Hierarchical Modeling for Large non-Gaussian Datasets in R

Andrew O. Finley and Sudipto Banerjee

October 11, 2012

1 Data preparation and initial exploration

We make use of several libraries in the following example session, including:

- `library(spBayes)`
- `library(fields)`
- `library(geoR)`
- `library(gstat)`
- `library(lattice)`
- `library(MBA)`
- `library(maptools)`
- `library(rgdal)`
- `library(cluster)`
- `library(sp)`

The data used for this session is from the USDA Forest Service Forest Inventory and Analysis (FIA) national dataset. These are FIA's *fuzzed* data and can be download at `http://fia.fs.fed.us/tools-data/`. Our objective here is to determine if there is spatial patterns in tree species abundance across Minnesota. However, with $n = 5767$ forest inventory plots it is prohibitively expensive to fit a full geostatistical model. Rather, we resort to using the predictive process to capture dominate spatial structures across the domain.

In the code block below we plot the sample locations, Figure 1, and take a look at the interpolated surface of tree species count, Figure 2.

```
> MN.shp <- readShapePoly("MN-data/minnesota_aea.shp")
> MN.spp <- read.table("MN-data/MN_spp.txt", header = TRUE)
> n.spp <- MN.spp[, "n.spp"]
> coords <- MN.spp[, c("x", "y")]
> plot(coords, pch = 19, cex = 0.5, xlab = "Easting (m)",
+     ylab = "Northing (m)")
```

```
> res <- 100
> surf <- mba.surf(cbind(coords, n.spp), no.X = res,
+     no.Y = res, h = 7, extend = TRUE, sp = TRUE)$xyz.est
> surf <- surf[!is.na(overlay(surf, MN.shp)), ]
```

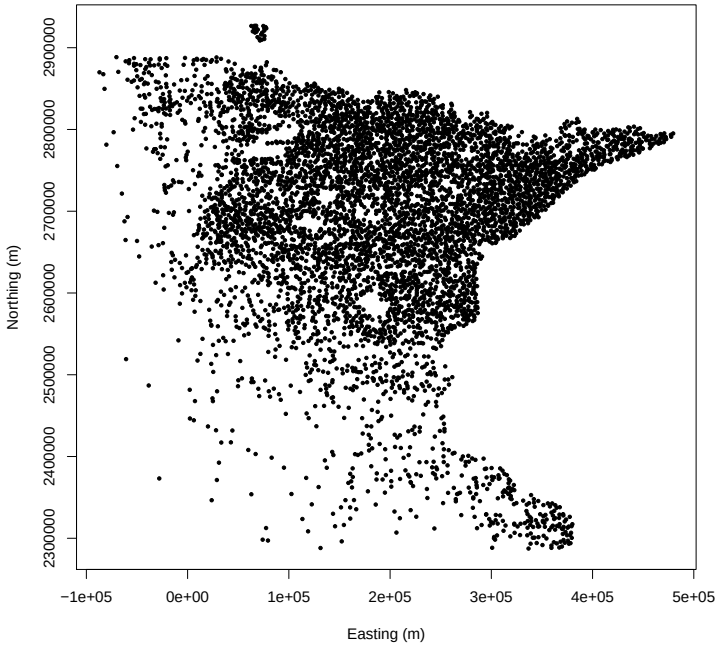

Figure 1: Forest inventory plot locations across MN.

