# Stat 534 Project:
# Extrapolation from Poisson Process Intensity Surface Models

Kenny Flagg

March 23, 2017

## 1 Introduction

One little-studied application of inhomogeneous point process intensity estimation is the use of an intensity surface estimated from events in a subregion to predict the intensity over the entire region of interest. This procedure would be relevant whenver it is known or suspected that some type of plant, animal, or other item occurs in the region, and the goal of the analysis is to map the trend in where these tend to be located rather than estimate parameters of some process at work across hypothetical replicates of similar regions. It may be prohibitively expensive or difficult to observe all events over the entire region, so a sample of subregions is taken. For example, conservationists want to study the spatial distribution of an endangered plant across a large region of thick jungle so that they can establish a preserve where the plant is protected. They cannot search the entire jungle, so they take a simple random sample of quadrats and record the locations of the plants in those quadrats. They will then want to fit a model describing the trends in intensity of these plants extrapolated over the entire jungle. In this setting, the objective is to map the realized spatial intensity of the plant over this jungle so that these plants can be protected, not to estimate parameters of the process that arranges plants of this species at jungles like this one.

Another situation where it is useful to extrapolate point pattern intensity outside of the observed region is in mapping subsurface geomagnetic anomalies, such as the munitions debris found at former military test ranges. This is frequently done in the early stages of an unexploded ordnance (UXO) remediation, where the intensity surface of inert munitions fragments is used to identify the locations of targets so that the search for UXO can be focused on the sections of the site most likely to contain it. The anomalies are only observed when detection equipment passes directly over them, but the project leaders are concerened with finding UXO and do not want to waste resources finding every inert fragment that could be found by metal detectors. The most common data collection method is to take a systematic sample of straight-line transects and observe anomalies in rectangular regions centered along the transects. Frequently, moving averages of the intensity are computed in circular windows and an intensity map is produced using ordinary kriging to predict the intensity in a grid of these windows. There are several problems with this approach: it assumes stationarity when the moving averages are believed to be non-stationary, it ignores the point process nature of the data, and it is very sensitive to the window size (Flagg 2016; Matzke et al. 2014). In this project, I evaluate the use of polynomial trend surface models for the log-intensity at a simulated UXO site.

## 2 Surface Fitting by Maximum Likelihood

It is theoretically possible to use maximum likelihood or maximum pseudo-likelihood methods to fit trend surface models or regression models (with covariates) for the intensity of an inhomogeneous point process, but these are not widely used because, in the most general cases, the problems are "notoriously intractable" (Diggle 2013). However, for spatial Poisson processes, trend surface models are tenable with the help of some numerical methods.

The log-likelihood of an unmarked Poisson process on a region $D$ with intensity $\lambda(\mathbf{s})$ is found by conditioning on the number of events in the following manner. The number of events in $D$ follows a Poisson distribution with mean $\mu = \int_D \lambda(\mathbf{s})d\mathbf{s}$. Given that $n$ events occured, their locations $\mathbf{s}_1, \ldots, \mathbf{s}_n$ are independent and identically distributed with density $\lambda(\mathbf{s})/\mu$. Then the log-likelihood of the intensity is

$$\begin{aligned}\ell(\lambda) &= \{-\mu + n\log(\mu) - \log(n!)\} + \sum_{i=1}^{n}\{\log(\lambda(\mathbf{s}_i)) - \log(\mu)\} \\ &= \sum_{i=1}^{n}\log(\lambda(\mathbf{s}_i)) - \int_D \lambda(\mathbf{s})d\mathbf{s} - \log(n!).\end{aligned}$$

A natural way to model the intensity is to define a log-linear model

$$\log(\lambda(\mathbf{s})) = \mathbf{x}(\mathbf{s})^T\boldsymbol{\beta}$$

where $\mathbf{x}(\mathbf{s})$ can, in principle, include functions of the spatial coordinates and also covariates. Unfortunately, the likelihood becomes

$$\sum_{i=1}^{n}\mathbf{x}(\mathbf{s})^T\boldsymbol{\beta} - \int_D \exp\left(\mathbf{x}(\mathbf{s})^T\boldsymbol{\beta}\right)d\mathbf{s} - \log(n!)$$

so $\mathbf{x}(\mathbf{s})$ must be known (or predicted) either across the entire region $D$ or at enough locations to approximate the integral numerically. Thus, covariates require some extra data collection or modeling effort to incorporate, but trend surfaces are feasible.

