (theta):
## (Intercept)      Z_pix
## 1.772969     3.180419
##
## Fitted exp(theta):
## (Intercept)      Z_pix
## 5.888309    24.056836

```

```

1 glm_mod <- glm(n ~ z + offset(log(area)), aggregated, family = poisson)
2 summary(glm_mod)

```

```

##
## Call:
## glm(formula = n ~ z + offset(log(area)), family = poisson, data = aggregated)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 1.7710    0.5266   3.363 0.000771 ***
## z          3.1865    0.6880   4.632 3.63e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##

```

```

##      Null deviance: 41.956  on 9  degrees of freedom
## Residual deviance: 12.350  on 8  degrees of freedom
## AIC: 44.889
##
## Number of Fisher Scoring iterations: 5

```

Forest Inventory and Analysis

The data for this homework come from the US Forest Service's Forest Inventory and Analysis (FIA) program, which "collects, processes, analyzes, and reports on data necessary for assessing the extent and condition of forest resources in the United States" (<https://research.fs.usda.gov/programs/fia>). The data are publicly available, and access can be facilitated by the rFIA package. Two other resources I used to prepare the data for this assignment were (1) the USFS list of tree species and codes <https://usfs-public.box.com/v/FIA-TreeSpeciesList>), and (2) metadata about the meanings of the variables appearing in the FIA records. (<https://research.fs.usda.gov/understory/forest-inventory-and-analysis-database-user-guide-nfi>). You don't need to know about the data pre-processing to be able to complete this assignment, but I'm happy to share more details about what I did for those who are interested.

The file `FIA_AZ_Douglas-Fir.csv` contains records about **Douglas Fir** trees in Arizona at the scale of FIA ground plots.

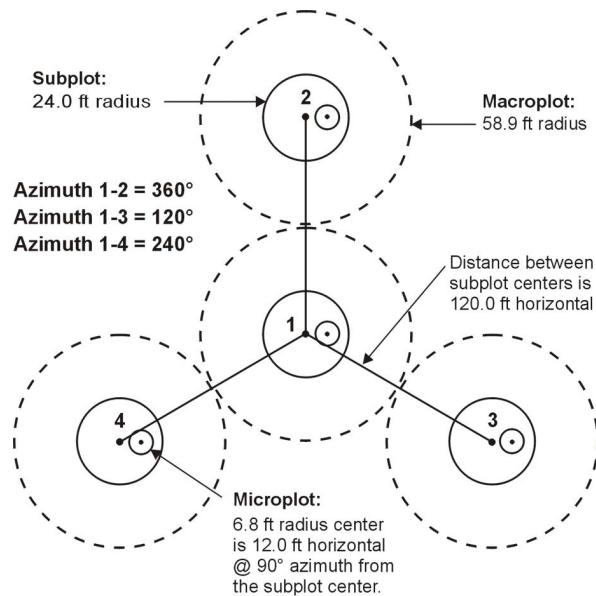


Figure 2: FIA sub-plot structure

Variable name	Definition
PLOT	Plot ID
INVYR	Inventory year. The year the data for the plot were collected.
Common.Name	Common name for tree species of interest: Douglas fir.
DEAD	Number of newly dead (since last site visit) Douglas fir trees recorded at plot.
ELEV	Elevation in feet at plot coordinates.
LON, LAT	Longitude and latitude of plot.

Figure 2 is taken from the metadata document referenced above and shows how four sub-plots are arranged within each plot. All the sub-plots associated with a given plot have the same recorded coordinates and elevation, and

the rows in FIA_AZ_Douglas-Fir.csv refer to observations aggregated to the plot level. A primary goal for this homework is to try and understand patterns in the deaths of Douglas Fir trees attributable to time and elevation.

II. Modeling counts [23 pts]

(2) [5 pts] Make a plot of the FIA plot locations with:

- (i) map tiles in the background for context,
- (ii) points colored by the number of dead trees recorded in a given year (you can include multiple inventory years for the same FIA plot as overlapping points), and
- (iii) points scaled in size by the number of dead trees recorded. You may need to experiment with the relationship between point size and number of dead trees to get a reasonable figure, but make sure larger points correspond to more dead trees.

