Baseball Research Log

Chris Comiskey

Winter 2017

Week of December 5, 2016

- An option:
  - Started with $\boldsymbol{\beta} \sim \mathrm{N}(\mathbf{0}, 5\mathbf{I}_6)$ non-informative prior
  - ...Define $\boldsymbol{\theta} = \boldsymbol{R}\boldsymbol{\beta}$
  - ...giving $\mathrm{Var}(\boldsymbol{\theta}) = \boldsymbol{R}\mathrm{Var}(\boldsymbol{\beta})\boldsymbol{R}'$
  - Then use prior distribution $\boldsymbol{\theta} \sim \mathrm{N}(\boldsymbol{R}\hat{\boldsymbol{\beta}}, 5 * \boldsymbol{R}\boldsymbol{R}')$ (...but diagonal only)
- `Hitter6.stan`. Used $\boldsymbol{R}$ and polar $\hat{\boldsymbol{\beta}}_{\mathrm{GLM}}$ to estimate $\mathrm{E}[\boldsymbol{\theta}]$ and $\mathrm{Var}[\boldsymbol{\theta}]$, as in previous bullet point.

```
data {
  int<lower=0> N;                 // Number of observations
  int<lower=0> p;                 // number of predictors
.
.
.
      Z ~ normal(0, 1);  // Each element is N(0,1)
      Z_mod = Sigma * Z; // (Cov matrix Cholesky)*MVN(0,1)
      hit ~ bernoulli_logit(beta0 + Q*theta + Z_mod);
}
```

- Reading Finley et al. [2007]
- Also looking at Andy's notes from NCAR.
- Reading slides titled "functional-casestudy.pdf"
- Idea: could define β in `Hitter6.stan` as $\hat{\beta}_{\mathrm{GLM}}$, to get idea of length-scale ballpark.
- Alarming question: is Z a parameter?
- Ran `Hitter5.stan` overnight, for N = 2000. Only made it through 350 samples. Prohibitive.
- Gotta try sampling loop. For i in 1:100, sample 200, fit with `Hitter5.stan`, then compile all those draws. Would that work/help?
- Reading Neal et al. [2011] to learn about Hamiltonian MCMC (HMC) algorithm, which Stan uses. Hamiltonian Dynamics; Mr. Meyers would be proud.
- Wikipedia: "In mathematics, the **gradient** is a generalization of the usual concept of derivative to functions of several variables."

- What if I used my variable resolution grid to estimate variance parameters?
  - Using the usual box centers reduction would be shoddy because many boxes would have very little data, and thus misrepresent Cov(Box1, Box2).
  - For example, $\hat{p}_{\text{Box1}} = 1$ for one out of one, against $\hat{p}_{\text{Box2}} = 0.12$ for 12 out of 100, misrepresents Cov(Box1, Box2).
  - Using my variable resolution grid would offer advantages, by eliminating $\hat{p}_{\text{Box1}} = 1$ scenarios.

Meeting

- Alix was worried, because I wanted to meet *this week*, that I had found a job and was going to leave early! She said she has had other students do the same, and it is very hard to finish in that fashion.
- Alix read my paper, made some comments, said it was good. She thinks at the end there should be a section of sorts that helps the (baseball) reader understand and interpret the new map, rather than than just understanding the algorithm. What do the box sizes tell us? What does the particular spatial variation in box size tell us?
- Alix is thinking space might be a dual stopping criteria. For example, can hitters even distinguish pitches in two adjacent very small boxes? They probably look like the identical pitch, so why assign separate batting averages.
- "Banish from your mind" the question of whether or not this is PhD level research. She said there is **no way anyone below the PhD level could make their way through** (i) Finley's papers, (ii) the QR decomposition of $\boldsymbol{X\beta}$ (linear model theory) in my logistic regression model (GLM) and the resultant change in the prior distributions (Bayesian-LMT), or (iii) MCMC with Hamiltonian dynamics, etc (just the stuff we talked about today). And, she said, it's very cool we're applying it to something totally different (other than forestry, etc), baseball, where it totally applies. Bottom line: **Alix is sure.** She's my primary adviser. "God," as Bruce put it. Her opinion carries all the weight. The jury starts and stops with her. She believes it, I get a PhD.
- Can always contact Andy and/or Malcom is needbe
- Need to try leaving $\theta \sim N(\mathbf{0}, 5 * \boldsymbol{R} * \boldsymbol{R^T})$. In other words, shoul be fine to leave the mean uninformative if adjust variance.
- Okay that θs correlated according to $Var(\theta) = 5 * \boldsymbol{R} * \boldsymbol{R^T}$ because coefficient estimates always are, even in simple linear regression.
- My idea of using variable resolution grid to change grid-reduced estimates of correlation parameters is worth coming back to. She wasn't sure if the base method existed (or not) to improvise upon.

Carrying on...

- As mentioned above. Correct adjusted prior variances on $\hat{\theta}$, with mean zero. `Hitter5WOSC`

```
> print(fit_hitterWOSC, pars=c("beta0","beta"), digits = 3)
Inference for Stan model: Hitter5WOSC.
3 chains, each with iter=500; warmup=250; thin=1;
post-warmup draws per chain=250, total post-warmup draws=750.

          mean se_mean    sd   2.5%    25%    50%    75%  97.5% n_eff  Rhat
beta0   -4.068   0.055 0.682 -5.475 -4.516 -4.066 -3.627 -2.800   155 1.015
beta[1]  1.168   0.038 0.487  0.219  0.847  1.150  1.473  2.228   168 1.014
```

```
beta[2] -1.929   0.127 1.874 -5.803 -3.199 -1.812 -0.535  1.314   217 1.008
beta[3] -0.316   0.006 0.086 -0.494 -0.369 -0.315 -0.261 -0.149   236 1.009
beta[4] -3.978   0.064 1.093 -6.225 -4.742 -3.910 -3.187 -2.059   290 1.006
beta[5] -1.691   0.059 0.887 -3.357 -2.289 -1.722 -1.065  0.106   229 1.008
beta[6] -0.471   0.013 0.212 -0.861 -0.618 -0.483 -0.321 -0.056   282 1.006
```

As they should be! They match the plane Jane GLM fit without spatial correlation random effect.

- `Hitter5WOSC.stan`

```
data {
  int<lower=0> N;                 // Number of observations
  int<lower=0> p;                 // number of predictors
  matrix[N,p] Q;                  // QR decomp - Q
  matrix[p,p] R;                  // QR decomp - R
  int<lower=0, upper=1> hit[N];   // 0/1 outcomes; array of integers
  vector[p] theta_SDs;            // theta prior SDs
}
transformed data{
  matrix[p,p] R_inv;
  R_inv = inverse(R);
}
parameters {
  real beta0;                       // intercept
  vector[p] theta;
}
transformed parameters {
      vector[p] beta;
      beta = R_inv*theta;
}
model {
      beta0 ~ normal(0,5);
      theta[1] ~ normal(0, theta_SDs[1]);
      theta[2] ~ normal(0, theta_SDs[2]);
      theta[3] ~ normal(0, theta_SDs[3]);
      theta[4] ~ normal(0, theta_SDs[4]);
      theta[5] ~ normal(0, theta_SDs[5]);
      theta[6] ~ normal(0, theta_SDs[6]);
      hit ~ bernoulli_logit(beta0 + Q*theta);
}
```

- **Ask Alix:** Try sampling loop... For i in 1:100, sample 200, fit with `Hitter5.stan`, then compile all those draws. Would that work/help?

Week of January 9, 2017

- Where was I?
  1. STAN needs to go faster; enter Andrew Finley
  2. Andrew Finley, reduce dimensionality of correlation
  3. Paper; Alix edits/feedback to look at

4. Expand picture effort to include 3D representation of CIs
5. Hamilton equations

- Confirmed: Using 9177 obs, QR decomposition, and QR inflated variances, but leaving out spatial correlation, all incorporated into `Hitter5WOSC.stan` yields same estimates as `glm()`. So stan code is working, and QR decomposition is working. As of now, running the same procedure WITH spatial correlation is prohibitively slow.

MCMC Using Hamiltonian Dynamics [Neal et al., 2011]

- Paraphrased: MCMC originated with (Metropolis et al. 1953) to simulate states for molecules... another approach (Alder and Wainwright, 1959) formalized Newtons laws of motion as Hamiltonian dynamics.
- Duane et al. 1987 paper united approaches as Hybrid MC, or Hamiltonian MC $\rightarrow$ HMC for lattice field theory quantum dynamic simulations.
- Statistical use began in 1996 with neural network models, other applications followed.
- Differential equations and "leapfrog" scheme, elementary mathematics

Steps.

1. "Define a Hamiltonian function in terms of the probability distribution we wish to sample from."
2. Interested in "position" variables. Introduce auxiliary "momentum" variables (typically independent Gaussian distributions)
3. The HMC method alternates:
   (a) Simple updates for these momentum variables
   (b) Metropolis updates - computing Hamiltonian dynamic trajectory for new proposal state, leapfrog implementation method, distant proposals w/ high acceptance probability; bypass slow random walk proposal distribution

- Physical interpretation - frictionless puck on varying height surface. Puck at **position** q with **momentum** p and mass m, **potential energy** $U(q)$, **kinetic energy** $K(p) = |p|^2/(2m)$.
- Wikipedia: "In statistics, especially in Bayesian statistics, the **kernel** of a probability density function (pdf) or probability mass function (pmf) is the form of the pdf or pmf in which any factors that are not functions of any of the variables in the domain are omitted."