Berman and Turner (1992) proposed a practical method for fitting these models with generalized linear model (GLM) software using quadratic approximations to the likelihood, even generalizing to other link functions besides the log link. Their method is based on partitioning $D$ such that each subset in the partition contains at most one event, and then maximizing a weighted pseudolikelihood based upon a binomial or Poission distribution for the count of events in each subset. They comment that the covariance matrices produced by standard GLM software under this method are reasonable approximations to the correct covariance matrices that would be derived from the true likelihood as long as the quadrature approximation of the likelihood converges to the true likelihood as the resolution of the partition is increased. This was followed up by Baddeley et al. (2014), who developed a logistic regression-based approach that avoids the quadratic approximation, thereby reducing bias in the coefficient estimates. Both methods are available in the `spatstat` package (Baddeley, Rubak, and Turner 2015) for R (R Core Team 2017). The `spatstat` function `ppm` implements these methods using the base R `glm` function to maximize the likelihood; I will use the Berman and Turner method because it is the default in `ppm` and therefore better reflects the experience of a naïve practitioner who is not an expert on spatial point processes.

# 3 Examples

To illustrate model fitting and prediction, I present two examples before proceeding to the simulation study. The first is a toy example where I simulate from a Poisson process with a true log-linear intensity function and then estimate coefficients using the true model form. The second example introduces the my UXO site and provides a case study of fitting a polynomial trend surface to one realization of the the site.

# 4 True Log-Linear Intensity Surface

Previously this semester, we worked with an example of a Poisson process on the unit square $\{(x, y) : 0 < x < 1, 0 < y < 1\}$ with the log-linear intensity surface

$$\log(\lambda(x, y)) = 5x + 2y.$$

I will continue this example, fitting the model

$$\log(\lambda(x, y)) = \beta_0 + \beta_1 x + \beta_2 y$$

first using the process observed over the full region (the entire unit square), and then using the process observed in a subregion treating $\{(x, y) : 0.6 < x < 0.9, 0.6 < y < 0.9\}$ as unobserved. This unobserved section is a region of relatively high intensity, so if the estimation or prediction methods are sensitive to the observation window I expect omitting this section to have a large effect.

I will use both estimated models to predict on $\{(x, y) : -0.5 < x < 1.5, -0.5 < y < 1.5\}$. Figure 1 shows the realization of this process that I will use for both model fits.

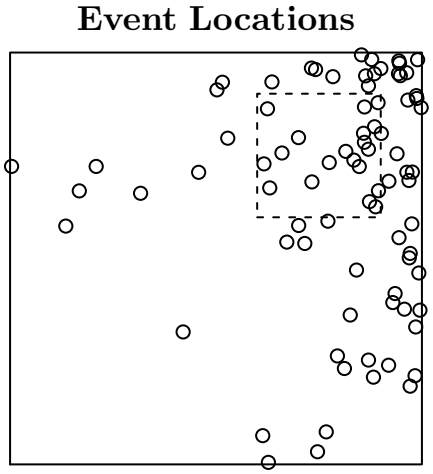



Figure 1: One realization of a Poisson process with log-linear intensity on the unit square. The dashed square shows the outline of the "unobserved" region; when fitting the model to the subregion, the points inside it will be discarded and it will not be considered in constructing the approximation to the likelihood.

| | Estimate | S.E. |
|---|---|---|
| $\widehat{\beta}_0^{(f)}$ | 0.16 | 0.54 |
| $\widehat{\beta}_1^{(f)}$ | 4.61 | 0.58 |
| $\widehat{\beta}_2^{(f)}$ | 2.02 | 0.42 |

Table 1: Estimated coefficients for the log-linear trend model fit using the full region.

| | Estimate | S.E. |
|---|---|---|
| $\widehat{\beta}_0^{(s)}$ | 0.20 | 0.56 |
| $\widehat{\beta}_1^{(s)}$ | 4.54 | 0.59 |
| $\widehat{\beta}_2^{(s)}$ | 2.00 | 0.44 |

Table 2: Estimated coefficients for the log-linear trend model fit using the subregion.