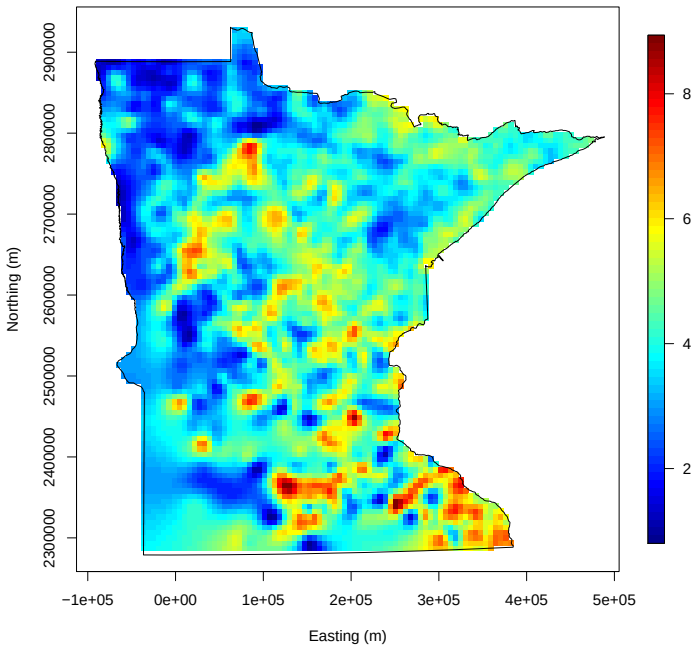

Figure 2: Interpolated surface of tree species counts.

```
> surf <- as.image.SpatialGridDataFrame(surf)
> image.plot(surf, xaxs = "r", yaxs = "r", xlab = "Easting (m)",
+     ylab = "Northing (m)")
> plot(MN.shp, add = TRUE)
```

We define the knot grid in the code block below and plot them in Figure 3

```
> x.range <- range(coords[, 1])
> y.range <- range(coords[, 2])
> knots <- expand.grid(x = seq(x.range[1], x.range[2],
+     length.out = 15), y = seq(y.range[1], y.range[2],
+     length.out = 15))
> coordinates(knots) <- c("x", "y")
> knots <- knots[!is.na(overlay(knots, MN.shp))]@coords
> plot(MN.shp, axes = TRUE, xlab = "Easting (m)", ylab = "Northing (m)")
> points(knots)
```

Next we call `spGLM` using the defined knot locations. Note, this is far too few MCMC samples, but gives us a feel for the computing efficiency gained by using the preditive process.

