Modelling County Data in QWI.

Jul 22nd, 2018

QWI example. Subset: MO state.

Exploring and modelling for QWI data. Subset: MO state (115 county equivalents).

Available time periods are from 1995, Q1 to 2017, Q3.

The first quarter (1995, Q1) and the most recent quarter (2017, Q3) have no data avilable for monthly average income and are therefore excluded from the analysis.

Furthermore, for computational reasons we limit the analysis to the last 10 year-interval: 2007, Q1 to 2017 Q2 ($T = 38$, $n = 115$. Number of counties invariant over time).

The response variable of interest is the total average monthly income; `EarnS` is such for workers with stable jobs (i.e., the workers who held the same job consistently throughout the quarter). Let's take a particular subset of the data that Bradley et al. (2016) already examined: education sector in MO. This sector is of particular interest, as the paper showed that there is a marked gender income dispariy in the sector (as all other industry groups), and for both genders yields the lowest income on average.

```
library(tidyverse)
## -- Attaching packages ---------------------------------------- tidyverse 1.2.1 --
## ✓ ggplot2 2.2.1     ✓ purrr   0.2.5
## ✓ tibble  1.4.2     ✓ dplyr   0.7.5
## ✓ tidyr   0.8.1     ✓ stringr 1.3.1
## ✓ readr   1.1.1     ✓ forcats 0.3.0
## -- Conflicts ------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
library(parallel)
library(doMC)
## Loading required package: foreach
##
## Attaching package: 'foreach'
## The following objects are masked from 'package:purrr':
##
##     accumulate, when
## Loading required package: iterators
library(ngspatial)
## Loading required package: Rcpp
## Loading required package: batchmeans
## batchmeans: Consistent Batch Means Estimation of Monte Carlo Standard Errors
## Version 1.0-3 created on 2016-07-03.
## copyright (c) 2012-2016, Murali Haran, Penn State University
##                          John Hughes, University of Colorado Denver
## For citation information, type citation("batchmeans").
## Type help(package = batchmeans) to get started.
## ngspatial: Fitting the Centered Autologistic and Sparse Spatial Generalized
## Linear Mixed Models for Areal Data
## Version 1.2-1 created on 2018-01-12.
## copyright (c) 2013-2018, John Hughes, University of Colorado Denver
## For citation information, type citation("ngspatial").
```

```
## Type help(package = ngspatial) to get started.
library(forecast)
source("000_utils.R")

# Missouri state QWI dataset
c_C <- paste(rep("c", 14), collapse="")
c_N <- paste(rep("n", 66), collapse="")
colTypes <- paste0(c_C, c_N) # col. types info. in column_definitions.txt
qwiMO <- read_csv("data/qwi_mo_sa_f_gc_ns_oslp_u.csv", col_types = colTypes) %>%
  filter(year!=1995|quarter!=1, year!=2017|quarter!=3)
```