For clarity, I denote the estimated models as

$$\log\left(\widehat{\lambda}_f(x,y)\right) = \widehat{\beta}_0^{(f)} + \widehat{\beta}_1^{(f)}x + \widehat{\beta}_2^{(f)}y$$

for the model fit to the full region, and

$$\log\left(\widehat{\lambda}_s(x,y)\right) = \widehat{\beta}_0^{(s)} + \widehat{\beta}_1^{(s)}x + \widehat{\beta}_2^{(s)}y$$

for the model fit to the subregion. I use `ppm` to fit the models with its default likelihood approximation and isotropic edge correction. The coefficitents and standard errors are similar for both estimated models, and the true values $\beta_0 = 0$, $\beta_1 = 5$, and $\beta_2 = 2$ all fall within one standard error of the estimates (Tables 1 and 2).

After estimating the models, I use `spatstat`'s `predict.ppm` method to predict the intensity surface on a $128 \times 128$ lattice of locations in the region $\{(x,y) : -0.5 < x < 1.5, -0.5 < y < 1.5\}$ (Figure 2). As expected from the similarity of the coefficient estimates, the predicted log-intensity surfaces for both models show very similar linear trends (top row of the figure). At a glance the prediction standard errors for both models appear similar to the predicted intensities, as seen in the middle row of the figure (shown on a logarithmic scale for visibility). In a linear model setting we would expect the prediction standard error to increase with the distance from the observations, but no such trend is apparent here; even in the bottom left of the prediction region the standard errors decrease as the predicted intensity decreases. However, it is illuminating to plot the predictor's relative standard error (the ratio of the standard error to the predicted intensity, bottom row of the figure). This ratio is not constant, and in fact it is lowest where the highest intensity of events was observed.

In summary, the estimation procedure does an adequate job of estimating the parameters of the true model for these data, and the standard error of the predicted intensity appears to be more strongly related to the distance from the observed events than to the distance from the observed region.