```
1 library(tidyverse)
2 library(ggmap)
3 library(sf)
4 library(viridis)
5
6 plot_csv <- read.csv("FIA_AZ_Douglas-Fir.csv")
7 plots <- plot_csv %>%
8   group_by(LAT, LON, PLOT) %>%
9   summarise(DEAD = sum(DEAD))

## `summarise()` has grouped output by 'LAT', 'LON'. You can override using the
## `.`groups` argument.

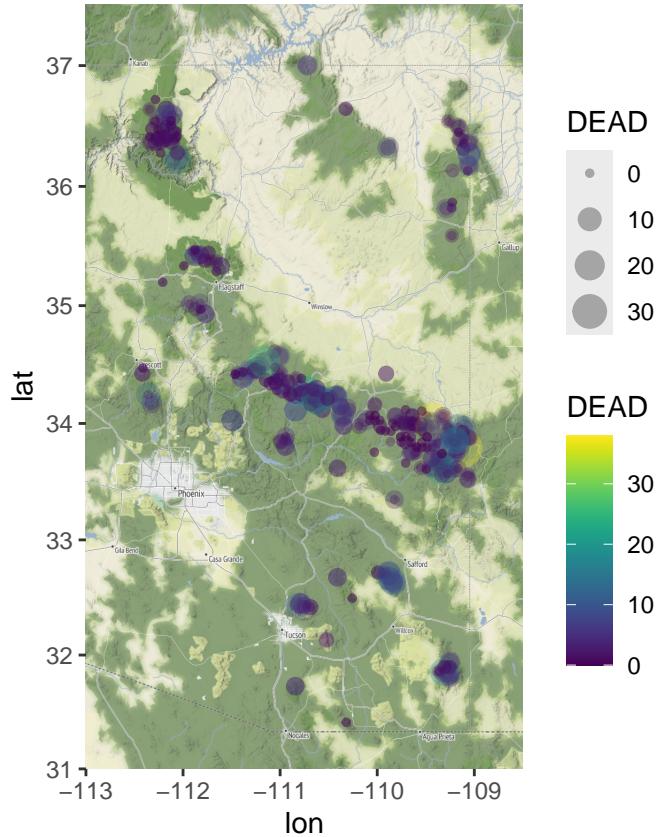
1 plot_loc <- st_as_sf(plot_csv, coords = c("LON", "LAT"), crs = "+proj=longlat")
2
3 st_bbox(plot_loc)
```

```
##           xmin         ymin         xmax         ymax
## -112.41988  31.40994 -109.05352  37.00189

1 fia_map <- get_stadiamap(bbox = c(left = -113,
2                                     bottom = 31,
3                                     right = -108.5,
4                                     top = 37.5),
5                                     zoom = 8)
```

i © Stadia Maps © Stamen Design © OpenMapTiles © OpenStreetMap contributors.

```
1 ggmap(fia_map) +
2   geom_point(data = plot_csv, aes(x = LON, y = LAT,
3                                     color = DEAD, size = DEAD),
4             alpha = 0.3) +
5   scale_color_viridis(option = "D")
```



- (3) [3 pts] Explain in your own words why a geostatistical model with a conditionally Gaussian response is inappropriate for the number of dead trees at each plot.

Because dead trees are recorded as counts, we should be modeling these assuming a Poisson distribution for the response.

- (4) [3 pts] Fit a generalized linear model (GLM) with a Poisson response distribution and log link function to the number of dead trees. Include linear relationships for elevation and year, as well as their interaction. Report 95% confidence intervals for each estimated effect. What is the predicted number of dead trees for a plot at an elevation of 7,000 ft in the year 2024?

