gps data mapping

Barbara E. Mottey

2/17/2021

```r
dat <- readRDS('C:/Users/sympl/Documents/UMass/msthesis/Data/completedata.rds')

to_factors <- c("fuel_bin", "gender", "residence", "wealth", "education", "marital_s", "region")
dat %<>% mutate_at(to_factors, funs(factor(.)))
```

```
## Warning: `funs()` is deprecated as of dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

LMER in INLA

```r
library(INLA)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Loading required package: parallel
```

```
## Loading required package: foreach
```

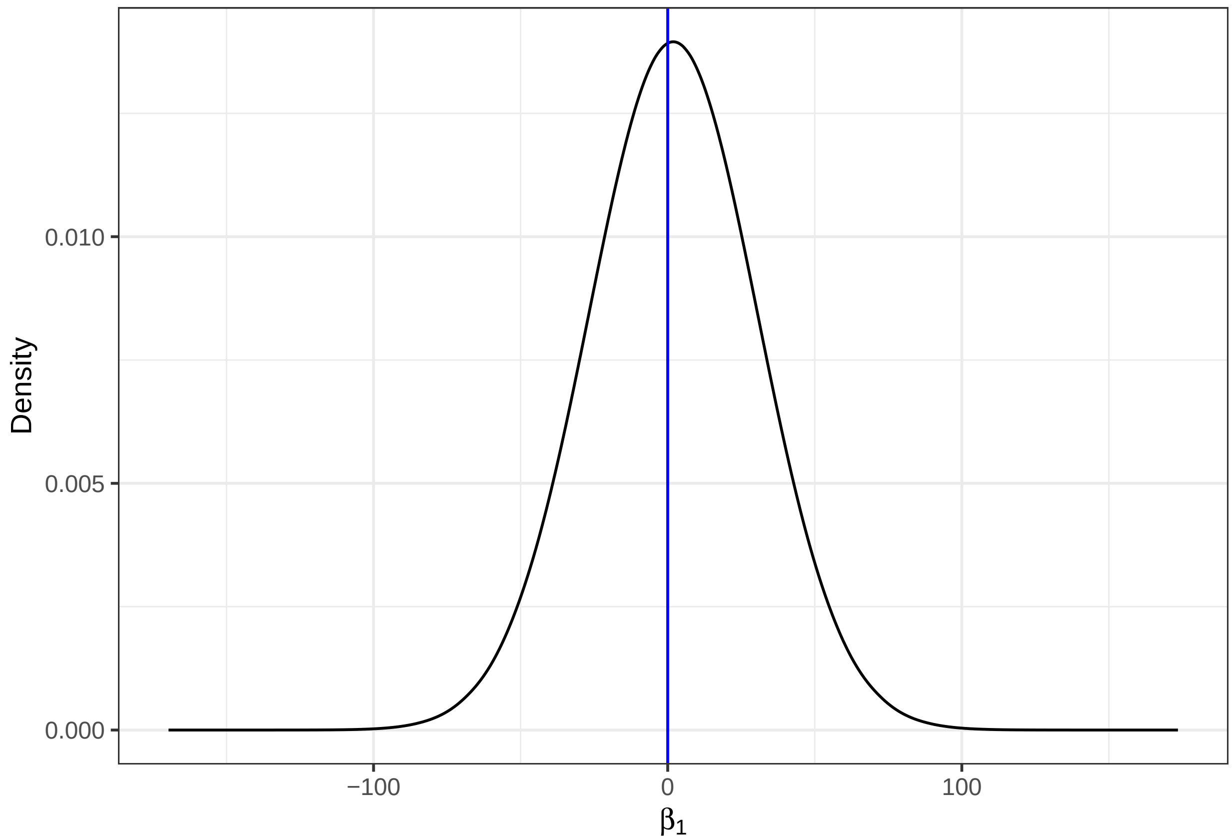
```
##
## Attaching package: 'foreach'
```

```
## The following objects are masked from 'package:purrr':
##
##     accumulate, when

## This is INLA_21.01.26 built 2021-01-26 11:21:34 UTC.
##  - See www.r-inla.org/contact-us for how to get help.
##  - Save 379.8Mb of storage running 'inla.prune()'
```