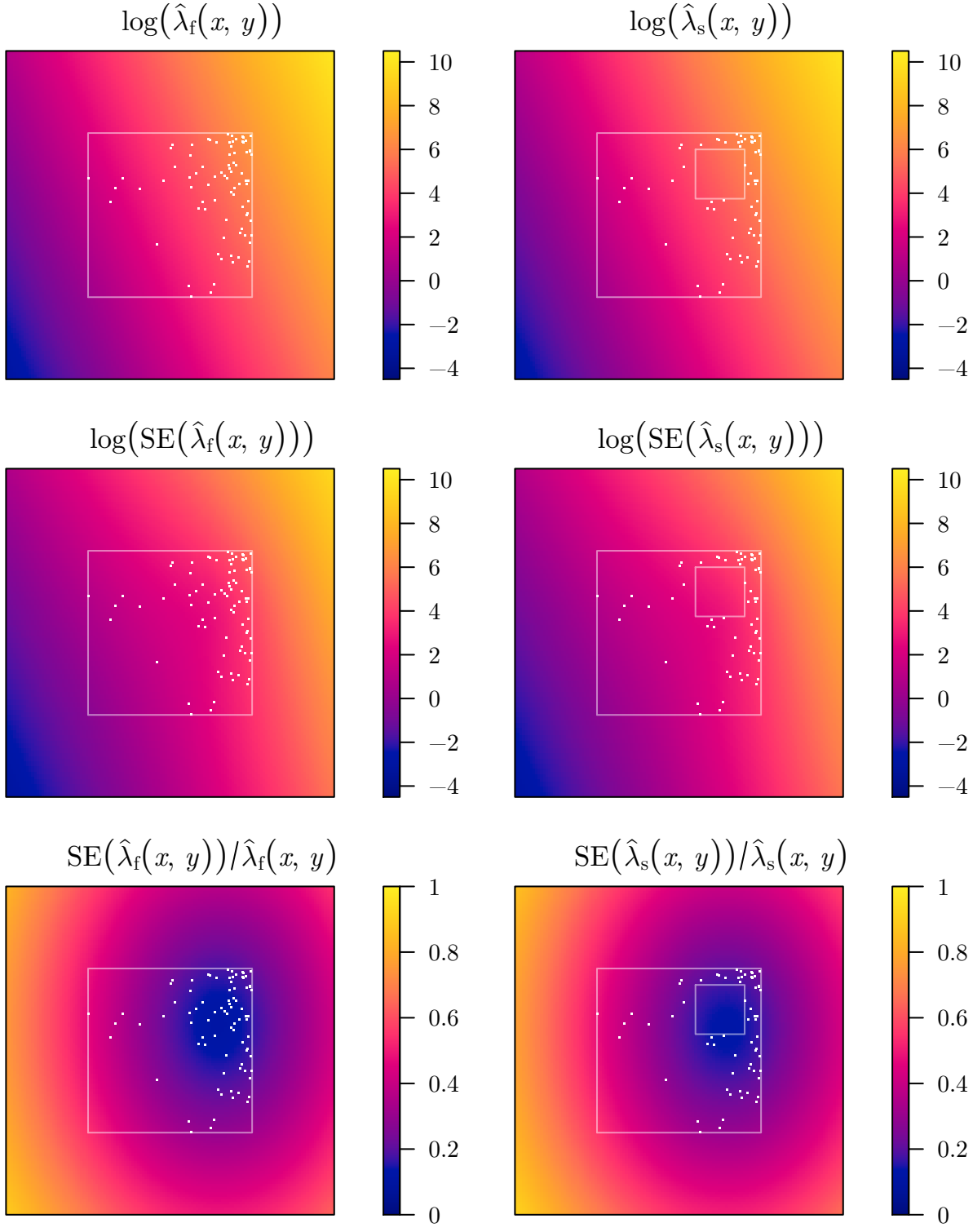



Figure 2: Estimated log-intensity surface, log-scale prediction standard errors, and relative standard error from models fit using all events in the full simulation region (left) and events in a subset of the simulation region (right). The regions and event locations are overlaid in white.

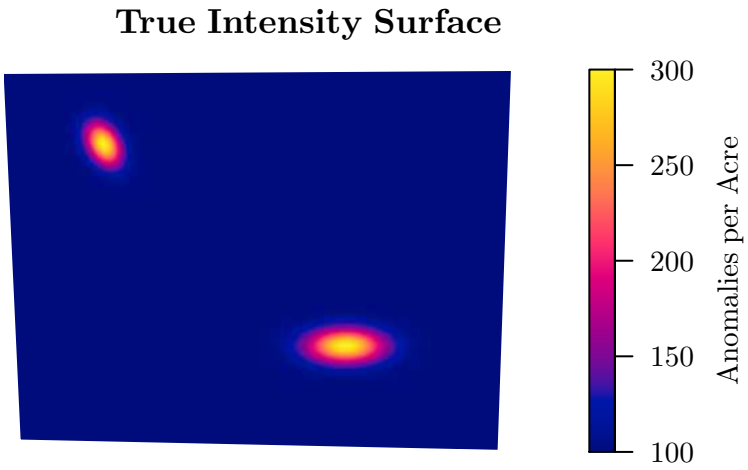



Figure 3: True intensity at the simulated UXO site.

## 5 Simulated UXO Site

In my writing project, I evaluated ordinary kriging of moving average intensity values on a simulated tank training range. I will use the same site for this project. The site is a 952.38 acre quadrilateral region with homogeneous Poisson process producing background anomalies with an intensity of 100 anomalies per acre, and two regions of concentrated munitions use generated as Poisson processes with bivariate Gaussian intensity. The total intesnsity at the center of each concentrated munitions use region is 300 anomalies per acre (Figure 3).

I simulate data collection along parallel straight-line transects spaced 396 feet apart using a detector with a footprint six feet wide (so there is an unobserved region 390 feet wide between each pair of adjacent transects). The location of the first transect is selected randomly, and the locations of all anomalies within three feet of a transect are recorded (Figure 4). In total, 14.7 acres are observed, covering 1.5% of the qegion of interest.

use `polynom()` not `poly()`!