0.0.1 Equations

- q = position, U(q) = potential energy
  - **q** = variables of interest
  - let $U(q) = -\log f_q(q)$
- p = momentum, K(p) = kinetic energy
  - Introduce artificially, same dimension as **q**

- Partial derivatives determine change over time. For $i = 1, \ldots, d$:

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}$$

$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}$$

- $H(q,p) = U(q) + K(p) = -\log f_q(q) + p^T \boldsymbol{M}^{-1} p/2$
  - Total = Potential + Kinetic
  - $U(q) = -\log f_q(q)$
  - $K(p) = -\log\ f_p(p) = p^T \boldsymbol{M}^{-1} p/2$, (using p $\sim$ N(0,$\mathbf{M}$))
- Rewrite:

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i} = [\boldsymbol{M}^{-1} p]_i$$

$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i} = -\frac{\partial U}{\partial q_i}$$

- Solution: q(t) = ?, p(t) = ?

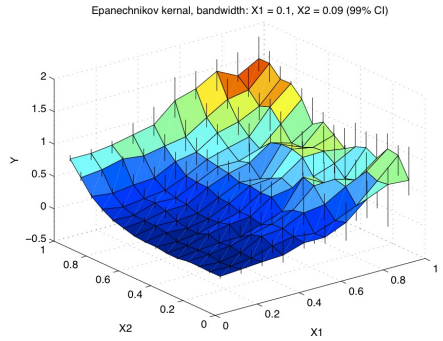
Week of January 17, 2017

Hamiltonian Mechanics/Dynamics

- Leapfrog method = for calculating new position (q) and momentum (p) through tiny time steps
  - for **discretizing Hamilton equations**
  - akin to Taylor Series appoximations
  - Postion (q) (or momentum (p)) at t_0 plus time step times rate of change of position (q) (momentum (p)) variable at t_0
  - Leapfrom Method does half step for momentum (p), full step for postion (q), other half step for momentum (p). Damn good.
- **Statistical ensemble** - probability distribution for the state of a system (set of all possible copies)
- Short version: randomly sample from K(p) (kinetic, momentum), calculate U(k) (potential, position*) — that's your Metropolis proposal.
- **Solve** a differential equation: If $\frac{dq}{dt} = p$, and $\frac{dp}{dt} = q$, then q(t) = ?, and p(t) = ?

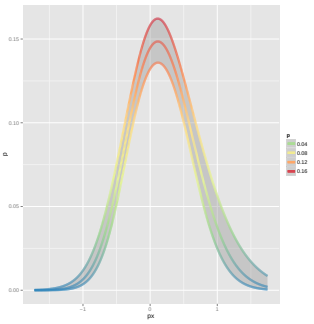
Pictures

Consider this presentation of CIs in 3D.



Note the line segments through the surface.
Brainstorming.

- Could do a *shiny* **app (!!)** that gives a $1-\alpha$ slider to move ~~three-dimensionally~~ through the CI, showing the lower $(\alpha/2)$ and upper bound $(1-\alpha/2)$ maps on the left (lower) and right (upper) of the point estimate map. The left and right would converge to middle map (point estimate) for 0% CI.
- Could even... **have option for cutoff(!!)** as part of app! ...and px, pz, CI percentile
- Could have a profile plot attached, with px or pz on x-axis, $\hat{p}$ on y-axis—with colors.



- Confusion: In my heat maps, color = $\hat{p}$. In my mental images of CIs $\hat{p}$ the 3D qualities of height and width represent $\hat{p}$ too.
- **shiny** app is the key.

Shiny

- See `Shiny.R`
- Tutorials: https://shiny.rstudio.com/tutorial/

Lesson 1 - Recap

To create your own Shiny app:

1. Make a directory named for your app.
2. Save your apps `server.R` and `ui.R` script inside that directory.
3. Launch the app with runApp (button) or RStudios keyboard shortcuts (Command+Shift+Enter).
4. Exit the Shiny app by clicking escape.

Lesson 2 - Build a User Interface

- create a user-interface with `fluidPage()`, `titlePanel()` and `sidebarLayout()`

```
# ui.R

shinyUI(fluidPage(

  titlePanel("title panel"),

  sidebarLayout(
    sidebarPanel("sidebar panel"),
    mainPanel("main panel")
  )

))
```

- create an HTML element with one of Shinys tag functions; i.e. `h2()`, `br()`, etc.
- set HTML tag attributes in the arguments of each tag function; i.e. `h2("header", align = "center")`
- add an element (i.e. `img(...)` or `h2(...)`) to your web page by passing it to `titlePanel`, `sidebarPanel` or `mainPanel`.
- add multiple elements to each panel by separating them with a comma
- add images by placing your image in a folder labeled `www` within your Shiny app directory and then calling the `img` function (`img(src = "bigorb.png", height = 72, width = 72)`)

Lesson 3 - Add Control Widgets

- Widget - A web element users can interact with.
- Widgets have help pages; `?actionButton`

Meeting

- Dissertation
  - Chapter 1: I made the Alix feedback changes (Alix happy), I will add Shiny CI presentation stuff to round it out. Doesn't need to be more technical.
  - Chapter 2 (and this term): `spbayes()`, fit model with random effect.
  - Chapter 3: Does random effect improve* model? *How do we measure *improve.
- Need to brush up on "credible interval," the Bayesian paradigm's answer to confidence intervals.
- I explained my plan for Shiny app approach to CI. They liked the idea. Alix especially.
  - Gameplan is to show CI for variable resolution heat map
  - And for continuous model **without** random effect, but to only briefly introduce the model at this point, and allude to thorough introduction to come in Chapter 2.
- Charlotte went to RStudio conference.

  - Mango had a table there, promoting their training mostly (she thinks).
  - She said they have fully bought into the Hadley Wickham universe of packages.
  - Mango now teaches R by starting with `dplyr`, data frames, etc.
  - (This is great news for me!)
  - She said Mango was one of the companies on her radar if academia didn't work out!
- Alix asked: "Is HMC what `spbayes()` uses?"
  - I don't know, but I don't need to own Hamiltonian dynamics if `spbayes()` doesn't use it. Good point Alix.

Week of January 23, 2017

Lesson 4 - Reactive *Output*

- Use an `*Output` (e.g. `textOutput(...)`) function in the `ui.R` script to place reactive objects in your Shiny app

```
# ui.R

shinyUI(fluidPage(
  titlePanel("censusVis"),

  sidebarLayout(
    sidebarPanel(
      helpText("Create demographic maps with
               information from the 2010 US Census."),

      selectInput("var",
                  label = "Choose a variable to display",
                  choices = c("Percent White", "Percent Black",
                              "Percent Hispanic", "Percent Asian"),
                  selected = "Percent White"),

      sliderInput("range",
                  label = "Range of interest:",
                  min = 0, max = 100, value = c(0, 100))
      ),

    mainPanel(
      textOutput("text1"),  # **Notice "output$text1"
      textOutput("text2")
    )
  )
))
```

- Use a `render*` function in the `server.R` script to tell Shiny how to build your objects

```
# server.R

shinyServer(function(input, output) {
```

```
    output$text1 <- renderText({            # **Notice "output$text1"
      paste("You have selected", input$var) # **Notice "input$var"
      })

    # ** Notice "output$text2" contains "input$range[1]"
    output$text2 <- renderText({
      paste("Your range is", input$range[1], "to", input$range[2])
    })

  })
```

- Surround R expressions by braces, , in each `render*` function
- Save your `render*` expressions in the output list, with one entry for each reactive object in your app.
- Create reactivity by including an input value in a `render*` expression

* Workflow seems to me: `ui.R` takes **input** $\longrightarrow$ `server.R` receives **input** and creates **output** $\longrightarrow$ `ui.R` displays **output**.

* Reactivity: connect `input` *values* to `output` **objects**.

Lesson 5 - Use R Scripts and Data

```
# server.R

  # A place to put code ***

shinyServer(
  function(input, output){

      # Another place to put code ***

      output$map <- renderPlot({

          # A third place to put code ***

      )}

    }
  )
```

- The `server.R` script is run once, when you launch your app
- The unnamed function inside `shinyServer` is run once each time a user visits your app
- The R expressions inside `render*` functions are run many times. Shiny runs them once each time a user changes a widget.

How can you use this information?

- Source scripts, load libraries, and read data sets at the beginning of `server.R` outside of the `shinyServer` function. Shiny will only run this code once, which is all you need to set your server up to run the R expressions contained in `shinyServer`.

- Define user specific objects inside `shinyServer`s unnamed function, but outside of any `render*` calls. These would be objects that you think each user will need their own personal copy of. For example, an object that records the users session information. This code will be run once per user.
- Only place code that Shiny must rerun to build an object inside of a `render*` function. Shiny will rerun all of the code in a `render*` chunk each time a user changes a widget mentioned in the chunk. This can be quite often.
- You should generally avoid placing code inside a render function that does not need to be there. The code will slow down the entire app.

Recap

- You can create more complicated Shiny apps by loading R Scripts, packages, and data sets.
- The directory that `server.R` appears in will become the working directory of the Shiny app
- Shiny will run code placed at the start of `server.R`, before `shinyServer`, only once during the life of the app.
- Shiny will run code placed inside `shinyServer` multiple times, which can slow the app down.
- You also learned that `switch` is a useful companion to multiple choice Shiny widgets. Use switch to change the values of a widget into R expressions.