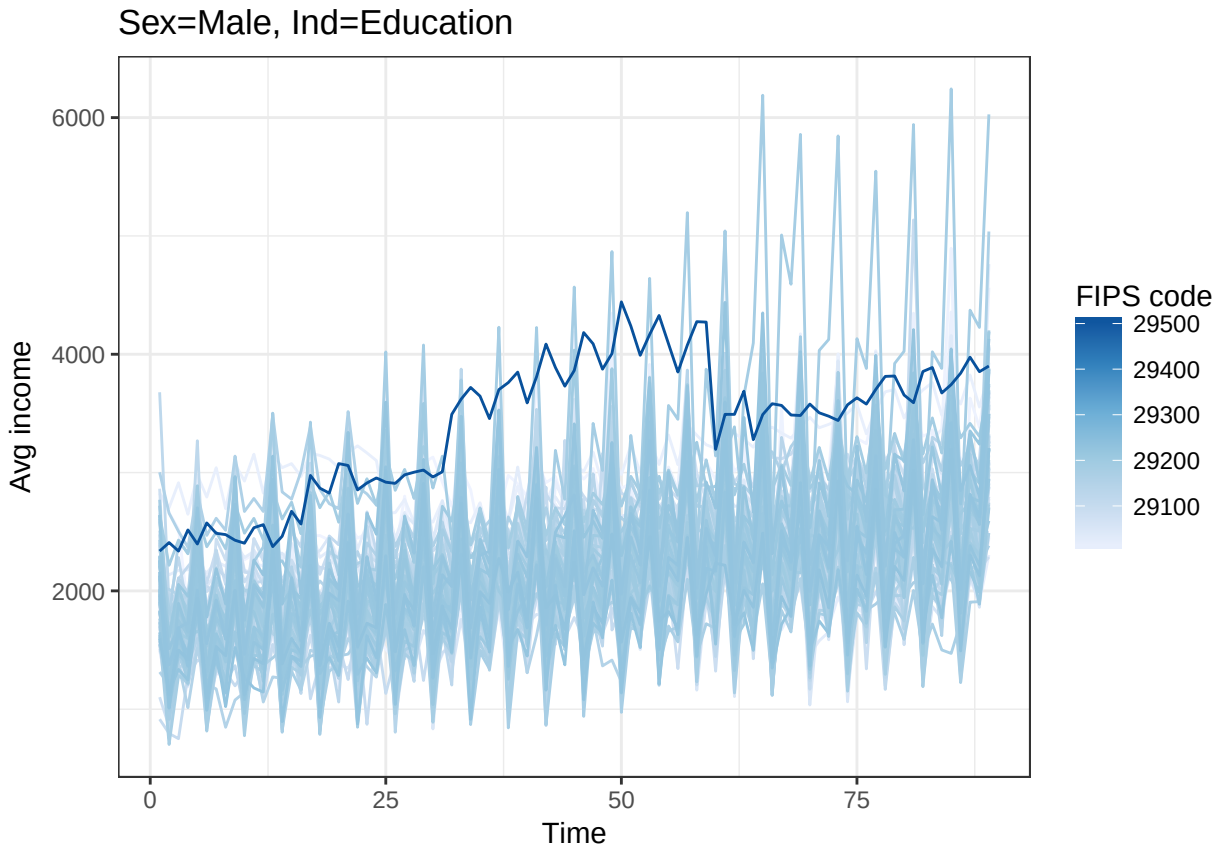
First, we will fit a **descriptive (marginal) model** as proposed in Griffith (2013); that is, we regress the response on exogenous variables and spatiotemporal random effects, which are replaced with eigenvectors obtained by spatiotemporal Moran basis expansion in Griffith (2013). The difference here is that the parameter estimation is done through Bayesian MCMC scheme, as proposed by Hughes and Haran (2013) in a spatial-only setting. Since both publications do not incorporate sources of across-variable variances in a multivariate setting, for our interest we will have to fit two separate models for male and female groups. MSTM as proposed by Bradley et al. (2016) aims at this, which will be our next implementation (and expansion).

The exogenous variable included is the intercept term and time (indexed from 1 to 89 for each quarter). Two different adjacency matrices are tried, one that only has nonzero elements at lag 1, and the other that tries to account for long-lagged "adjacency" (seasonality) through heuristic judgments. The first 100 eigenvectors from Moran basis eigendecomposition are used as spatial regressors.