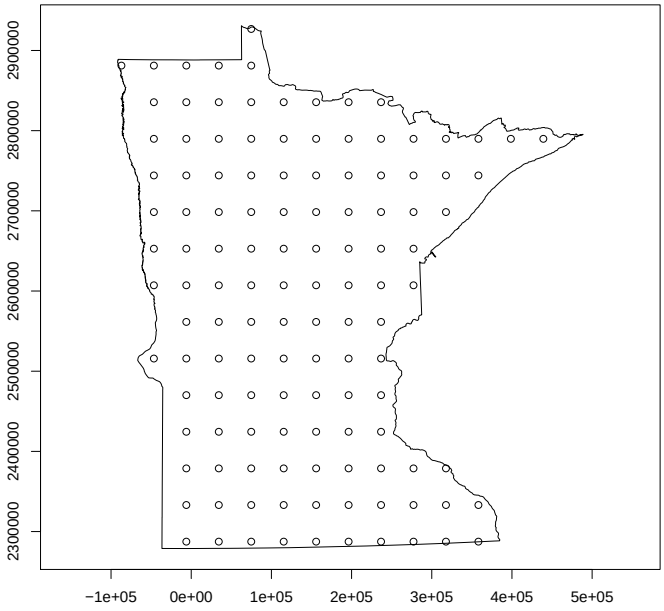

Figure 3: Predictive process knot locations.

```
> coords <- as.matrix(coords/1000)
> knots <- as.matrix(knots/1000)
> max.dist <- max(iDist(coords))
> max.dist

[1] 744.672

> beta.starting <- coefficients(glm(n.spp ~ 1, family = "poisson"))
> beta.tuning <- t(chol(vcov(glm(n.spp ~ 1, family = "poisson"))))
> n.samples <- 1000
> spp.spGLM <- spGLM(n.spp ~ 1, family = "poisson",
+     coords = coords, knots = knots, starting = list(beta = beta.starting,
+         phi = 3/100, sigma.sq = 0.2, w = 0), tuning = list(beta = beta.tuning,
+         phi = 0.1, sigma.sq = 0.005, w = 1e-04),
+     priors = list("beta.Flat", phi.Unif = c(3/400,
+         3/1), sigma.sq.IG = c(2, 0.2)), cov.model = "exponential",
+     n.samples = n.samples, sub.samples = c(750, n.samples,
+         5), verbose = TRUE, n.report = 100)

----------------------------------------
        General model description
----------------------------------------
Model fit with 5767 observations.

Number of covariates 1 (including intercept if specified).

Using the exponential spatial correlation model.

Using modified predictive process with 124 knots.

Number of MCMC samples 1000.

Priors and hyperpriors:
        beta flat.

        sigma.sq IG hyperpriors shape=2.00000 and scale=0.20000

        phi Unif hyperpriors a=0.00750 and b=3.00000

Metropolis tuning values:
        beta tuning:
        0.006

        sigma.sq tuning: 0.07071

        phi tuning: 0.31623
```

```
Metropolis starting values:
        beta starting:
                1.437

        sigma.sq starting: 0.20000

        phi starting: 0.03000

------------------------------------------------
                Sampling
------------------------------------------------
Sampled: 100 of 1000, 10.00%
Report interval Metrop. Acceptance rate: 47.00%
Overall Metrop. Acceptance rate: 47.00%
------------------------------------------------
Sampled: 200 of 1000, 20.00%
Report interval Metrop. Acceptance rate: 41.00%
Overall Metrop. Acceptance rate: 44.00%
------------------------------------------------
Sampled: 300 of 1000, 30.00%
Report interval Metrop. Acceptance rate: 43.00%
Overall Metrop. Acceptance rate: 43.67%
------------------------------------------------
Sampled: 400 of 1000, 40.00%
Report interval Metrop. Acceptance rate: 44.00%
Overall Metrop. Acceptance rate: 43.75%
------------------------------------------------
Sampled: 500 of 1000, 50.00%
Report interval Metrop. Acceptance rate: 38.00%
Overall Metrop. Acceptance rate: 42.60%
------------------------------------------------
Sampled: 600 of 1000, 60.00%
Report interval Metrop. Acceptance rate: 36.00%
Overall Metrop. Acceptance rate: 41.50%
------------------------------------------------
Sampled: 700 of 1000, 70.00%
Report interval Metrop. Acceptance rate: 41.00%
Overall Metrop. Acceptance rate: 41.43%
------------------------------------------------
Sampled: 800 of 1000, 80.00%
Report interval Metrop. Acceptance rate: 32.00%
Overall Metrop. Acceptance rate: 40.25%
------------------------------------------------
Sampled: 900 of 1000, 90.00%
Report interval Metrop. Acceptance rate: 49.00%
Overall Metrop. Acceptance rate: 41.22%
```

```
-------------------------------------------------
Sampled: 1000 of 1000, 100.00%
-------------------------------------------------

> spp.spGLM$p.samples[, "phi"] <- 3/spp.spGLM$p.samples[,
+     "phi"]
> summary(mcmc(spp.spGLM$p.samples))$quantiles[, c(1,
+     3, 5)]

                    2.5%          50%        97.5%
(Intercept)   1.38215234   1.39685473   1.41159315
sigma.sq      0.04372357   0.05789481   0.07469157
phi         209.59781766 278.42186055 340.10919786

> beta.hat <- mean(spp.spGLM$p.samples[, 1])
> w.hat <- rowMeans(spp.spGLM$sp.effects)
> n.spp.fitted <- exp(beta.hat + w.hat)
```

Finally, let's take a look at the fitted values, Figure 4.

```
> coords <- MN.spp[, c("x", "y")]
> par(mfrow = c(2, 1))
> surf <- mba.surf(cbind(coords, n.spp), no.X = res,
+     no.Y = res, extend = TRUE, sp = TRUE)$xyz.est
> surf <- surf[!is.na(overlay(surf, MN.shp)), ]
> surf <- as.image.SpatialGridDataFrame(surf)
> image.plot(surf, xaxs = "r", yaxs = "r", xlab = "Easting (m)",
+     ylab = "Northing (m)", main = "Interpolated species count")
> surf <- mba.surf(cbind(coords, n.spp.fitted), no.X = res,
+     no.Y = res, extend = TRUE, sp = TRUE)$xyz.est
> surf <- surf[!is.na(overlay(surf, MN.shp)), ]
> surf <- as.image.SpatialGridDataFrame(surf)
> image.plot(surf, xaxs = "r", yaxs = "r", xlab = "Easting (m)",
+     ylab = "Northing (m)", main = "Fitted species counts")
```

