Bayesian Small Area Estimation of Child Mortality using Survey Data

A Case Study of U5MR in Kenya

Zehang Richard Li

2019/09/18

The `SUMMER` package offers a class of tools for small-area estimation with survey data in space and time. Such data are usually collected with complex stratified designs, which must be acknowledge in the analysis. In this vignette, we use two examples to illustrate spatial and spatial-temporal smoothing of child mortality rates. The first example uses real data from Kenya 2014 DHS survey, which requires separate authorization with DHS program to obtain the data. The second example uses a simulated DHS model data that is available as part of the package.

The estimation of subnational under-five mortality rates (U5MR) is the main purpose of this package, although similar analysis could be carried out for other quantities of interest such as neonate mortality rates.

Pre-processing the data

First, we load the package and the necessary data. INLA is not in a standard repository, so we check if it is available and install it if it is not installed. For this vignette, we used INLA version 19.09.03.

```
library(SUMMER)
if (!isTRUE(requireNamespace("INLA", quietly = TRUE))) {
  install.packages("INLA", repos=c(getOption("repos"),
                   INLA="https://inla.r-inla-download.org/R/stable"), dep=TRUE)
}
```

The DHS data can be obtained from the DHS program website at https://dhsprogram.com/data/dataset/Kenya_Standard-DHS_2014. For the analysis of U5MR, we will use the Births Recode in .stata format. Notice that registration with the DHS program is required in order to access this dataset. The map files for this DHS can be freely downloaded from http://spatialdata.dhsprogram.com/boundaries/.

With both the DHS birth record data and the corresponding shapefiles saved in the local directory. We can load them into `R` with packages `readstata13` and `rgdal`. We also automatically generates the spatial adjacency matrix `Amat` using the function `getAmat()`.

```
library(readstata13)
filename <- "../Demo/KEBR71DT/KEBR71FL.DTA"
births <- read.dta13(filename, generate.factors = TRUE)

library(rgdal)
mapfilename <- "../Demo/shps/sdr_subnational_boundaries.shp"
geo <- readOGR(mapfilename, verbose = FALSE)
Amat <- getAmat(geo, geo$REGNAME)
Amat
```

```
##               central coast eastern nairobi north eastern nyanza
## central             0     0       1       1             0      0
## coast               0     0       1       0             1      0
## eastern             1     1       0       1             1      0
## nairobi             1     0       1       0             0      0
```

```
## north eastern       0     1       1       0             0      0
## nyanza              0     0       0       0             0      0
## rift valley         1     1       1       1             0      1
## western             0     0       0       0             0      1
##               rift valley western
## central                  1       0
## coast                    1       0
## eastern                  1       0
## nairobi                  1       0
## north eastern            0       0
## nyanza                   1       1
## rift valley              0       1
## western                  1       0
```

The `Amat` matrix encodes the spatial adjacency matrix of the 8 Admin-1 region groups, with column and row names matching the regions used in the map. This adjacency matrix will be used for the spatial smoothing model. It can also be created by hand if necessary.

Bayesian space-time smoothing of direct estimates

Prepare person-month data

We first demonstrate the method that smooths the direct estimates of subnational-level U5MR. For this analysis, we consider the 8 Admin-1 region groups. In order to calculate the direct estimates of U5MR, we need the full birth history data in the format so that every row corresponds to a birth and columns that contain:

- Indicators corresponding to survey design, e.g., strata (`v023`), cluster (`v001`), and household (`v002`)
- Survey weight (`v025`)
- Date of interview in century month codes (CMC) format, i.e., the number of the month since the beginning of 1990 (`v008`)
- Date of child's birth in CMC format (`b3`)
- Indicator for death of child (`b5`)
- Age of death of child in months (`b7`)

The birth history data from DHS is already in this form and the `getBirths` function default to use the current recode manual column names (as indicated above). The name of these fields can be defined explicitly in the function arguments too. We reorganize the data into the 'person-month' format with `getBirths` function and reorder the columns for better readability.

```
dat <- getBirths(data = births, surveyyear = 2014, strata = c("v023"),
    year.cut = seq(1985, 2020, by = 5))
dat <- dat[, c("v001", "v002", "v024", "time", "age", "v005", "strata",
    "died")]
colnames(dat) <- c("clustid", "id", "region", "time", "age", "weights",
    "strata", "died")
head(dat)
```

```
##   clustid id  region  time  age weights strata died
## 1       1  6 nairobi 05-09    0 5476381      1    0
## 2       1  6 nairobi 05-09 1-11 5476381      1    0
## 3       1  6 nairobi 05-09 1-11 5476381      1    0
## 4       1  6 nairobi 05-09 1-11 5476381      1    0
## 5       1  6 nairobi 05-09 1-11 5476381      1    0
## 6       1  6 nairobi 05-09 1-11 5476381      1    0
```