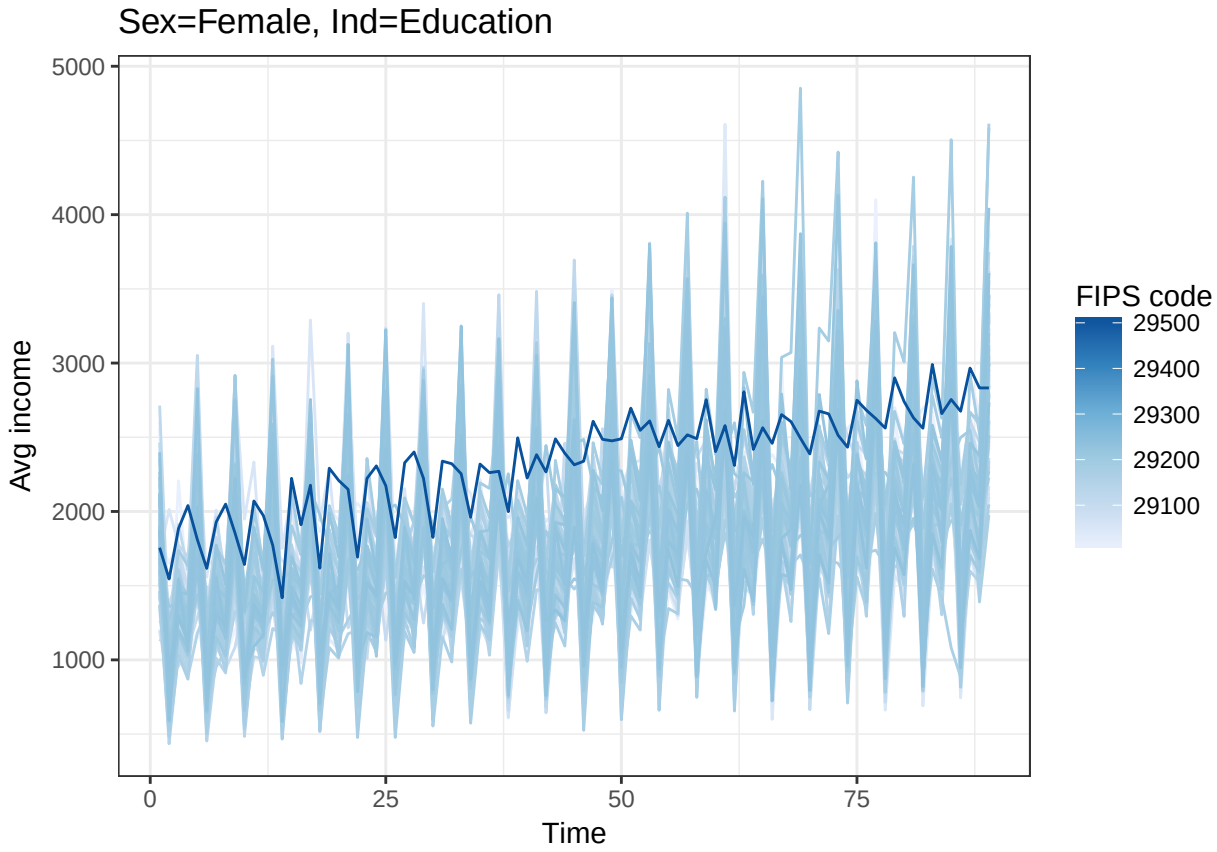
Preliminary Analysis

```
source("005.01_county-datawork.R")
##
## Attaching package: 'Matrix'
## The following object is masked from 'package:tidyr':
##
##     expand
## Parsed with column specification:
## cols(
##   countyname = col_character(),
##   fipscounty = col_character(),
##   neighborname = col_character(),
##   fipsneighbor = col_character()
## )
indicesMO <- which(grepl("^29...$", colnames(A)))
subregionsMO <- g_countyNames[indicesMO] %>%
  strsplit(" County") %>%
  sapply(function(x) x[1]) %>%
  gsub("\\.", "", .) %>%
  tolower() %>% as.character()
subregionsMO[32] <- "de kalb"
subregionsMO[115] <- "st louis city"

## (...) map visualization

# Our data for modelling.
```