```

1 library(spmmodel)
2 glm1 <- sglm(DEAD ~ INVYR + ELEV + INVYR:ELEV, data = plot_csv,
3               family = poisson,
4               spcov_type = "none")
5 confint(glm1)

```

```

##                   2.5 %      97.5 %
## (Intercept) 1.908171e+02 4.761903e+02
## INVYR        -2.363664e-01 -9.456062e-02
## ELEV         -7.472039e-02 -3.810758e-02
## INVYR:ELEV   1.896148e-05  3.715311e-05

```

```

1 exp(predict(glm1, newdata = data.frame(INVYR = 2024, ELEV = 7000), interval = "confidence"))

##          fit      lwr      upr
## 1 3.399008 2.913927 3.964841

```

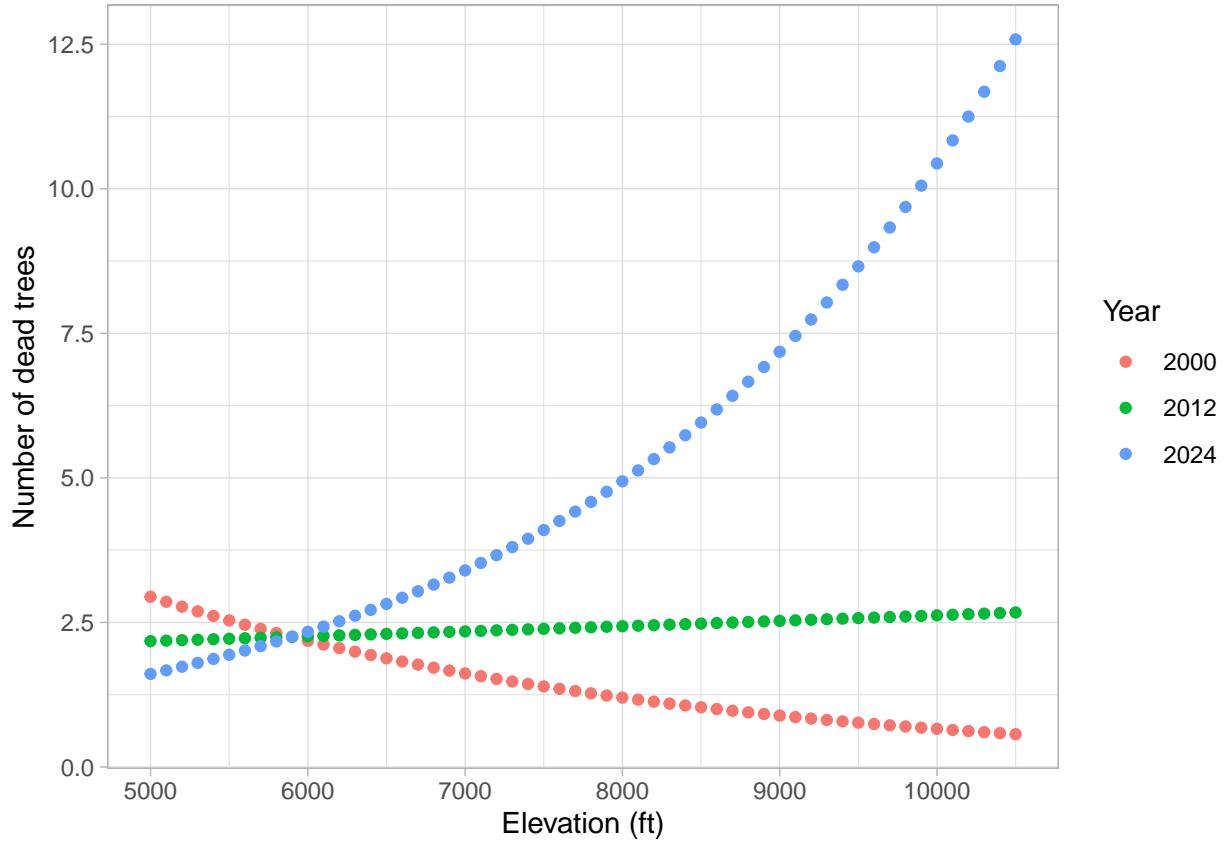
For a plot at 7,000 feet in the year 2024, the expected number of dead trees is approximately 3.

- (5) [5 pts] Create a figure like the one below that shows the predicted number of dead trees by plot elevation for the years 2000, 2012, and 2024. **PEC**. Explain in your own words how the effect of elevation on number of dead trees appears to have changed from 2000 to 2024.