`plot` doesn't work, and `predict` doesn't work on the original window

## 6 Discussion and Conclusion

smallish SEs outside observed region because of faith in the model form – need model checking

What geostatisticians and ordnance engineers will notice is that that SEs don't reflect added uncertainty about unobserved regions. Note kriging accounts for lack of model fit by autocorrelated errors – prediction SEs increase far from obs because of lack of info about local deviations from mean. For Poisson process, the events are independent given the intensity so local variation is not helpful for prediction. Thus we are heavily reliant on the correctness of the model form and the representativeness of the observed subregion, so model checking is crucial. If there is prior uncertainty about the form of the trend in a certain part of the site, that region must be observed!

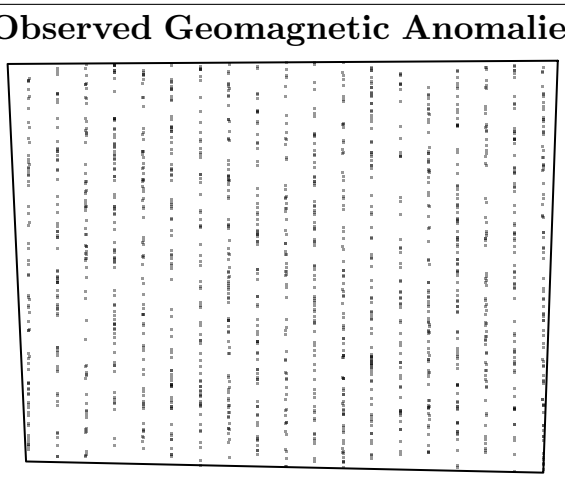



Figure 4: Observed geomagnetic anomalies from one realization of the UXO site. The transects are not shown on this plot because they would obscure the points.

how much of the region do you need to observe to trust your model?

# A R Code Appendix

# References

Baddeley, Adrian, Ege Rubak, and Rolf Turner (2015). *Spatial Point Patterns: Methodology and Applications with R*. London: Chapman and Hall/CRC Press.

Baddeley, Adrian et al. (2014). "Logistic regression for spatial Gibbs point processes". In: *Biometrika* 101.2, pp. 377–392.

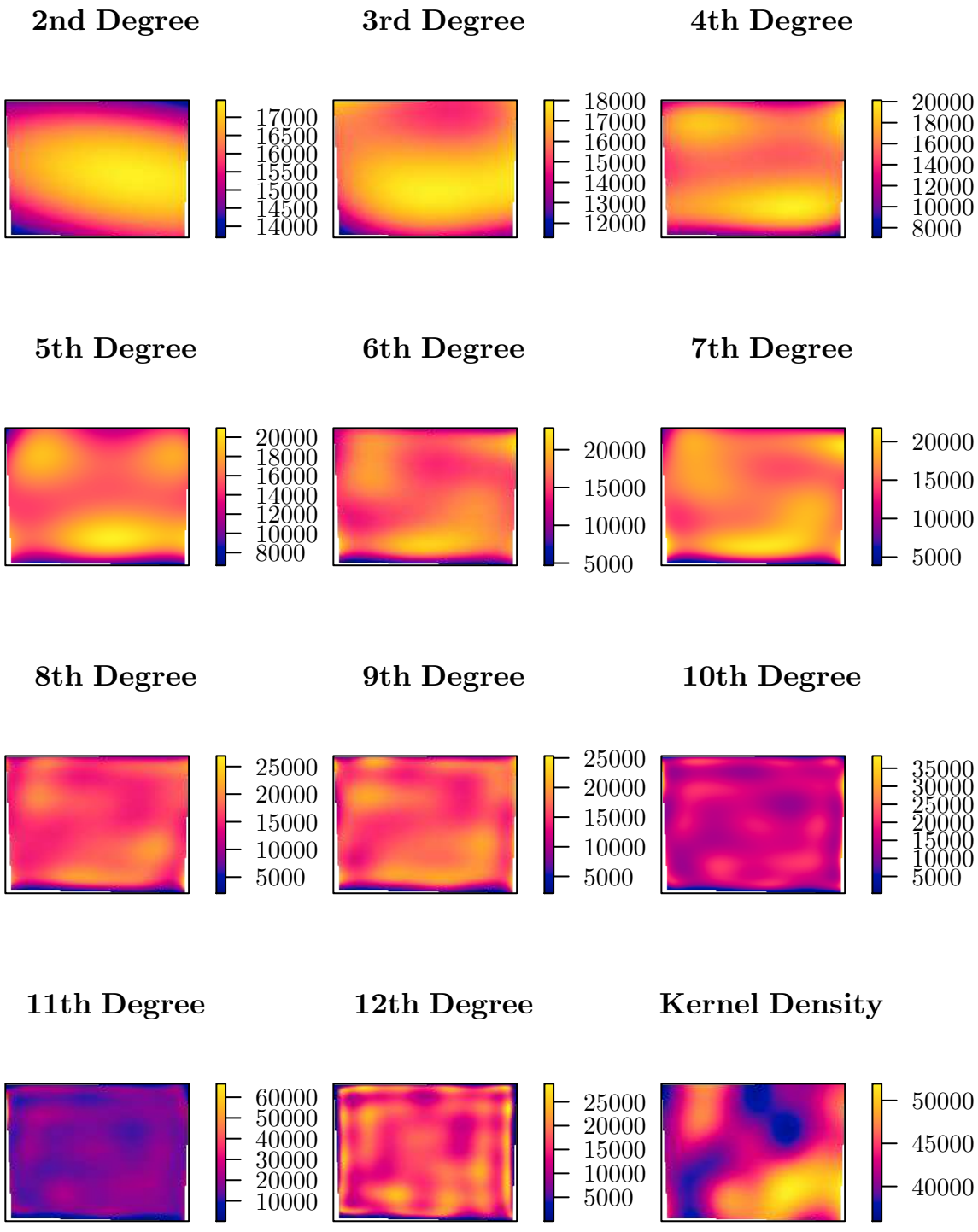
Berman, Mark and Rolf Turner (1992). "Approximating point process likelihoods with GLIM". In: *Applied Statistics*, pp. 31–38.