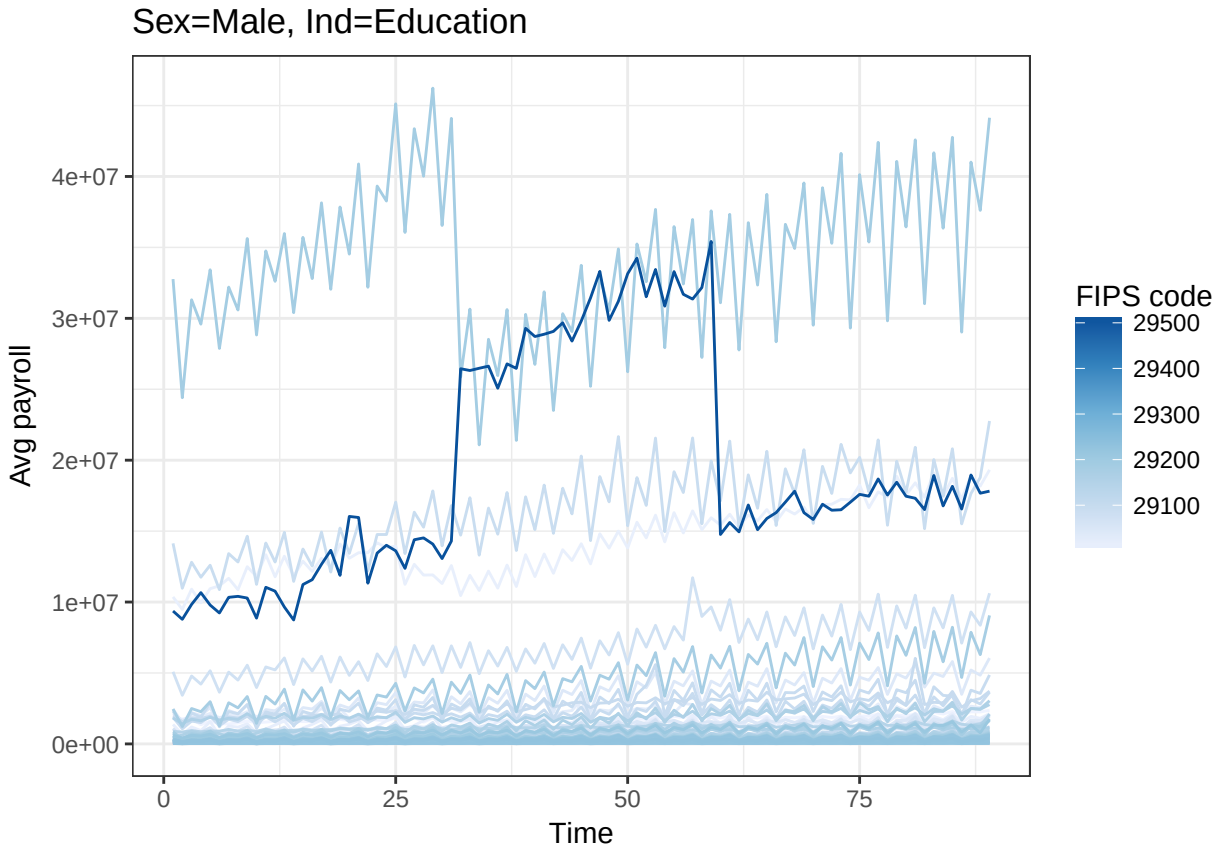
```
c_dataMO <- qwiMO %>%
  filter(geo_level=="C", industry=="61", ind_level=="S", sex!="0") %>%
  mutate(time = as.factor(paste(year, quarter, sep="Q")),
         index = sapply(time, function(s) grep(s, levels(time)))) %>%
  group_by(time, index, geography, sex) %>%
  summarise(AvgIncome = mean(EarnS, na.rm=T),
            Payroll=mean(Payroll, na.rm=T))
## male and female groups
dataMaleMO <- c_dataMO %>% filter(sex=="1") %>% mutate(logInc = log(AvgIncome))
dataFemMO <- c_dataMO %>% filter(sex=="2") %>% mutate(logInc = log(AvgIncome))

## ggplot visualization
theme_set(theme_bw())
ggplot(data=dataMaleMO, aes(index, AvgIncome, group=geography)) +
  geom_line(aes(colour=as.integer(geography)), show.legend=T) +
  scale_colour_distiller(palette="Blues", direction=1) +
  labs(title = "Sex=Male, Ind=Education", x = "Time", y = "Avg income", colour = "FIPS code")
```