```

1 elev <- seq(5000, 10500, 100)
2 yr2000 <- rep(2000, length(elev))
3 yr2012 <- rep(2012, length(elev))
4 yr2024 <- rep(2024, length(elev))
5
6 plot2000 <- data.frame(elev, yr2000);names(plot2000) <- c("ELEV", "INVYR")
7 plot2012 <- data.frame(elev, yr2012);names(plot2012) <- c("ELEV", "INVYR")
8 plot2024 <- data.frame(elev, yr2024);names(plot2024) <- c("ELEV", "INVYR")
9
10 pred2000 <- exp(predict(glm1, plot2000))
11 pred2012 <- exp(predict(glm1, plot2012))
12 pred2024 <- exp(predict(glm1, plot2024))
13
14 predictions <- data.frame(c(pred2000, pred2012, pred2024), rep(elev, 3),
15                             c(yr2000, yr2012, yr2024))
16 names(predictions) <- c("count", "elev", "year")
17
18 ggplot(predictions, aes(x = elev, y = count))+
19   geom_point(aes(color = factor(year)))+
20   theme_light()+
21   labs(x = "Elevation (ft)", y = "Number of dead trees", color = "Year")

```



It is possible that in 2024, higher-elevation forests were more likely to experience beetle outbreaks or severe wildfires that caused tree mortality. Higher elevation forests could have higher tree density due to moisture availability which can increase susceptibility to density-dependent drivers of mortality.

For all following spatial models, use projected spatial coordinates for plot location according to a UTM Zone 12 coordinate reference system.

```
1 plot_utm <- st_transform(plot_loc, crs = "epsg:32612")
```

- (6) [4 pts] Fit a spatial GLM to counts of dead trees with a Poisson response distribution and log link function and the same predictors as the non-spatial GLM. Assume the latent spatial random effect has a Gaussian-shaped covariance function. Report 95% CIs for the regression coefficients. What is the predicted number of dead trees at the plot with ID 82202 in the year 2024?

```
1 # fit glm:
2 glm_sp <- spglm(DEAD ~ INVYR + ELEV + INVYR:ELEV, data = plot_utm,
3                   family = poisson,
4                   spcov_type = "gaussian")
5 exp(confint(glm_sp))
```

```
##                      2.5 %         97.5 %
## (Intercept) 1.707517e+40 1.471908e+240
## INVYR        7.598576e-01  9.552268e-01
## ELEV         9.167884e-01  9.729920e-01
## INVYR:ELEV  1.000014e+00  1.000043e+00
```

```

1 library(sf)
2
3 plot_info <- plot_utm %>%
4   select(PLOT, ELEV, geometry) %>%
5   unique()
6
7 # data setup
8 pred_sp <- data.frame(2024, plot_info$ELEV[which(plot_info$PLOT == 82202)],
9                       plot_info$geometry[which(plot_info$PLOT == 82202)])
10 names(pred_sp) <- c("INVYR", "ELEV", "geometry")
11 pred_sp <- st_as_sf(pred_sp)
12
13 #prediction:
14 predict(glm_sp, pred_sp, type = "response")

```

```

##          1
## 7.568594

```

We predict that approximately 8 dead trees will occur in this plot in 2024.

- (7) [3 pts] Compute AIC for the models with and without a spatial random effect. In addition, use cross validation to estimate predictive bias mean-square prediction error for each model. Which model shows evidence of the best fit to the data based on predictive performance?