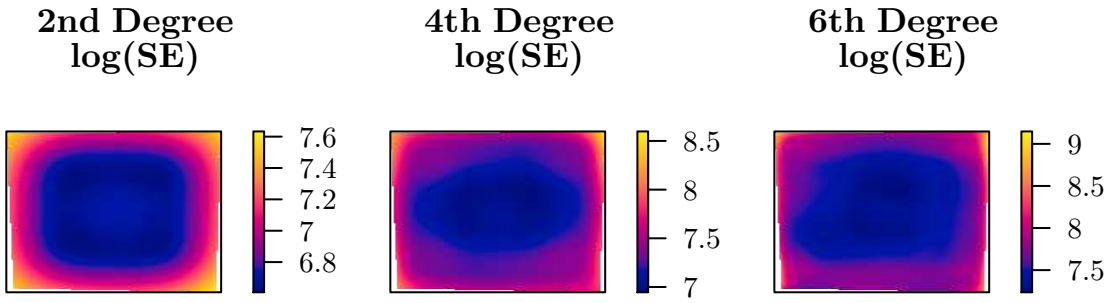
Diggle, Peter J. (2013). *Statistical Analysis of Spatial and Spatio-Temporal Point Patterns*. 3rd ed. CRC Press.

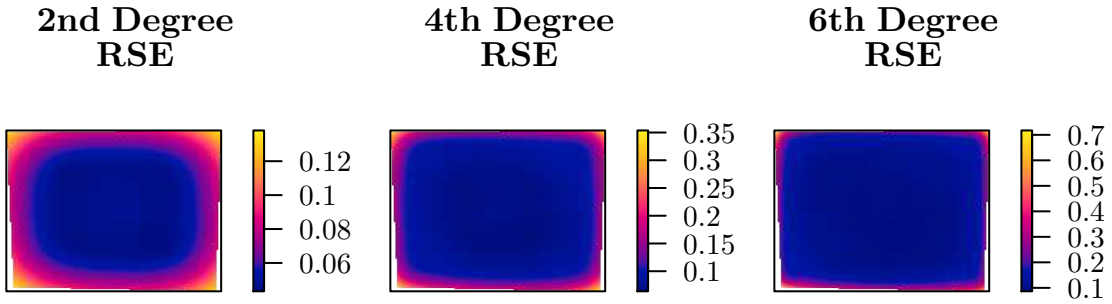
Flagg, KA (2016). "Visual Sample Plan and Unexploded Ordnance: What do we need to know to find UXO?" M.S. writing project. Montana State University, Bozeman. URL: `https://github.com/kflagg/vspuxo`.