Lesson 6 - Reactive *Expressions*

Reactive expressions save their calculated value, and only recalculate when a widget input changes.

```
# server.R

library(quantmod)
source("helpers.R")

shinyServer(function(input, output) {

  # Use reactive({.}) expression to create object
  dataInput <- reactive({
    getSymbols(input$symb, src = "yahoo",
      from = input$dates[1],
      to = input$dates[2],
      auto.assign = FALSE)
  })

  # Use reactive object in render*({.}) statment
  output$plot <- renderPlot({
    chartSeries(dataInput(), theme = chartTheme("white"),
      type = "line", log.scale = input$log, TA = NULL)
  })
})
```

"When you click the Plot y axis on the log scale widget button, `input$log` will change and `renderPlot` will re-execute. Now

1. `renderPlot` will call `dataInput()`

2. `dataInput` will check that the "dates" and "symb" widgets have not changed
3. `dataInput` will return its saved data set of stock prices *without re-fetching data from Yahoo*
4. `renderPlot` will re-draw the chart with the correct axis.

Shiny will automatically re-build an object if

- an input value in the objects's `render*` function changes, or
- a reactive expression in the objects's `render*` function becomes obsolete

Reactive expressions save their results, and will only re-calculate if their input has changed.

Lesson 7

Share your apps...

- **Shinyapps.io**

  The easiest way to turn your Shiny app into a web page is to use shinyapps.io (http://my.shinyapps.io/), RStudio's hosting service for Shiny apps.

  shinyapps.io lets you upload your app straight from your R session to a server hosted by RStudio. You have complete control over your app including server administration tools. You can find out more about shinyapps.io by visiting shinyapps.io (http://my.shinyapps.io/).

Finley, Quest for Gaussian Predictive Process Models

- This [Finley et al., 2009a] looks possibly helpful.
- Bingo. [Finley et al., 2009b].
  - Abstract: "...spatially-varying multinomial logistic regression models to predict forest type groups... spatially-varying impact of predictor variables... onerous computational burdens and we discuss dimension reducing spatial processes..."
- Maybe [Finley et al., 2011]
- Looks cool. [Guhaniyogi et al., 2011]
  - Large point referenced datasets occur frequently in the environmental and natural sciences. Use of Bayesian hierarchical spatial models for analyzing these datasets is undermined by onerous computational burdens associated with parameter estimation. Low-rank spatial process models attempt to resolve this problem by projecting spatial effects to a lower-dimensional subspace. This subspace is determined by a judicious choice of knots or locations that are fixed a priori. One such representation yields a class of predictive process models (e.g., Banerjee et al., 2008) for spatial and spatial-temporal data. Our contribution here expands upon predictive process models with fixed knots to models that accommodate stochastic modeling of the knots. We view the knots as emerging from a point pattern and investigate how such adaptive specifications can yield more flexible hierarchical frameworks that lead to automated knot selection and substantial computational benefits.
- [Finley et al., 2012]
- [Eidsvik et al., 2012]

  - The challenges of estimating hierarchical spatial models to large datasets are addressed. With the increasing availability of geocoded scientific data, hierarchical models involving spatial processes have become a popular method for carrying out spatial inference. Such models are customarily estimated using Markov chain Monte Carlo algorithms that, while immensely flexible, can become prohibitively expensive. In particular, fitting hierarchical spatial models often involves expensive decompositions of dense matrices whose computational complexity increases in cubic order with the number of spatial locations. Such matrix computations are required in each iteration of the Markov chain Monte Carlo algorithm, rendering them infeasible for large spatial datasets. The computational challenges in analyzing large spatial datasets are considered by merging two recent developments. First, the predictive process model is used as a reduced-rank spatial process, to diminish the dimensionality of the model. Then a computational framework is developed for estimating predictive process models using the integrated nested Laplace approximation. The settings where the first stage likelihood is Gaussian or non-Gaussian are discussed. Issues such as predictions and model comparisons are also discussed. Results are presented for synthetic data and several environmental datasets.

- **spBayes for large univariate and multivariate point-referenced spatio-temporal data models** [Finley et al., 2013]
  - In this paper we detail the reformulation and rewrite of core functions in the spBayes R package. These efforts have focused on improving computational efficiency, flexibility, and usability for point-referenced data models. Attention is given to algorithm and computing developments that result in improved sampler convergence rate and efficiency by reducing parameter space; decreased sampler run-time by avoiding expensive matrix computations, and; increased scalability to large datasets by implementing a class of predictive process models that attempt to overcome computational hurdles by representing spatial processes in terms of lower-dimensional realizations. Beyond these general computational improvements for existing model functions, we detail new functions for modeling data indexed in both space and time. These new functions implement a class of dynamic spatio-temporal models for settings where space is viewed as continuous and time is taken as discrete.

Gaussian Predictive Process Models for Large Spatial Data Sets [Banerjee et al., 2008]

- "**Spatial predictive process** - project the original (spatial) process onto a subspace that is generated by realizations of the original process at a specified set of locations."
- "We regard the predictive process as a competing model specification with computational advantages, but induced by an underlying full rank process."
- Paraphrased: "Our method similar to reduced rank kriging method proposed by Cressie and Johannesson (2008)... but theirs is ineffective for hierarchical models, that have random effects at second stage of specification, and no data to provide an empirical covariance function." (pg 829)
- Knot selection: "We need a criterion to decide between a regular grid and placing more knots where we have sampled more. One approach would be a so-called **space filling knot selection** following the design ideas of Nychka and Saltzman (1998). Such designs are based on geometric criteria, measures of how well a given set of points covers the study region, independent of the assumed covariance function. Stevens and Olsen (2004) showed that spatial balance of design locations is more efficient than simple random sampling."
- "A direct assessment of knot performance is comparison of the covariance function of the parent process with that of the predictive process, $\tilde{C}(\boldsymbol{s}, \boldsymbol{s}'; \boldsymbol{\theta}) = \boldsymbol{c}^T(\boldsymbol{s}; \boldsymbol{\theta}) \cdot \boldsymbol{C}^{*-1}(\boldsymbol{\theta}) \cdot \boldsymbol{c}(\boldsymbol{s}'; \boldsymbol{\theta})$, where predictive process $\tilde{w}(\boldsymbol{s}) \sim \text{GP}\{0, \tilde{C}(\cdot)\}$

1. $Y(\boldsymbol{s}) = \boldsymbol{x}^T(\boldsymbol{s})\boldsymbol{\beta} + w(\boldsymbol{s}) + \epsilon(\boldsymbol{s})$
   - $\epsilon(\boldsymbol{s}) \sim^{iid} N(0, \tau^2)$
2. $w(\boldsymbol{s}) \sim GP\{0, C(\boldsymbol{s}, \boldsymbol{s}')\}$
   - Gaussian process, covariance function $C(\boldsymbol{s}, \boldsymbol{s}')$
3. $\boldsymbol{Y} \sim N(\boldsymbol{X}\boldsymbol{\beta}, \Sigma_{\boldsymbol{Y}})$
   - $\Sigma_{\boldsymbol{Y}} = C(\boldsymbol{\theta}) + \tau^2 \boldsymbol{I}_N$
   - $\boldsymbol{X} = [\boldsymbol{x}^T(\boldsymbol{s}_i)]_{i=1}^n$
   - $C(\boldsymbol{\theta}) = [C(\boldsymbol{s}_i, \boldsymbol{s}_j; \boldsymbol{\theta})]_{i,j=1}^n$
4. $\boldsymbol{S}^* = \{\boldsymbol{s}_1^*, \ldots, \boldsymbol{s}_m^*\}$ (knots)
5. $\boldsymbol{w}^* = [w(\boldsymbol{s}_i^*)]_{i=1}^m$ (random effect at knots)
6. $\boldsymbol{w}^* = [w(\boldsymbol{s}_i^*)]_{i=1}^m \sim \text{MVN}\{\boldsymbol{0}, \boldsymbol{C}^*(\boldsymbol{\theta})\}$
   - $\boldsymbol{C}^*(\boldsymbol{\theta}) = \left[C(\boldsymbol{s}_i^*, \boldsymbol{s}_j^*)\right]_{i,j=1}^m$ (Covariance of knots with themselves; $m \times m$)
7. $\tilde{w}(\boldsymbol{s}_0) = E[w(\boldsymbol{s}_0)|\boldsymbol{w}^*] = \boldsymbol{c}^T(\boldsymbol{s}_0; \boldsymbol{\theta}) \cdot \boldsymbol{C}^{*-1}(\boldsymbol{\theta}) \cdot \boldsymbol{w}^*$
   - $\boldsymbol{c}(\boldsymbol{s}_0; \boldsymbol{\theta}) = \left[C(\boldsymbol{s}_0, \boldsymbol{s}_j^*; \boldsymbol{\theta})\right]_{j=1}^m$ (covariance of $\boldsymbol{s}_0$ with knots; $m \times 1$ vector)
8. $\tilde{w}(\boldsymbol{s}) \sim \text{GP}\{0, \tilde{C}(\cdot)\}$
   - $\tilde{w}(\boldsymbol{s})$ is predictive process
   - $\tilde{C}(\boldsymbol{s}, \boldsymbol{s}'; \boldsymbol{\theta}) = \boldsymbol{c}^T(\boldsymbol{s}; \boldsymbol{\theta}) \cdot \boldsymbol{C}^{*-1}(\boldsymbol{\theta}) \cdot \boldsymbol{c}(\boldsymbol{s}'; \boldsymbol{\theta})$ (**game changer**)
   - $\boldsymbol{c}(\boldsymbol{s}; \boldsymbol{\theta}) = \left[C(\boldsymbol{s}, \boldsymbol{s}_j^*)\right]_{j=1}^m$ (covariance of $\boldsymbol{s}$ with knots)
9. $Y(\boldsymbol{s}) = \boldsymbol{x}^T(\boldsymbol{s})\boldsymbol{\beta} + \tilde{w}(\boldsymbol{s}) + \epsilon(\boldsymbol{s})$
   - **Predictive process model**
   - $\tilde{w}(\boldsymbol{s})$ is spatially varying linear transformation of $\boldsymbol{w}$.