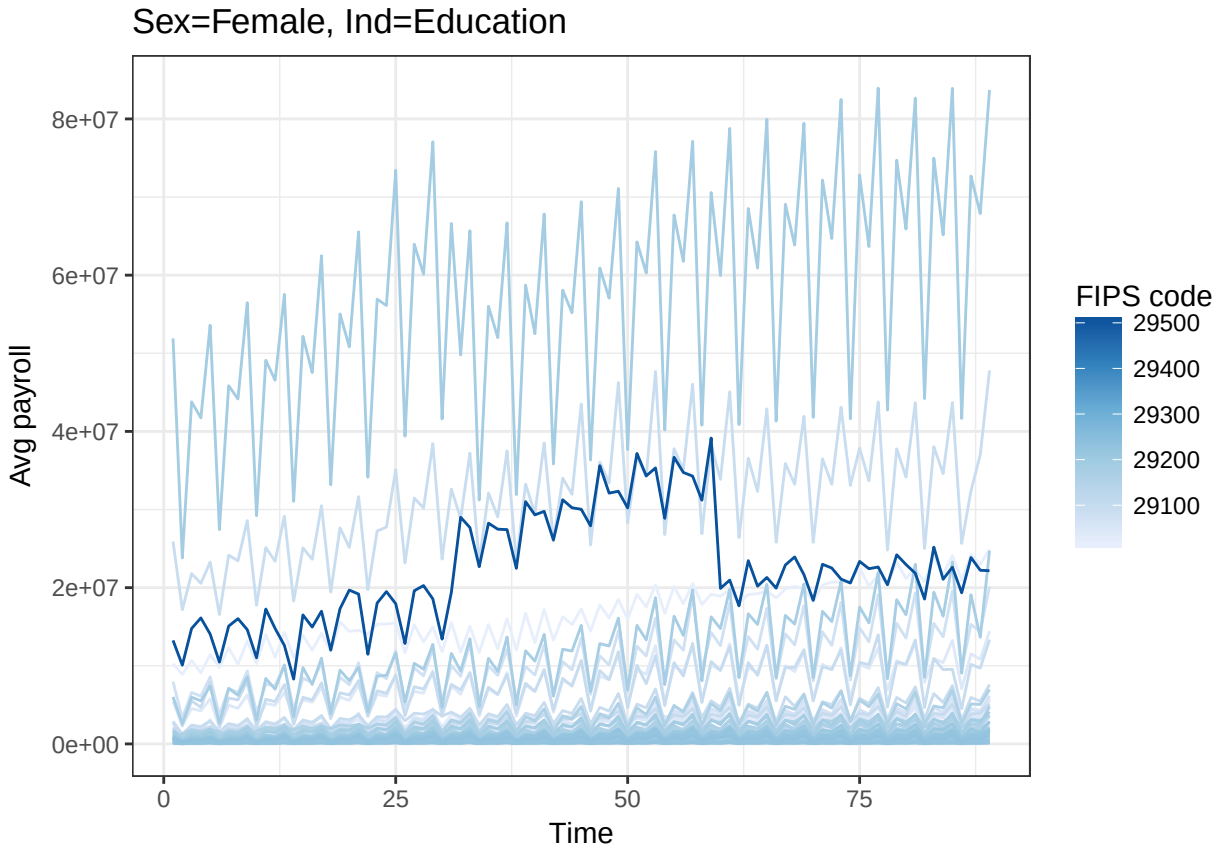

```
ggplot(data=dataFemMO, aes(index, AvgIncome, group=geography)) +
  geom_line(aes(colour=as.integer(geography)), show.legend=T) +
  scale_colour_distiller(palette="Blues", direction=1) +
  labs(title = "Sex=Female, Ind=Education", x = "Time", y = "Avg income", colour = "FIPS code")
```




```
ggplot(data=dataMaleMO, aes(index, Payroll, group=geography)) +
  geom_line(aes(colour=as.integer(geography)), show.legend=T) +
  scale_colour_distiller(palette="Blues", direction=1) +
  labs(title = "Sex=Male, Ind=Education", x = "Time", y = "Avg payroll", colour = "FIPS code")
```




```
ggplot(data=dataFemMO, aes(index, Payroll, group=geography)) +
  geom_line(aes(colour=as.integer(geography)), show.legend=T) +
  scale_colour_distiller(palette="Blues", direction=1) +
  labs(title = "Sex=Female, Ind=Education", x = "Time", y = "Avg payroll", colour = "FIPS code")
```