```

1 AIC(glm1, glm_sp)

##          df      AIC
## glm1     0 4213.763
## glm_sp   3 2960.742

1 sapply(list(glm1, glm_sp), function(x) loocv(x)[,1:3])

##      [,1]      [,2]
## bias -2.966063e-06 0.3598374
## MSPE  16.94964    9.483584
## RMSPE 4.116994    3.079543

```

While the model without spatial effects had a smaller bias, the model with spatial dependence had smaller prediction errors and a lower AIC (better fit to the data).

III. Modeling whether tree death occurred at all [11 pts]

Instead of modeling the counts of dead trees for each plot and inventory year, we could instead model whether *any* dead trees were recorded as a binary outcome (0 = no dead trees, 1 = at least one dead tree).

- (8) [3 pts] Fit a new non-spatial logistic regression model for the binary outcome of “at least one dead tree recorded” as a function of elevation, year, and their interaction. Report 95% CIs for each regression coefficient. Qualitatively speaking, how much evidence is there for a non-zero interaction effect?

```

1 plot_bin <- plot_utm %>%
2   mutate(DEAD = case_when(DEAD == 0 ~ 0,
3                           DEAD != 0 ~ 1))
4
5 binary_glm <- spglm(DEAD ~ INVYR*ELEV, data = plot_bin, family = binomial, spcov_type = "none")
6 exp(confint(binary_glm))/(1+exp(confint(binary_glm)))

```

```

##           2.5 %    97.5 %
## (Intercept) 1.729595e-94 1.0000000
## INVYR        4.140332e-01 0.5269327
## ELEV         4.733571e-01 0.5027686
## INVYR:ELEV  4.999986e-01 0.5000133

```

There seems to be a strong amount of evidence that there are interactions between elevation and year; CI for this effect does not include zero.

(9) [4 pts] Fit a spatial logistic regression model for the binary outcome of “at least one dead tree recorded” as a function of elevation and year and their interaction. Assume the latent spatial random effect has a Gaussian-shaped covariance function. Report 95% CIs for the regression coefficients. What is the predicted probability of recording at least one dead tree at the plot with ID 82202 in the year 2000? 2024?

```

1 binary_spglm <- spglm(DEAD ~ INVYR*ELEV, data = plot_bin, family = binomial, spcov_type = "gaussian")
2 exp(confint(binary_spglm))/(1+exp(confint(binary_spglm)))

```

```

##           2.5 %    97.5 %
## (Intercept) 2.481470e-238      NaN
## INVYR        3.839855e-01  0.5678345
## ELEV         4.604010e-01  0.5085094
## INVYR:ELEV  4.999957e-01  0.5000197

```

```

1 # predictions:
2
3 # set up new data:
4 pred_b1 <- data.frame(2000, plot_info$ELEV[which(plot_info$PLOT == 82202)],
5                         plot_info$geometry[which(plot_info$PLOT == 82202)])
6 names(pred_b1) <- c("INVYR", "ELEV", "geometry")
7 pred_b1 <- st_as_sf(pred_b1)
8
9 pred_b2 <- data.frame(2024, plot_info$ELEV[which(plot_info$PLOT == 82202)],
10                         plot_info$geometry[which(plot_info$PLOT == 82202)])
11 names(pred_b2) <- c("INVYR", "ELEV", "geometry")
12 pred_b2 <- st_as_sf(pred_b2)
13
14 # predictions:
15 predict(binary_spglm, pred_b1, type = "response") # 2000

```

```

##          1
## 0.8630693

```

```
1 predict(binary_spglm, pred_b2, type = "response") # 2024  
  
## 1  
## 0.9906045
```

There is an 86% chance of observing a dead tree in the plot in 2000, and a 99% chance in 2024.

- (10) [4 pts] In your opinion, do you think it makes more sense to model counts of dead trees or whether any trees died at all? Explain your reasoning. Can you think of any situation in which the approach you *did not* identify might make more sense?

In a classic ecologist/statistician way, I feel like there are equivalently interesting applications for both modeling approaches. I would be interested in modeling whether trees died at all to look at changes in the spatial extent of mortality risk. But, I would probably use the approach for modeling counts of dead trees to look at the severity of factors influencing tree mortality across space.