spBayes for Large Point-Referenced Spatio-Temporal Data Models

- See `spBayes.R`
- `spGLM()` - "fits univariate Bayesian generalized linear spatial regression models. Given a set of knots, spGLM will also fit a predictive process model" (from help page)
- ```
  spGLM(formula, family="binomial", weights,
        data = parent.frame(), coords, knots,
        starting, tuning, priors, cov.model,
        amcmc, n.samples, verbose=TRUE,
        n.report=100, ...)
  ```
- I don't understand: `tuning` and `amcmc` (adaptive MCMC)
- `tuning` - a list with each tag corresponding to a parameter name. **The value portion of each tag defines the variance of the Metropolis sampler Normal proposal distribution.**
  ```

- Exponential Covariance: From `spBayes` spatial binomial example. (from `spBayes()` cran documentation)
  - `R <- sigma.sq*exp(-phi*as.matrix(dist(coords)))`
  - $\rightarrow Cov(s_1, s_2) = \sigma^2 exp[-\phi \cdot d(s_1, s_2)]$
- `priors` argument - a list with each tag corresponding to a parameter name. Valid tags are `sigma.sq.ig`, `phi.unif`, `nu.unif`, `beta.norm`, and `beta.flat`.
  - Variance parameter `simga.sq` is assumed to follow an inverse-Gamma distribution,
  - whereas the spatial decay `phi` and smoothness `nu` parameters are assumed to follow Uniform distributions.
  - The hyperparameters of the inverse-Gamma are passed as a vector of length two, with the first and second elements corresponding to the shape and scale, respectively.
  - The hyperparameters of the Uniform are also passed as a vector of length two with the first and second elements corresponding to the lower and upper support, respectively.
  - If the regression coefficients are each assumed to follow a Normal distribution, i.e., `beta.norm`, then mean and variance hyperparameters are passed as the first and second list elements, respectively.
  - If beta is assumed flat then no arguments are passed. The default is a flat prior.

Fitting predictive process model with spBayes spGLM() - To do's

- Confirm it is running the model I think it is (Use Stan to confirm?)
- Need to run diagnostics after (working on it)
- Understand adaptive MCMC (punt)
- Understand tuning (variance of proposal distribution)
- Choose knots [Royle and Nychka, 1998] (can wait)
- Posterior credible intervals (??) (Gelman text)

Charlotte Meeting

- AMCMC - later
- Choose knots - later
- **Understand tuning - now**
- Diagnostics - after
- Run back to stan carrying predictive process models? Not just yet.

* **Note: grab Gelman's Bayesian book from office, read up on Metropolis-Hastings algorithm**

* Metropolis Note: when drawing from the proposal distribution, $Q(\cdot|\theta_t)$, if Q is the Normal, then $Q(\cdot|\theta_t)$ means $N(\theta_t, \cdot)$... as I suspected—but is sloppy notation.

Week of January 30, 2017

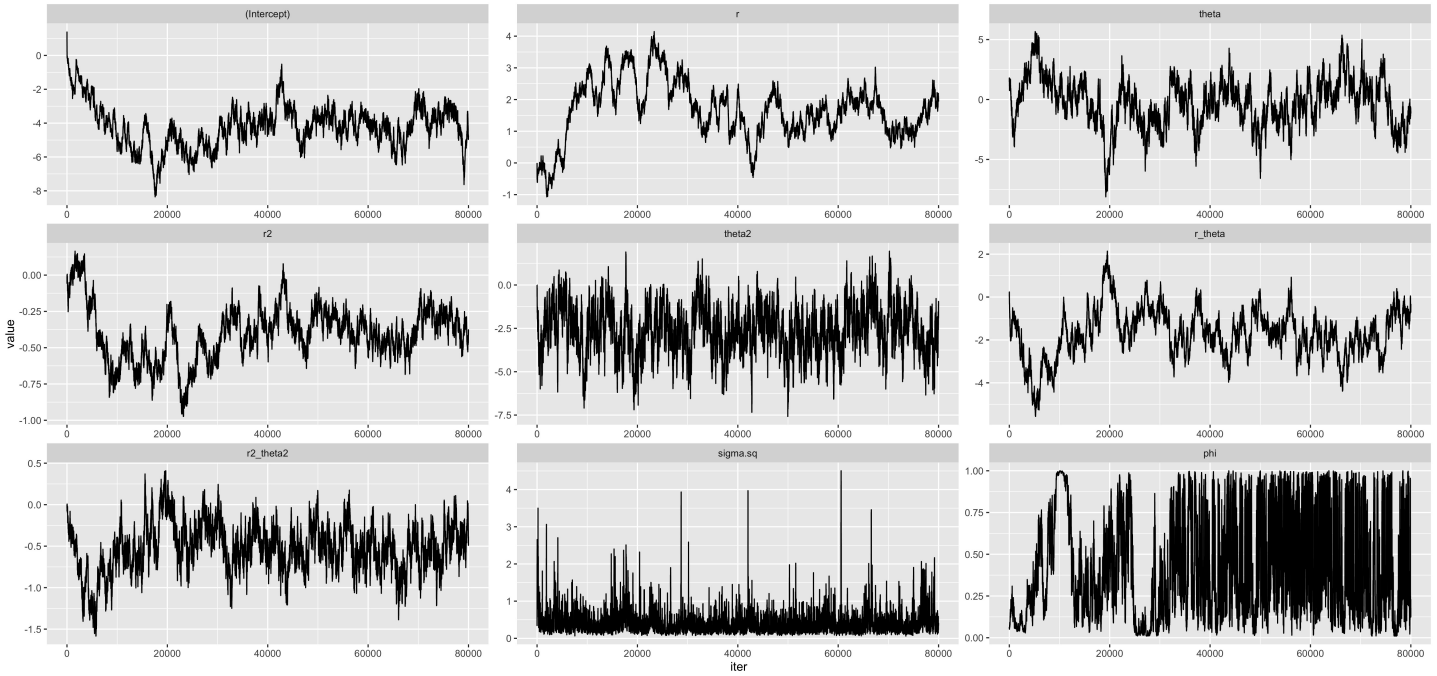
spBayes

- Current problem: the M-H MCMC acceptance rates are way too low.
- Problem question: Is `spGLM()` using proposal distributions with different support than parameters. e.g. Normal for σ^2?
- Right now trying to better understand M-H algorithm, using **BanerjeeLecture.pdf**, among other resources. Non-acceptance means r is very small, because perhaps the proposal variance is too big, and the jumps away are drastically reducing the likelihood.
- Trying **adaptive MCMC again, and it works**. It's slower, and, at n = 100 pitches, not converging. At n = 200 pitches, it seeeeems *closer* to converging (non-technical assessment!). Hijacked adaptive parameters:

```
parameter acceptance tuning
beta[0] 44.0 0.56789
beta[1] 40.0 0.19771
beta[2] 38.0 1.12920
beta[3] 40.0 0.05303
beta[4] 60.0 0.80597
beta[5] 64.0 0.26464
beta[6] 82.0 0.04910
sigma.sq 52.0 0.25585
phi     70.0 0.13728
```

- I think the thing to do now is add some knots at my variable resolution grid box centers, and see how it does!
- Variable Res. heat map knots
  - Using 97 knots resulting from an $n_b < 200$ cutoff, and n = 300 observations, `spGLM()` took about 3 mins. The trace plots did not suggest convergence.
  - Same, but n = 500, about 4 mins. Still does not look convergent. (10,000 iterations)
  - n = 1000, 6.7 mins. Convergence *maybe* a smidge better. But time is good!! (10,000 iterations)
  - n=1000; 30,000 iterations; looks better. Not exactly convergent, but closer.
  - 7 mins, n = 1000, knots = 49, 30K samples
  - X mins, n = 9172, knots = 49, "80,000 samples completed" but the dreaded rainbow pinwheel never went away when computer tried to resuscitate in morning. We'll never know...
  - 54 mins, n = 3000, kn = 49, 80K samples