Preliminary analysis suggests that not only all economic variables have quarterly seasonality (which is expected), but also that some have potentially more complex dependence structures than a simple AR(1) behavior. For a payroll variable, almost half of the counties (51) had ARIMA models fitted that had greater complexity than an AR model of order 1 or seasonal order of 1 and a drift. Often they had AR or SAR order of 2 (or both), indicating enough motivation for a spatiotemporal model that addresses AR(2) dependence of latent state vectors.

```
payArima_l <- list()
c_countycodes <- unique(c_dataMO$geography)
for (i in 1:length(c_countycodes)) {
    payArima_l[[i]] <- ts(filter(dataFemMO, geography==c_countycodes[i])$Payroll, freq=4)
}

# Fit auto.arima to every list at parallel
myCl <- makeCluster(detectCores()-1, type="FORK")
payArModel_l <- parLapply(myCl, payArima_l, function(ts) auto.arima(ts, start.p = 1, start.q = 1))
c_arOrders <- t(sapply(payArModel_l, function(x) x$arma))
c_arOrders[(c_arOrders[,1]>1|c_arOrders[,2]>1|c_arOrders[,3]>1|c_arOrders[,4]>1),1:4]
##       [,1] [,2] [,3] [,4]
##  [1,]    2    0    0    1
##  [2,]    2    0    0    0
##  [3,]    0    0    0    2
##  [4,]    0    1    2    1
##  [5,]    2    1    0    0
##  [6,]    0    2    0    0
##  [7,]    0    1    0    2
##  [8,]    2    0    0    1
```

```
##  [9,]    2    1    0    1
## [10,]    2    0    0    0
## [11,]    3    0    1    0
## [12,]    3    0    1    1
## [13,]    2    0    0    0
## [14,]    2    0    0    0
## [15,]    0    1    0    2
## [16,]    1    0    0    2
## [17,]    3    0    0    1
## [18,]    2    0    1    1
## [19,]    2    2    0    2
## [20,]    2    0    0    0
## [21,]    2    1    2    0
## [22,]    1    2    0    0
## [23,]    2    1    1    0
## [24,]    3    0    0    1
## [25,]    3    0    1    0
## [26,]    2    0    0    0
## [27,]    2    2    0    1
## [28,]    2    0    0    1
## [29,]    2    0    1    1
## [30,]    1    0    2    1
## [31,]    2    0    1    0
## [32,]    2    0    1    1
## [33,]    2    0    0    0
## [34,]    2    1    0    1
## [35,]    3    0    0    0
## [36,]    2    0    0    1
## [37,]    2    1    0    0
## [38,]    2    0    0    2
## [39,]    0    1    0    2
## [40,]    3    0    1    1
stopCluster(myCl)
```

Fitting through ngspatial package

Log QWI suggests a gaussian distribution. Preliminary analysis suggests a quarterly seasonality. Therefore, two different binary adjacency matrices with non-zero elements at different lags are tested here (heuristic from ACF/PACF). The logic behind the lags is an expanded form of SARIMA model (seasonal order of 1, and AR order of 1 each).

Since effectively we are dealing with a spatial analysis with more dimensions, the same methodology of `sparse-sglmm` function from `ngspatial` package (Hughes, J.) is used to fit the model with spatiotemporal eigenvector filters. The parameter estimation scheme differs from that of Grffith (2013) in that it employs a Bayesian MCMC. Specifically, a model is fit on the first 88 available quarters; then, for the most recent quarter (2017 Q2), a one-step forecast will be made and prediction errors will be examined. For reference purposes, a spatial-only model will be also fitted for the previous quarter (2017 Q1) and used for forecasting.

In a gaussian setting, the MCMC scheme does not require tuning (all updates are Gibbs updates). Prior for exogenous variable regression coefficient β is set to be $N(\mathbf{0}, 10^5\mathbf{I})$. Hyperprior for the common precision τ_δ for process error term δ is $\Gamma(0.01, 100)$.

```r
## Comparing three different approaches to adj. matrix.
set.seed(12345)
t <- 89 # 89 quarters
adjTime <- binAdjTime_fn(t, lags=c(1,2,4,5,6))
## Areal adjacency matrix for MO.
adjAreal <- A[indicesMO, indicesMO]
## space-time adjacency matrix can be specified in two ways: "lagged" and "contemporaneous".
adjLagged_fn <- function(At, As) {
  n <- nrow(As)
  At %x% (As + diag(1, n, n))
}
adjContemp_fn <- function(At, As) {
  nt <- nrow(At)
  ns <- nrow(As)
  (diag(1, nt, nt) %x% As) + (At %x% diag(1, ns, ns))
}

adjStLag <- adjLagged_fn(adjTime, adjAreal)
adjStCon <- adjContemp_fn(adjTime, adjAreal)

# Model fitting: ngspatial
## Since male and female groups share adjacency matrices, more efficient coding can be possible
## thru first eigendecomposition then using internal fit methods.
## (eigendecomposition takes up considerable time)
hyper <- list(sigma.b=10^5)
modelShorthand_fn <- function(data, adjMat, hyper) {
  sparse.sglmm(logInc ~ 1 + index, gaussian, data, A=adjMat, method="RSR", x=T, y=T, attractive=100, hy
}
myCl <- makeCluster(detectCores()-1, type="FORK")
modelGSTList <- clusterMap(myCl, modelShorthand_fn,
                           list(dataMaleMO, dataFemMO, dataMaleMO, dataFemMO),
                           list(adjStLag, adjStLag, adjStCon, adjStCon),
                           list(hyper, hyper, hyper, hyper))
# modelMaleLagGST <- modelShorthand_fn(dataMaleMO, adjStLag, list(sigma.b=sigma.b))
# modelFemLagGST <- modelShorthand_fn(dataFemMO, adjStLag, list(sigma.b=sigma.b))
# modelMaleConGST <- modelShorthand_fn(dataMaleMO, adjStCon, list(sigma.b=sigma.b))
# modelFemConGST <- modelShorthand_fn(dataFemMO, adjStCon, list(sigma.b=sigma.b))
stopCluster(myCl)
```