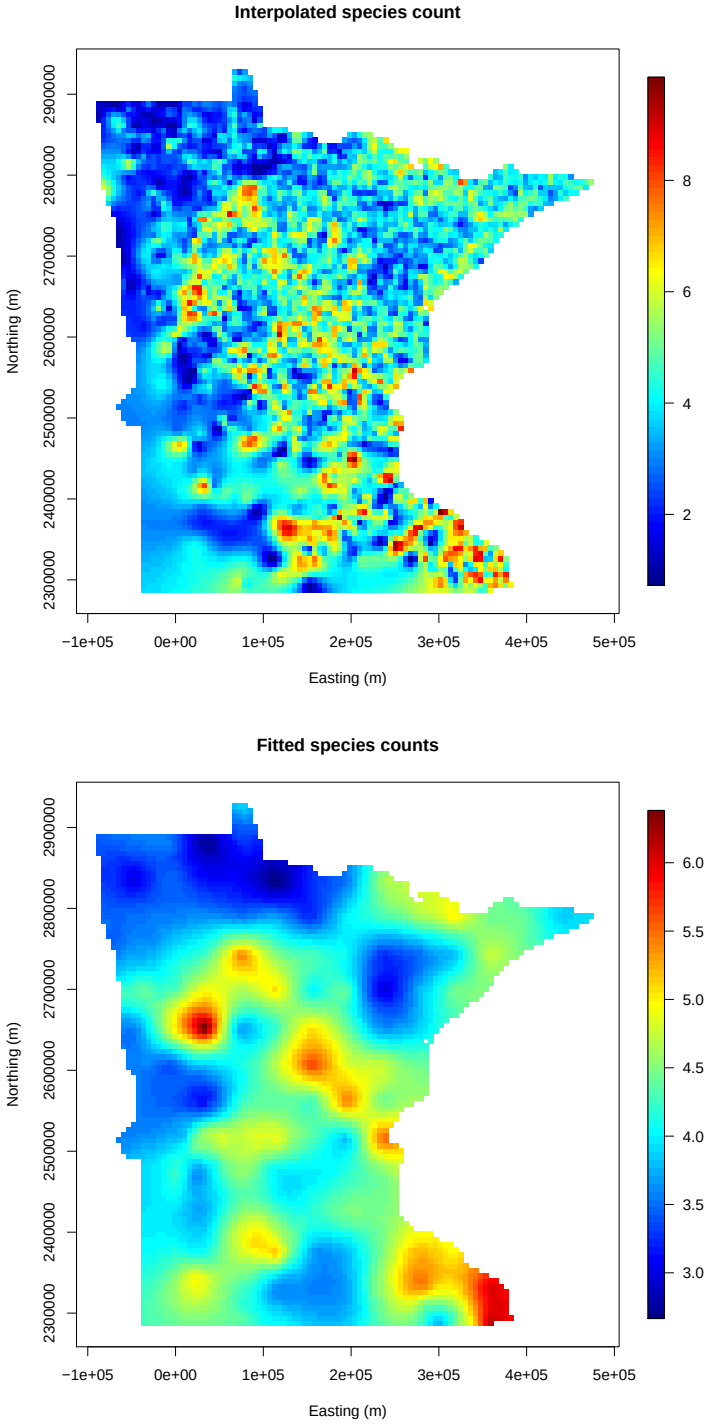

Figure 4: Observed versus fitted tree species counts.

2 References

Banerjee, S., Carlin, B.P., and Gelfand, A.E. (2004). *Hierarchical Modeling and Analysis for Spatial Data*, Boca Raton, FL: Chapman and Hall/CRC Press.

Bivand, R.B., Pebesma, E.J., and Gómez-Rubio, V. (2008). *Applied Spatial Data Analysis with R*, UseR! Series, Springer.

Diggle, P.J. and Riberio, P.J. (2007). *Model-based Geostatistics*, Series in Statistics, Springer.

Gelman, A., Carlin, J.B., Stern, H.S., and Rubin, D.B. (2004). *Bayesian Data Analysis*. Second Edition. Boca Raton, FL: Chapman and Hall/CRC Press.

Spiegelhalter, D.J., Best, N.G., Carlin, B.P., and van der Linde, A. (2002). Bayesian measures of model complexity and fit (with discussion). *Journal of the Royal Statistical Society, Series B* **64**, 583–639.