```
                Mean     SD  Naive SE Time-series SE
(Intercept) -4.2154 1.2638 0.0044682       0.240307
r            1.6878 0.9192 0.0032499       0.270467
theta       -0.2776 2.0000 0.0070711       0.275294
r2          -0.4029 0.1878 0.0006639       0.037039
theta2      -2.5969 1.3842 0.0048940       0.097303
r_theta     -1.7129 1.1326 0.0040042       0.202156
r2_theta2   -0.5211 0.2922 0.0010331       0.030658
sigma.sq     0.3920 0.2736 0.0009674       0.009469
phi          0.4095 0.2872 0.0010156       0.018573
```



- Need to think. Digest. Good progress.
- Recall: Stan guys suggested identifiability and convergence issues can result from... something. Should look back at what they said.
- Stan implementation online uses version from "Improved..." follow-up paper Finley wrote.

Shiny it Up!

- Two heat maps to Shiny:
  1. Empirical Var Res; Normal approx. to Binomial: $\hat{p} \pm z_{1-\alpha/2} * \sqrt{\hat{p}(1-\hat{p})/n}$
  2. Polar model (w/o random effect); each point has its own SE; MLE estimates are Normally distributed (gotta double check in GLM text); use `predict()` function; use `Field_of_Dreams3.R` as a reference; SE of the fit is calculated, so only new $z_{1-\alpha/2}$ will need to be plugged into fit $\pm z_{1-\alpha/2}$*se.fit each time
- Not gonna be too bad.
- Confusing the heck out of me. Review.
  - α = P(No coverage)
  - CI% = 1 - α
  - $P(Z > z_\alpha) = \alpha$

    ```
    > qnorm(0.05, lower.tail = FALSE)
    [1] 1.644854
    ```

    So $P(Z > 1.645) = 0.05$. Think of Normal curve.
  - For 95% CI, we say $\alpha = 0.05$.

    * This necessitates $z_{1-\alpha/2}$ for CI.
    * $\hat{p} \pm z_{1-\alpha/2} * \sqrt{\hat{p}(1-\hat{p})/n}$

- I got... my Shiny app... TO WORK!!!

Meeting

Items to report

- Mango interview, R Coding Test
  - Alix: progress updates! Tell them if anything happens.
  - Definitely give them a heads up if they are going to get a call (as a reference).
  - Alix said: "You're one of the most curious people I've ever met. It's a great part of who you are. Always reading all those books, walking around with... [imitates fiddlink] in your hand." She said I could/should have added that to my well-rounded answer.
  - Charlotte/Alix said the same job here (statistical consultant) would bring in $120,000 - $140,000
- Heat Map CI Shiny app
  - Add simple, point and click CI??? (Charlotte's challenge)
- spBayes works-ish, trace plots
  - Trace plots indicate non-convergence, but this should not discourage, but instead encourage more iterations. More more more iterations.
  - Also keep in mind we're reducing information in some way, using knots only.
  - Both thought 97 was a lot.
  - Alix very excited it's working. "Andy is going to be so excited; we're using his package on baseball data!"

Evaluate Inverse you say?? Yes.

- $\text{logit}\{\text{EY(s)}\} = \boldsymbol{X}(s)\boldsymbol{\beta} + Z(s)$, with $Z(s) \sim MVN\{\boldsymbol{0}, \Sigma_s\}$
- $f(\boldsymbol{\beta}, \phi, \sigma^2, \boldsymbol{Z}|\boldsymbol{Y}) \propto f(\boldsymbol{Y}|\boldsymbol{\beta}, \phi, \sigma^2, \boldsymbol{Z}) f(\boldsymbol{\beta}) f(\boldsymbol{Z}|\phi, \sigma^2) f(\phi) f(\sigma^2)$
- M-H proposal, iteration i: $Z_{10,i}$

$$r = \frac{f(Z_{10,i}|\boldsymbol{Z}_{1:9,i}, \boldsymbol{Z}_{11:n,i-1}, \boldsymbol{\beta}_{i-1}, \phi_i, \sigma_i^2)}{f(Z_{10,i-1}|\boldsymbol{Z}_{1:9,i}, \boldsymbol{Z}_{11:n,i-1}, \boldsymbol{\beta}_{i-1}, \phi_i, \sigma_i^2)}$$

- Note: $f(z_1, z_2, z_3|\boldsymbol{Y}) = f(z_1|z_2, z_3, \boldsymbol{Y}) f(z_2, z_3|\boldsymbol{Y})$. So...

$$r \propto \frac{f(\boldsymbol{Y}|\boldsymbol{\theta}_i) f(\boldsymbol{\beta}) f(Z_{10,i}|\boldsymbol{Z}_{1:9,i}, \boldsymbol{Z}_{11:n,i-1}, \phi_i, \sigma_i^2) f(\boldsymbol{Z}_{1:9,i}, \boldsymbol{Z}_{11:n,i-1}|\phi_i, \sigma_i^2) f(\phi) f(\sigma^2)}{f(\boldsymbol{Y}|\boldsymbol{\theta}_{i-1}) f(\boldsymbol{\beta}) f(Z_{10,i-1}|\boldsymbol{Z}_{1:9,i}, \boldsymbol{Z}_{11:n,i-1}, \phi_i, \sigma_i^2) f(\boldsymbol{Z}_{1:9,i}, \boldsymbol{Z}_{11:n,i-1}|\phi_i, \sigma_i^2) f(\phi) f(\sigma^2)}$$

$$r \propto \frac{f(\boldsymbol{Y}|\boldsymbol{\theta}_i) f(Z_{10,i}|\boldsymbol{Z}_{1:9,i}, \boldsymbol{Z}_{11:n,i-1}, \phi_i, \sigma_i^2)}{f(\boldsymbol{Y}|\boldsymbol{\theta}_{i-1}) f(Z_{10,i-1}|\boldsymbol{Z}_{1:9,i}, \boldsymbol{Z}_{11:n,i-1}, \phi_i, \sigma_i^2)}$$

- And $f(\boldsymbol{Z})$, $f(Z_i|\boldsymbol{Z}_{-i})$, etc. are $MVN\{\cdot, \Sigma^*\}$, where Σ^* either is, or is some function of, $\Sigma_{\boldsymbol{s}}$; with PDF kernal containing $\Sigma_{\boldsymbol{s}}^{*-1}$, (containing ϕ_i, σ_i^2).

Next Steps

- Convergence, more iterations for `m1 <- spGLM(...)`
- Add Charlotte requested feature to Shiny app
- Add var. res. map to shiny app
- Write. Add ending to chapter one. Start chapter two: introduce random effect, SPGLMM, stan/dimensionality issues/refinement (appendix?), Finley predictive process model dimension reduction
- Fit predictive process model in Stan?

February 6, 2017

- Fit predictive process model in Stan.
  - [Neal et al., 2011] Figure 5.4, Figure 5.5, Figure 5.6 illustrate the much better, *smarter* mixing with HMC over random-walk Metropolis.
  - Samples more correlated from random-walk Metropolis [Neal et al., 2011]
  - Speed (LondonR rstan presentation)
  - No-U-Turn sampler [Hoffman and Gelman, 2014];
  - rstan: "full Bayesian inference using the No-U-Turn sampler (NUTS), a variant of Hamiltonian Monte Carlo (HMC)" [Stan Development Team, 2016]
- Idea of day: Collapse all locations to knot locations, estimate ϕ and σ^2. What else?
- Going to try to create $N \times M$ covariance(obs, knots) matrix in stan; pg 387 in Stan the Manual; did it.
- Random side-note - **Clayton Copulas** look like my hit distribution.
- Maybe we should do a WAY BETTER JOB of what Cross and Sylvan did. Use league averages, exponential covariance structure with predictive process model knots to estimate covariance parameters. Get rid of hokey r, θ business.
- Could have Ben show me how to access the mysterious, high power processors we have access to.

Knot Selection

[Banerjee et al., 2008]

- In forest biomass example Banerjee et al. [2008] says "With only 36 knots the distance between adjacent knots (40 km) seemed to exceed the effective spatial ranges that were supported by the data and **led to unreliable convergence of process parameters**." (Boldface mine)
- "...knot selection is required and as we demonstrated in Section 5.1 some sensitivity to the number of knots is expected. Although for most applications a reasonable grid of knots should lead to robust inference, with **fewer knots** the separation between them increases and estimating random fields with **fine scale spatial** dependence becomes **difficult**. Indeed, learning about **fine scale spatial dependence** is always a **challenge** (see, for example, Cressie (1993), page 114)." (boldface mine)

Meeting

Charlotte

- Get around "for" loop somehow?
- Try binning distances idea?
- Look at HMC convergence more closely, make sure this is worth it.
- Websites for HTML/CSS (Codeacademy),
  C++ (http://adv-r.had.co.nz/Rcpp.html),
  GitHub (http://happygitwithr.com/)

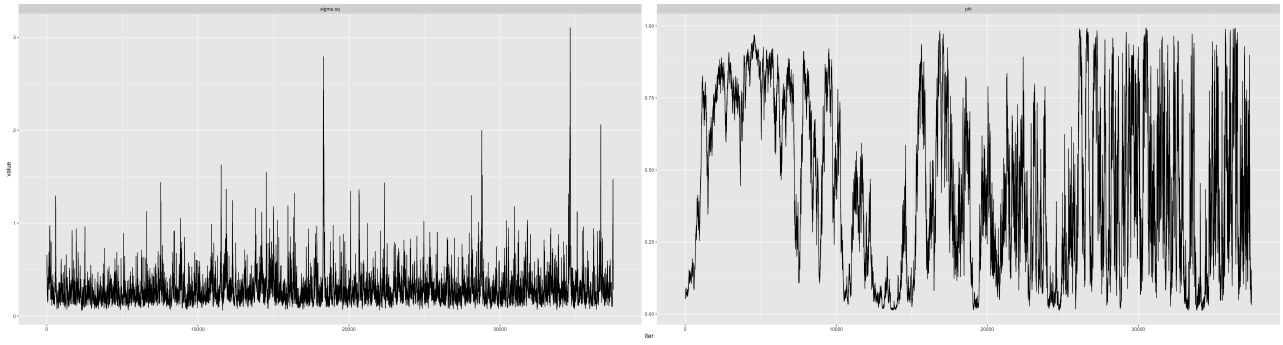
Alix

- Email Debashis again if I feel like it, drop Alix's name
- **Anything in paper about how to choose the number of knots? Look into it.**
- Just choose a hitter with fewer swings. Russian pencil method.
- Try with way fewer knots. Ten knots.
- Email Andy if I feel like it.
- Alix doesn't want to try the aggregating thing yet, says I haven't struggled with predictive process models enough yet. (Admittedly, it has only been a few weeks or so)
- Fix βs, estimate covariance parameters? Try it.
- Start thinking about cross-validation.
- Run the long, five hour estimations, do something else while it goes.
- Alix thinks–and I reluctantly agree–that sometimes I am too quick to say "good enough." ...as I was with package question on the Mango "R Coding Test."

Fix Covariates

- spBayes, n=3000, one chain, 37500 iteration