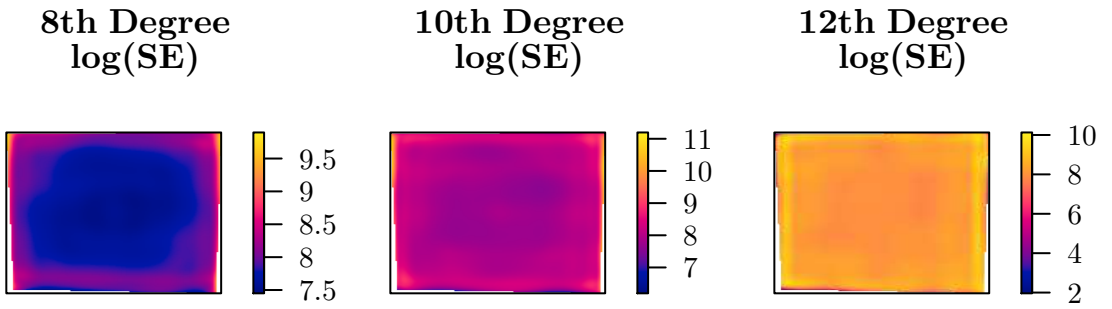
Matzke, Brett et al. (2014). *Visual Sample Plan Version 7.0 User's Guide*. Pacific Northwest National Laboratory. Richland, Washington. URL: `http://vsp.pnnl.gov/docs/PNNL-23211.pdf`.

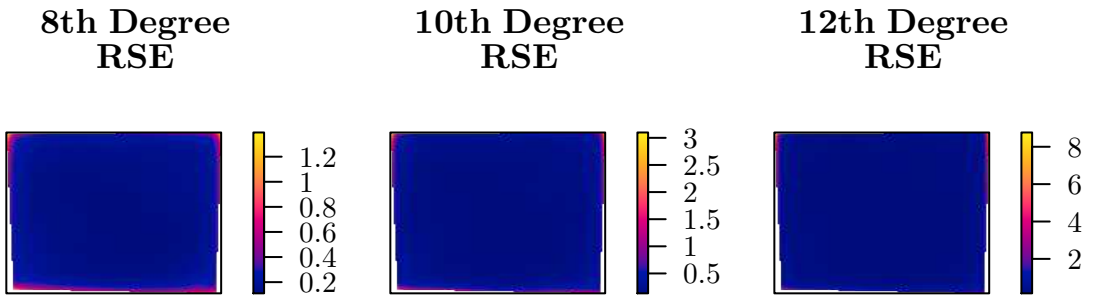
R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. URL: `https://www.R-project.org/`.


2nd Degree
17000
16500
16000
15500
15000
14500
14000
3rd Degree
18000
17000
16000
15000
14000
13000
12000
4th Degree
20000
18000
16000
14000
12000
10000
8000
5th Degree
20000
18000
16000
14000
12000
10000
8000
6th Degree
20000
15000
10000
5000
7th Degree
20000
15000
10000
5000
8th Degree
25000
20000
15000
10000
5000
9th Degree
25000
20000
15000
10000
5000
10th Degree
35000
30000
25000
20000
15000
10000
5000
11th Degree
60000
50000
40000
30000
20000
10000
12th Degree
25000
20000
15000
10000
5000
Kernel Density
50000
45000
40000


2nd Degree log(SE)
7.6
7.4
7.2
7
6.8
4th Degree log(SE)
8.5
8
7.5
7
6th Degree log(SE)
9
8.5
8
7.5


2nd Degree RSE
0.12
0.1
0.08
0.06
4th Degree RSE
0.35
0.3
0.25
0.2
0.15
0.1
6th Degree RSE
0.7
0.6
0.5
0.4
0.3
0.2
0.1


8th Degree log(SE)
9.5
9
8.5
8
7.5
10th Degree log(SE)
11
10
9
8
7
12th Degree log(SE)
10
8
6
4
2


8th Degree RSE
1.2
1
0.8
0.6
0.4
0.2
10th Degree RSE
3
2.5
2
1.5
1
0.5
12th Degree RSE
8
6
4
2