We can use the model to forecast log average monthly income for the validation set (the most recent available quarter: 2017 Q2). Note that this model does not incorporate observation error terms (in the spirit of Cressie and Wikle's hierarchical model), mainly because state-level imputatino variances are not known for recent quarters and because of parameter identifiability issues. Model comparison criterion will be standard MSE (alternatively called SPE).

```r
# Visualization
# makeshift function for map visualizations.
c_makeDataMO_fn <- function(sglm, subregion, time) {
  ## formats a data frame for visualizing on MO state map
  data <- cbind.data.frame(
    subregion = subregion, time = time, Observed = sglm$y, Predicted = sglm$fitted.values, Residuals = s
  )
  right_join(map_data("county", "missouri"), data, by="subregion")
}
```

```
c_plotDataMO_fn <- function(df, time, fill, title=NULL) {
  ## df is a result from above function
  ## time controls which quarter to plot
  df_f <- filter(df, time==time)
  result <- ggplot(data=df_f, aes(long, lat)) +
    geom_polygon(aes_string(group="group", fill=fill)) +
    labs(title=title) +
    scale_fill_distiller(palette="YlGnBu", direction=1)
  result
}

plotDataList <- lapply(modelGSTList, function(x)
  c_makeDataMO_fn(x, subregion=rep(subregionsMO,t),
                  time = dataMaleMO$time[dataMaleMO$index <= t]))

plotFitted_l <- lapply(plotDataList, function(x)
  c_plotDataMO_fn(x, time="2017Q2", fill="Observed"))
plotFitted_l <- lapply(plotDataList, function(x)
  c_plotDataMO_fn(x, time="2017Q2", fill="Predicted"))
plotResid_l <- lapply(plotDataList, function(x)
  c_plotDataMO_fn(x, time="2017Q2", fill="Residuals"))

plotSummary_l <- lapply(plotDataList, function(x)
  list(beta = x$coefficients, beta.se = x$beta.mcse,
       tau.s = x$tau.s.est, tau.s.se = x$tau.s.mcse,
       gamma = x$gamma.est, gamma.se = x$gamma.mcse,
       tau.h = x$tau.h.est, dic = x$dic))
```

The produced images and coefficient summaries are included in the `modelPlots` directory.

```
modelMaleLagGST2$coefficients
modelMaleLagGST2$beta.mcse
modelMaleLagGST2$tau.s.est
modelMaleLagGST2$tau.s.mcse
modelMaleLagGST2$gamma.est
modelMaleLagGST2$gamma.mcse
modelMaleLagGST2$dic

modelMaleConGST2$coefficients
modelMaleConGST2$beta.mcse
modelMaleConGST2$tau.s.est
modelMaleConGST2$tau.s.mcse
modelMaleConGST2$gamma.est
modelMaleConGST2$gamma.mcse
modelMaleConGST2$dic

modelFemLagGST2$coefficients
modelFemLagGST2$beta.mcse
modelFemLagGST2$tau.s.est
modelFemLagGST2$tau.s.mcse
modelFemLagGST2$gamma.est
modelFemLagGST2$gamma.mcse
modelFemLagGST2$dic
```

```
modelFemConGST2$coefficients
modelFemConGST2$beta.mcse
modelFemConGST2$tau.s.est
modelFemConGST2$tau.s.mcse
modelFemConGST2$gamma.est
modelFemConGST2$gamma.mcse
modelFemConGST2$dic
```