```
           Mean     SD  Naive SE Time-series SE
XB       1.0000 0.0000 0.0000000       0.000000
sigma.sq 0.2921 0.1850 0.0009551       0.006772
phi      0.4318 0.2886 0.0014905       0.032011
```



February 13, 2017

Miscellaneous

- "Approximate Bayesian Inference for Latent Gaussian models using integrated nested Laplace approximations" [Rue et al., 2009] This looks cool. I want to try this.
- Review "posterior predictive distribution" (need Bayesian book)
- Scoring rules, [Bickel, 2007]
- Diagnose **convergence**
- "As with any knot based method, selection of knots is a challenging problem... Suppose for the moment that m is given. We are essentially dealing with a problem that is analogous to a **spatial design problem**..." [Finley et al., 2009a]
- **Spatial design problem**
- "There is a rich literature in spatial design with is summarized in, e.g., the recent paper of Xia et al. [2006]"
- "Approximately optimal spatial design approaches for environmental health data." [Xia et al., 2006]
- "Spatial sampling design for parameter estimation of the covariance function" [Zhu and Stein, 2005]
- **KL distance** - "Expressed in the language of Bayesian inference, the KullbackLeibler divergence from Q to P, denoted $D_{KL}(P||Q)$, is a measure of the information gained when one revises one's beliefs from the prior probability distribution Q to the posterior probability distribution P. In other words, it is the amount of information lost when Q is used to approximate P." (Wikipedia)

"Hierarchical spatial models for predicting tree species assemblages across large domains"[Finley et al., 2009b]; multinomial

- "While most of the models we formulate can possibly be estimated using maximum likelihood or variants thereof, we adopt a Bayesian approach [e.g., Gelfand et al. (2003)]. This is attractive, as it offers exact inference for the random spatial coefficients, and that too with non-Gaussian data, by delivering an entire posterior distribution at both observed and unobserved locations. Spatial interpolation for processes that are neither observed nor arise as residuals appears inaccessible with classical likelihood-based methods. On the other hand, Bayesian model fitting involves rather specialized Markov chain Monte Carlo (MCMC) methods [see, e.g., Robert and Casella (2005)] that raise concerns about computational expense and reproducibility of inference. These concerns have, however, started to wane with the delivery of relatively simpler R packages (www.r-project.org), including mcmc, MCMCpack, geoRglm and spBayes, that help automate such methods and diagnose convergence."
- "While our **primary contribution here lies in the novel application**, we also offer several methodological advancements."
- "Three MCMC chains of 75,000 iterations were run for each model. Then posterior inference was based on $3 \times 50{,}000 = 150{,}000$ post burn-in samples."
- Justifications (pg 7) for PPMs over other methods, for dealing with large spatial datasets: (i) adapts easily to multivariate processes, (ii) spatial interpolation convenient, (iii) non-Gaussian, (iv) Flexible, multi-parameter with several hyperparameters (over INLA, Integrated Nested Laplace Approximations)

- "With irregular locations, however, we may encounter substantial areas of sparse observations where placing would amount to wastage, possibly leading to inflated variance estimates and slower convergence. More practical space-covering designs [e.g., Royle and Nychka (1998)] can yield a representative collection of knots that better cover the domain."
  - This could be a contribution. Compare my "var-res-grid centers as knots" to Nychka.
- "We consider several different scoring rules to evaluate the predictive performance of the candidate models. A scoring rule provides a summary measure for evaluating a probabilistic prediction given the predictive distribution and the observed outcome. " pg 10
- "Gneiting and Raftery [2007] offer four scoring rules for prediction of categorical variables" page 10.
  - These guys have other papers, and LOTS of citations.
- **Confusion matrix** - (google search) A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. (google search)
- "We also compare our spatially-varying multinomial logistic regression models to common benchmark methods" page 11
- "For estimating predicive process models, we used 154, 200, and 254 knots over the domain."
  - Could the efficacy of the PPM with X knots inform, in reverse, a method for choosing the cutoff?
- "The spatially-varying coefficients model was the most computationally challenging, with each chain of the 254 knot model taking 5 hours to complete. **The `CODA` package in R (www.r-project.org) was used to diagnose convergence by monitoring mixing using GelmanRubin diagnostics and autocorrelations** [see, e.g., Gelman et al. (2004), Section 11.6]. Acceptable convergence was diagnosed within 25,000 iterations and, therefore, 150,000 samples (3 x 50,000) were retained for posterior analysis."
  - DIAGNOSTICS.

Dissertation (Middle Chunks) Outline

- Var-Res heat maps
  - Heat maps
  - Shiny
- MLE model
  - Profile plots to show interpretability potential
- Random effect model
  - MCMC - Markov Chain Monte Carlo
  - Stan - HMC
    - Code code code (communications, and Trangucci2017 handout)
    - Hamiltonian Monte Carlo [Neal et al., 2011]
  - spBayes, Predictive process models
    - Knot quantity - seems like comparison at range of values
    - placement decisions - my var-res-grid centers vs. [Royle and Nychka, 1998]

    - Metropolis nuts and bolts
    - **Convergence** - Finley et al. [2011] talks about convergence, Gelman-Rubin diagnostics ([Gelman et al., 2014]), mixing, and `CODA` package in R, classification confusion matrix, on page 12.

- Include Cross kriging, empirical Bayes boxes
- Model validation, comparison
    - Hosmer-Lemeshow - goodness of fit test
    - Finley et al. [2011] has "5.1 Model validation and benchmark comparisons" section
    - Finley et al. [2011] cites Gneiting and Raftery [2007] regarding scoring rules, etc., as have 1577 other papers

Scoring rule

- Wikipedia: "In decision theory, a score function, or scoring rule, measures the accuracy of probabilistic predictions. It is applicable to tasks in which predictions must assign probabilities to a set of mutually exclusive discrete outcomes. The set of possible outcomes can be either binary or categorical in nature, and the probabilities assigned to this set of outcomes must sum to one (where each individual probability is in the range of 0 to 1).

  ...Proper scoring rules are used in meteorology, finance, and pattern classification where a forecaster or algorithm will attempt to minimize the average score to yield refined, calibrated probabilities (i.e. accurate probabilities)."
- Wikipedia: "An example of probabilistic forecasting is in meteorology where a weather forecaster may give the probability of rain on the next day. One could note the number of times that a 25% probability was quoted, over a long period, and compare this with the actual proportion of times that rain fell. If the actual percentage was substantially different from the stated probability we say that the forecaster is poorly calibrated."
- Examples: logarithmic, Brier/quadratic, spherical

Strictly proper scoring rules, prediction, and estimation

[Gneiting and Raftery, 2007]

-

An Algorithm for the Construction of Spatial Coverage Designs... [Royle and Nychka, 1998]

- "Space-filling "coverage" designs are spatial sampling plans which optimize a distance-based criterion"

Spatial Sampling Design

- Note: Random effect is Gaussian, response is not.
- One option: simulation study
    1. Define multiple knot structures $k = 1, ..., n_k$
    2. Generate GRF data, using exponential covariance

3. Estimate parameters using each knot structure (Bayesian PPM approach; same code, $X\beta$ fixed), store estimates;
4. Iterate
5. Compare (via variance?) results

- "2.4 Selection of knots" [Banerjee et al., 2008] is a treasure trove of talk about this.
- "Bayesian Sampling Design" [Diggle and Lophaven, 2006] looks pretty applicable. (i) lattice plus close pairs, and (ii) lattice plus infill

Gaussian Predictive Process Models for Large Spatial Data Sets

[Banerjee et al., 2008]

- "We project the original process onto a subspace that is generated by realizations of the original process at a specified set of locations."
- "The w(s) are spatial random effects, providing local adjustment (with structured dependence) to the mean, interpreted as capturing the effect of unmeasured or unobserved covariates with spatial pattern."
- "We regard the predictive process model as a competing model with computational advantages."
- Other approaches "may be challenging for the hierarchical models" here, with "spatial modelling with random effects at the second stage of the specification. We have no 'data' to provide an empirical covariance function."
- "A direct assessment of knot performance is comparison of the covariance function of the parent process with that of the predictive process..."
- "There is little additional information in varying m for a given ϕ and ν (Matern params). What matters is the size of the range relative to the spacing of the grid for the knots."
- "Choice of m is governed by computational cost and sensitivity to choice... implement the analysis over different choices of m... consider run time... stability of predictive inference."