```
formula = c_weight~1+fuel_bin+w_age+bmi+gender+residence+wealth+education+marital_s+ f(region, model =
result<-inla(formula, family = "gaussian", data=dat, control.predictor = list(compute = TRUE))

summary(result)
```

```
##
## Call:
##    "inla(formula = formula, family = \"gaussian\", data = dat,
##    control.predictor = list(compute = TRUE))"
## Time used:
##     Pre = 4.38, Running = 15.9, Post = 1.94, Total = 22.2
## Fixed effects:
##                 mean      sd 0.025quant 0.5quant 0.975quant     mode kld
## (Intercept) 2994.148 126.412   2745.960 2994.145   3242.130 2994.148   0
## fuel_bin1      1.882  28.597    -54.264    1.881     57.981    1.882   0
## w_age          5.601   3.576     -1.420    5.601     12.617    5.601   0
## bmi            1.485   1.926     -2.295    1.485      5.263    1.485   0
## gender2      -46.128  26.280    -97.724  -46.128      5.426  -46.128   0
## residence2    15.802  26.772    -36.761   15.801     68.320   15.802   0
## wealth2        6.606  28.530    -49.409    6.605     62.574    6.606   0
## wealth3       13.851  28.131    -41.380   13.850     69.036   13.851   0
## wealth4       -8.812  27.855    -63.501   -8.813     45.832   -8.812   0
## wealth5        1.919  28.123    -53.296    1.919     57.089    1.919   0
## education1    -9.664  28.209    -65.047   -9.664     45.673   -9.664   0
## education2     0.546  26.680    -51.835    0.545     52.883    0.546   0
## education3     1.321  30.530    -58.620    1.320     61.212    1.321   0
## marital_s1    10.667  29.067    -46.401   10.666     67.686   10.667   0
##
## Random effects:
##   Name     Model
##     region IID model
##
## Model hyperparameters:
##                                              mean      sd 0.025quant  0.5quant
## Precision for the Gaussian observations      0.00    0.00       0.00      0.00
## Precision for region                    852364.70 1327.62  850056.28 852192.26
##                                         0.975quant      mode
## Precision for the Gaussian observations       0.00      0.00
## Precision for region                     855440.41 851593.50
##
## Expected number of effective parameters(stdev): 5.29(0.00)
## Number of equivalent replicates : 559.52
##
## Marginal log-Likelihood:  -24438.84
## Posterior marginals for the linear predictor and
##  the fitted values are computed
```

```r
## plot of posterior distribution for b1
marginal <- inla.smarginal(result$marginals.fixed$fuel_bin1) #posterior distn of b1 is stored here
marginal <- data.frame(marginal)
ggplot(marginal, aes(x = x, y = y)) + geom_line() + labs(x = expression(beta[1]), y = "Density") +
  geom_vline(xintercept = 0, col = "blue") + theme_bw()
```




```r
#2014 contains 1310 entries
d <- dat %>% filter(year_cmc=="2014") %>% group_by( cluster.no) %>%
  summarize(meanbw = mean(c_weight)) #394 unique clusters
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

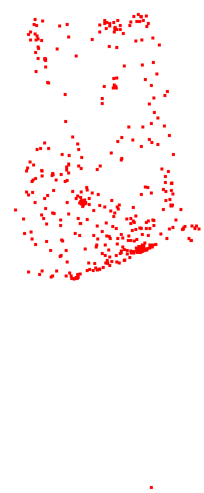
```r
#head(d)
#newdata %>% filter(year_cmc=="2014") %>% group_by(cluster.no) %>% summarise(meanbw = mean(c_weight)) #
#unique(d$cluster.no)

#table(dat$fuel_bin[dat$fuel_bin == 0])
```

##GPS coordinates

```r
#gc<-read.csv("GPS/GHGC72FL.csv")


ge.shp<-readOGR("C:/Users/sympl/Documents/UMass/msthesis/GPS/GHGE71FL/GHGE71FL.shp")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "C:\Users\sympl\Documents\UMass\msthesis\GPS\GHGE71FL\GHGE71FL.shp", layer: "GHGE71FL"
## with 427 features
## It has 20 fields
```

```
gt<-ge.shp@data

#summary(gt$ALT_GPS)

plot(ge.shp, pch=".", col="red")
```



#merge with geo data

```
#ge.shp@data$meanbw <- extract(d, ge.shp@data[, c("DHSCLUST")])
```