- "...improvement in estimation with increasing number of knots."
- "Tables 1–3 suggest that estimation is more sensitive to the number of knots than to the underlying design."
- Convergence diagnostics, CODA package
- "With only 36 knots the distance between adjacent knots (40 km) seemed to exceed the effective spatial ranges that were supported by the data and led to unreliable convergence of process parameters."
- "Our examples in Section 5.1 showed that even with fairly complex underlying spatial structures the predictive process model could effectively capture most of the spatial parameters with 529 knots (irrespectively of whether the total number of locations was 3000 or 15000).
- "In fact, the predictive process approach within a full MCMC implementation is perhaps limited to the order of 10^4 observations on modest single-processor machines."

20 February 2017

********** Begin INLA-SPDE-RINLA **********

Debashis meeting

- INLA is oversold; Does Matern $\nu = 1, 2, 3$ only (no exponential, $\nu = 1/2$)
- Matern $\nu = 1$ is reasonable-ish.
- Could add structure to cov. matrix, make it sparse, make cov = 0 for $d > d_0$
- SPDE is oversell too.
- INLA is basically Taylor series
- Will be bias in estimates, for binary data in particular.
- MCMC might work better with sparse cov matrix, R has a package for sparse matrices; does Stan use sparse matrices?

February 27, 2017

Matern Covariance (1)

$$C_\nu(d) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\frac{d}{\rho}\right)^\nu K_\nu \left(\sqrt{2\nu}\frac{d}{\rho}\right)$$

Where Γ is the gamma function, and K_ν is the modified Bessel function, ρ and ν are non-negative covariance parameters

- Recall, $\Gamma(n) = (n-1)!$
- $\nu = 1/2$ gives exponential covariance

$$C_{1/2}(d) = \sigma^2 exp(-d/\rho)$$

- Stationary and isotropic if Euclidean distance

Approximate Bayesian Inference for Hierarchical Gaussian Markov Random Field Models [Rue and Martino, 2007]

- $\mathcal{O}$ definition, Wikipedia

$$f(x) = \mathcal{O}(g(x)) \text{ as } x \to \inf \text{ iff } \exists M, x_0 \text{ such that } |f(x)| \leq M|g(x)| \text{ for all } x \geq x_0$$

- Gaussian Markov Random Field
  - $\boldsymbol{x} = \{x_i : i \in \mathscr{V}\}$
  - "$\boldsymbol{x}$ is a $n = |\mathscr{V}|$-dimensional Gaussian random vector with additional conditional independence/-Markov properties"
  - $\mathscr{V} = \{1, \ldots, n\}$
  - Graph $\mathscr{G} = \{\mathscr{V}, \mathscr{E}\}$, with vertices, edges.
  - "Two nodes, x_i and x_j are conditionally independent given the remaining elements of $\boldsymbol{x}$, if and only if $\{i, j\} \notin \mathscr{E}$.

  - "Then $\boldsymbol{x}$ is a GMRF with respect to $\mathscr{G}$."
  - "The edges in $\mathscr{E}$ are in one-to-one correspondence with the non-zero elements of the precision matrix of $\boldsymbol{x}$, $\boldsymbol{Q}$, in the sense that $\{i, j\} \in \mathscr{E}$ if and only if $Q_{ij} \neq 0$ for $i \neq j$."
  - "When $\{i, j\} \in \mathscr{E}$ we say that i and j are neighbours, which we denote $i \sim j$."
- Nice destription of hierarchical model architecture... "One of the main areas of application for GMRFs is that of (Bayesian) hierarchical models. A hierarchical model is characterised by **several stages** of observables and parameters. The **first stage**, typically, consists of distributional assumptions for the observables conditionally on latent parameters. For example if we observe a time series of counts y, we may assume, for y_i, $i \in D \subset R^?$ a Poisson distribution with unknown mean λ_i. Given the parameters of the observation model, we often assume the observations to be conditionally independent. The **second stage** consists of a prior model for the latent parameters λ_i or, more often, for a particular function of them. For example, in the Poisson case we can choose an exponential link and model the random variables $x_i = log(\lambda_i)$. At this stage **GMRFs provide a flexible tool to model the dependence** between the latent parameters and thus, implicitly, the dependence between the observed data. This dependence can be of various kind, such as temporal, spatial, or even spatiotemporal. The **third stage** consists of prior distributions for the unknown hyperparameters . These are typically precision parameters in the GMRF."
- "We propose a deterministic alternative to MCMC based inference... computed almost instant[ly]... proves to be quite accurate."
- Oops. **Gaussian Markov Random Field ≠ Gaussian Random Field**;
  GMRF has different properties; think graph theory edges, and non-edge-connected nodes have zero correlation, hence sparse matrices for INLA

Miscellaneous

- Covariance tapering? $d < d_0 \rightarrow d = 0$, then what?
- **range parameter: the Euclidean distance where $x(s_0)$ and $x(s_1)$ are almost independent.** [Lindgren et al., 2011]

Meeting

- I **CAN** make **half** of my talk "Lessons Learned"!!
- "Lessons Learned"
  - Zero-th problem (Journal club)
  - JSM talk, already published – do a lit review!
  - Write and show
  - "Secrets of Research Success" -Hugh Kearns
  - Koutsoyiannis
  - Richard Feynman
  - "Thinking, Fast and Slow"
  - "Set a pace you can keep."
- INLA
- "You're ready to be done." -Alix, as in "You have gained the research skills you need to go out there and be successful."

- "This is great! You're coming in here telling **me** what your dissertation is!"
- My plan for chapters is legit.
- Talked about family, job search, research, seminar talk, Chris Wolf.

March 6, 2017

An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach [Lindgren et al., 2011]

- "A GMRF is a discretely indexed Gaussian field x, where the full conditionals" depend only on a small set of symmetric neighbor relationships, which yields sparse matrices that lend themselves to approximations—INLA
- The GMRF "computational gain comes from the fact that the zero pattern of the precision matrix Q (the inverse covariance matrix) relates directly to the notion of neighbours..."
- Matrix **Q: the inverse covariance matrix**
- "The result is a basis function representation with piecewise linear basis functions, and Gaussian weights with Markov dependencies determined by a general triangulation of the domain." [Lindgren et al., 2011]
  - Recall "basis" from linear algebra, where a set of linearly independent vectors span a space
  - function space - space of *functions*
  - basis functions - a set of functions from which can build any function in the function space
- Matern (2):
$$r(\boldsymbol{u},\boldsymbol{v}) = \frac{\sigma^2}{2^{\nu-1}\Gamma(\nu)}(\kappa||\boldsymbol{u}-\boldsymbol{v}||)^\nu K_\nu(\kappa||\boldsymbol{u}-\boldsymbol{v}||)$$
  - scaling parameter $\kappa = \rho$ range parameter
  - Empirically derived: $\rho = \sqrt{8\nu}/\kappa$
- Linear fractional SPDE:
$$(\kappa^2 - \Delta)^{\alpha/2} x(\boldsymbol{u}) = \mathcal{W}(\boldsymbol{u})$$
  - $\alpha = \nu + d/2$
- Δ is Laplacian:
$$\Delta = \sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2}$$
- Marginal Variance:
$$\sigma^2 = \frac{\Gamma(\nu)}{\Gamma(\nu + d/2)(4\pi)^{d/2}\kappa^{2\nu}}$$

Bayesian Spatial Modelling with R-INLA [Lindgren and Rue, 2015]

- "...as discussed in Lindgren et al. (2011), one can express a large class of random field models as solutions to continuous domain stochastic partial differential equations (SPDEs), and write down explicit links between the parameters of each SPDE and the elements of precision matrices for weights in a discrete basis function representation."
- "As discussed in the introduction, **an alternative to traditional covariance based modelling is to use SPDEs**, but carry out the practical computations using Gaussian Markov random field (GMRF) representations. This is done by **approximating the full set of spatial random functions with weighted sums of simple basis functions**, which allows us to hold on to the continuous interpretation of space, while the **computational algorithms** only see discrete structures with Markov properties. Beyond the main paper Lindgren et al. (2011), this is further discussed by Simpson, Lindgren, and Rue (2012a,b)."
- "The simplest model for (spatial field) $x(\boldsymbol{s})$ currently implemented in R-INLA is the SPDE/GMRF version of the stationary Matern family, obtained as the stationary solutions to

$$(\kappa^2 - \Delta)^{\alpha/2}(\tau x(\boldsymbol{s})) = \mathcal{W}(\boldsymbol{s}),\ \boldsymbol{s} \in \Omega$$

  where

  - Δ is the Laplacian
  - κ is the spatial scale parameter
  - α controls the smoothness of the realisations
  - τ controls the variance
  - Ω is the spatial domain
  - $\mathcal{W}(\boldsymbol{s})$ is a Gaussian spatial white noise process

  Whittle (1954, 1963) shows stationary solutions on $\mathbb{R}^d$ have Matern covariances,

$$\mathrm{COV}(x(\mathbf{0}), x(\boldsymbol{s})) = \frac{\sigma^2}{2^{\nu-1}\Gamma(\nu)}(\kappa||s||)^\nu K_\nu(\kappa||s||)$$

  The parameters in the two formulations are coupled so that the Matern smoothness is $\nu = \alpha - d/2$ and marginal variance is

$$\sigma^2 = \frac{\Gamma(\nu)}{\Gamma(\alpha)(4\pi)^{d/2}\kappa^{2\nu}\tau^2}$$

  Exponential covariance: $\nu = 1/2$; (i) for $d = 1 \to \alpha = 1$, (ii) for $d = 2 \to \alpha = 3/2$"
- "The models discussed in Lindgren et al. [2011] and implemented in R-INLA are built on a basis representation

$$x(\boldsymbol{s}) = \sum_{k=1}^{n} \psi_k(\boldsymbol{s}) x_k$$

  where

  - $\psi_k(\cdot)$ are deterministic basis functions, and
  - the joint distribution of the weight vector $\boldsymbol{x} = \{x_1, \ldots, x_n\}$ is chosen so that the distribution of the functions $x(\boldsymbol{s})$ approximates the distribution of solutions to the SPDE on the domain."
  - piecewise polynomial basis functions
  - use Finite Element Method - project the SPDE onto the basis representation
-

Miscellaneous

I got `inla(...)` to run!!

- `geoR` package
- I am trying to generate Matern data with `grf(...)`, then estimate parameters with `inla(...)`. However, basic input to `inla(...)` implies INLA, without SPDE connection, and thus data on a grid. This means using my actual pitch locations from `hitter` will not work.

- Try to generate data on a grid, use `inla(...)` to estimate parameters
- inverse Precision matrix equals covariance matrix: $Q^{-1} = \Sigma$
- Precision matrix Q equals inverse covariance matrix: $Q = \Sigma^{-1}$

Simulate Matern data

`geoR` package, `grf(...)` function
Matern (3)

$$\rho(u; \phi, \kappa) = \{2^{\kappa-1}\Gamma(\kappa)\}^{-1}(u/\phi)^{\kappa}K_{\kappa}(u/\phi)$$

- u - vector/matrix/array with distances between pairs of data locations
- ϕ - range parameter, > 0
- κ - smoothness parameter, > 0
- $K_{\kappa}(\cdot)$ - modified Bessel function of third kind of order κ

Matern (4)

[Schabenberger and Gotway, 2004]

$$C(h) = \sigma^2 \frac{1}{\Gamma(\nu)} \left(\frac{\theta h}{2}\right)^{\nu} 2K_{\nu}(\theta h),\ \nu > 0, \theta > 0$$

- θ governs range of spatial dependence
- ν: smoothness increases with ν
- Page 143: The Matern Class of Covariance Functions
- Page 199: **THREE** Matern parameterizations
- Page 210: Bessel Functions
- Whittle Model, $\nu = 1$ (INLA/SPDE world, I think)

  $$C(h) = \sigma^2 \theta h K_1(\theta h)$$

  Whittle considered this the "elementary model" in $\mathbb{R}^2$

Stochastic Calculus

Stochastic Differential Equation (wikipedia)

- "A heuristic but helpful interpretation of the stochastic differential equation (of continuous time stochastic process X_t) is that in a small time interval of length δ the stochastic process X_t changes its value by an amount that is normally distributed (for example) with expectation $\mu(X_t, t)\delta$ and variance $\sigma(X_t, t)^2\delta$ and is independent of the past behavior of the process."

Stochastic Calculus [Mao, 2007]

- Integral of random process is another random process. Random process not integrable in traditional sense, so stochastic calculus created, by Ito, Ito Calculus. Use Brownian motion as some sort of reference point.

$$Y_t = \int_0^t H_s dX_s$$

  Integrand and integrator are stochastic processes.

March 27, 2017

Lecture Slides - GMRF, Dependent Spatial Data

[Lindstrom et al., 2011]

- Define problem, explain difficulty, define GMRF, **sparse** precision matrix Q^{-1}, Markov property, Precision matrix construction hard
- Matern family:

$$r(\boldsymbol{u}) = \frac{\sigma^2}{2^{\nu-1}\Gamma(\nu)}(\kappa||\boldsymbol{u}||)^\nu K_\nu(\kappa||\boldsymbol{u}||)$$

- Random fields with Matern covariance are solutions to:

$$(\kappa^2 - \Delta)^{\alpha/2} x(\boldsymbol{s}) = \mathcal{W}(\boldsymbol{s}),$$

  - $\alpha = \nu + d/2$
  - Δ is Laplacian: $\Delta = \sum_i \frac{\partial^2}{\partial x_i^2}$

- Construct GMRF from SPDE
  - Construct the solution as a finite basis expansion

$$x(\boldsymbol{s}) = \sum_k \psi_k(\boldsymbol{s}) x_k$$

    with a suitable distribution for the weights $\{x_k\}$.
  - Sotchastic weak solution given by weights $\{x_k\}$ such that the joint distribution fulfills

$$\sum_i \left\langle \psi_j, (\kappa^2 - \Delta)^{\alpha/2} \psi_i x_i \right\rangle \stackrel{D}{=} \langle \psi_j, \mathcal{W} \rangle \ \forall \boldsymbol{j}$$

Lecture Slides - GMRFs

[Lindstrom, 2014]

-

Lecture Slides - Latent Gaussian Processes and SPDEs

[Lindstrom, 2016]

-

As of Now

- INLA, SPDE, R-INLA
- Jobs
  - Post docs
  - Talk to Alix, Charlotte
  - Jaime, MIT/Cambridge John
  - Apply to junior quant/tech positions
- Writing
- Python (Code Academy), Github, Machine Learning (Dummies), C++ (Dummies)
-
- Reschedule defense. Replace Lisa?
- At least 2 weeks before your final oral examination:
  (1) Use online form to schedule your final oral examination,
  (2) Distribute a defendable copy of your thesis to your comittee,
  (3) Bring in or email pre-text pages of your thesis to the Graduate School and
  (4) submit a diploma application (**EXCEPT for SPRING Term** completion, when you **must submit by FIRST week** of Spring Term).
-
- Scoring rules - model evaluation/comparison (two papers)
- Spatial design analysis - knot selection (two papers)
- Convergence diagnostics (CODA package, Finley paper)
- Simulate, or real data?

References

Andrew O Finley, Sudipto Banerjee, and Bradley P Carlin. spbayes: an r package for univariate and multivariate hierarchical point-referenced spatial models. Journal of Statistical Software, 19(4):1, 2007.

Radford M Neal et al. Mcmc using hamiltonian dynamics. Handbook of Markov Chain Monte Carlo, 2: 113–162, 2011.

Andrew O Finley, Huiyan Sang, Sudipto Banerjee, and Alan E Gelfand. Improving the performance of predictive process modeling for large datasets. Computational statistics & data analysis, 53(8):2873–2884, 2009a.

Andrew O Finley, Sudipto Banerjee, and Ronald E McRoberts. Hierarchical spatial models for predicting tree species assemblages across large domains. The annals of applied statistics, 3(3):1052, 2009b.

Andrew O Finley, Sudipto Banerjee, and David W MacFarlane. A hierarchical model for quantifying forest variables over large heterogeneous landscapes with uncertain forest areas. Journal of the American Statistical Association, 106(493):31–48, 2011.

Rajarshi Guhaniyogi, Andrew O Finley, Sudipto Banerjee, and Alan E Gelfand. Adaptive gaussian predictive process models for large spatial datasets. Environmetrics, 22(8):997–1007, 2011.

Andrew O Finley, Sudipto Banerjee, and Alan E Gelfand. Bayesian dynamic modeling for large space-time datasets using gaussian predictive processes. Journal of geographical systems, 14(1):29–47, 2012.

Jo Eidsvik, Andrew O Finley, Sudipto Banerjee, and Håvard Rue. Approximate bayesian inference for large spatial datasets using predictive process models. Computational Statistics & Data Analysis, 56(6): 1362–1380, 2012.

Andrew O Finley, Sudipto Banerjee, and Alan E Gelfand. spbayes for large univariate and multivariate point-referenced spatio-temporal data models. arXiv preprint arXiv:1310.8192, 2013.

Sudipto Banerjee, Alan E Gelfand, Andrew O Finley, and Huiyan Sang. Gaussian predictive process models for large spatial data sets. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 70 (4):825–848, 2008.

J Andrew Royle and Doug Nychka. An algorithm for the construction of spatial coverage designs with implementation in splus. Computers & Geosciences, 24(5):479–488, 1998.

Matthew D Hoffman and Andrew Gelman. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. Journal of Machine Learning Research, 15(1):1593–1623, 2014.

Stan Development Team. Package rstan - R Interface to Stan, Version 2.14.1, 2016. URL `http://mc-stan.org`.

Håvard Rue, Sara Martino, and Nicolas Chopin. Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations. Journal of the royal statistical society: Series b (statistical methodology), 71(2):319–392, 2009.

J Eric Bickel. Some comparisons among quadratic, spherical, and logarithmic scoring rules. Decision Analysis, 4(2):49–65, 2007.

Gangqiang Xia, Marie Lynn Miranda, and Alan E Gelfand. Approximately optimal spatial design approaches for environmental health data. Environmetrics, 17(4):363–385, 2006.

Zhengyuan Zhu and Michael L Stein. Spatial sampling design for parameter estimation of the covariance function. Journal of Statistical Planning and Inference, 134(2):583–603, 2005.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. Journal of the American Statistical Association, 102(477):359–378, 2007.

Andrew Gelman, John B Carlin, Hal S Stern, and Donald B Rubin. Bayesian data analysis, volume 2. Taylor & Francis, 2014.

Peter Diggle and Søren Lophaven. Bayesian geostatistical design. Scandinavian Journal of Statistics, 33(1): 53–64, 2006.

HÅvard Rue and Sara Martino. Approximate bayesian inference for hierarchical gaussian markov random field models. Journal of statistical planning and inference, 137(10):3177–3192, 2007.

Finn Lindgren, Håvard Rue, and Johan Lindström. An explicit link between gaussian fields and gaussian markov random fields: the stochastic partial differential equation approach. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 73(4):423–498, 2011.

Finn Lindgren and Håvard Rue. Bayesian spatial modelling with r-inla. Journal of Statistical Software, 63 (19), 2015.

Oliver Schabenberger and Carol A Gotway. Statistical methods for spatial data analysis. CRC press, 2004.

Xuerong Mao. Stochastic differential equations and applications. Elsevier, 2007.

Johan Lindstrom, Finn Lindgren, and Havard Rue. Gaussian markov random fields: Efficient modelling of spatially dependent data. Lecture Slides, 2011. Centre for Mathematical Sciences, Lund University.

Johan Lindstrom. Gaussian markov random fields. Lecture Slides, 2014. Pan-American Advanced Study Institute, Buzios.

Johan Lindstrom. Latent gaussian processes and stochastic partial differential equations. Lecture Slides, 2016. Centre for Mathematical Sciences, Lund University.