Notice that we also need to specify the time intervals of interest. In this example, we with to calculate and predict U5MR in 5-year intervals from 1985-1990 to 2015-2019. For U5MR, we will use the discrete survival model to calculate direct estimates for each region and time. This step involves breaking down the age of each death into discrete intervals. The default option assumes a discrete survival model with six discrete hazards (probabilities of dying in a particular interval, given survival to the start of the interval) for each of the age bands: $[0, 1), [1, 12), [12, 24), [24, 36), [36, 48)$, and $[48, 60]$.

We may also calculate other types of mortality rates of interest using `getBirths`. For example, for U1MR,

```
dat_infant <- getBirths(data = births, surveyyear = 2014, month.cut = c(1,
    12), strata = c("v023"))
```

And the smoothing steps can be similarly carried out.

Horvitz-Thompson estimators of U5MR

Using the person-month format data, we can calculate Horvitz-Thompson estimators using `getDirect` for a single survey or `getDirectList` for multiple surveys. The discrete hazards in each time interval are estimated using a logistic regression model, with weighting to account for the survey design. The direct estimates are then calculated using the discrete hazards. In order to correctly account for survey design, We need to specify the stratification and cluster variables. In the Kenya DHS example, a two-stage stratified cluster sampling design was used, where strata are specified in the `strata` column, and clusters are specified by the cluster ID (`clusterid`) and household ID (`id`).

```
years <- levels(dat$time)
direct0 <- getDirect(births = dat, years = years, regionVar = "region",
    timeVar = "time", clusterVar = "~clustid+id", ageVar = "age", weightsVar = "weights",
    geo.recode = NULL)
head(direct0)
```

```
##   region years  mean lower upper logit.est var.est region_num survey
## 1    All 85-89 0.086 0.073 0.100      -2.4  0.0074          0     NA
## 2    All 90-94 0.098 0.089 0.109      -2.2  0.0032          0     NA
## 3    All 95-99 0.091 0.084 0.099      -2.3  0.0023          0     NA
## 4    All 00-04 0.079 0.073 0.085      -2.5  0.0016          0     NA
## 5    All 05-09 0.060 0.055 0.065      -2.8  0.0019          0     NA
## 6    All 10-14 0.052 0.048 0.057      -2.9  0.0023          0     NA
##   logit.prec
## 1        135
## 2        316
## 3        444
## 4        614
## 5        526
## 6        433
```

Adjustments using external information

Sometimes additional information are available to adjust the direct estimates from the surveys. For example, in countries with high prevalence of HIV, estimates of U5MR can be biased, particularly before ART treatment became widely available. Pre-treatment HIV positive women had a high risk of dying, and such women who had given birth were therefore less likely to appear in surveys. The children of HIV positive women are also more likely to have a higher probability of dying compared to those born to HIV negative women. Thus we expect that the U5MR is underestimated if we do not adjust for the missing women.

Suppose we can obtain the ratio of the reported U5MR to the true U5MR, r_{it}, at region i and time period t, we can apply the adjustment factor to the direct estimates and the associated variances. The HIV adjustment factors were calculated for the 2014 Kenya DHS survey and included in the package.

```
data(KenData)
direct <- getAdjusted(data = direct0, ratio = KenData$HIV2014)
```

National estimates of U5MR

The direct estimates calculated using `getDirect` contains both national estimates and subnational estimates for the 8 regions, over the 6 time periods and the projection period 2015-2019. We first fit a model with temporal random effects only to smooth the national estimates over time. In this part, we use the subset of data region variable being "All". We can fit a Random Walk 2 only model defined on the 5-year period.

```
fit1 <- fitINLA(data = direct, geo = NULL, Amat = NULL, year_label = years,
    year_range = c(1985, 2019), rw = 2, is.yearly = FALSE, m = 5)
```

We can also estimate the Random Walk 2 random effects on the yearly scale.

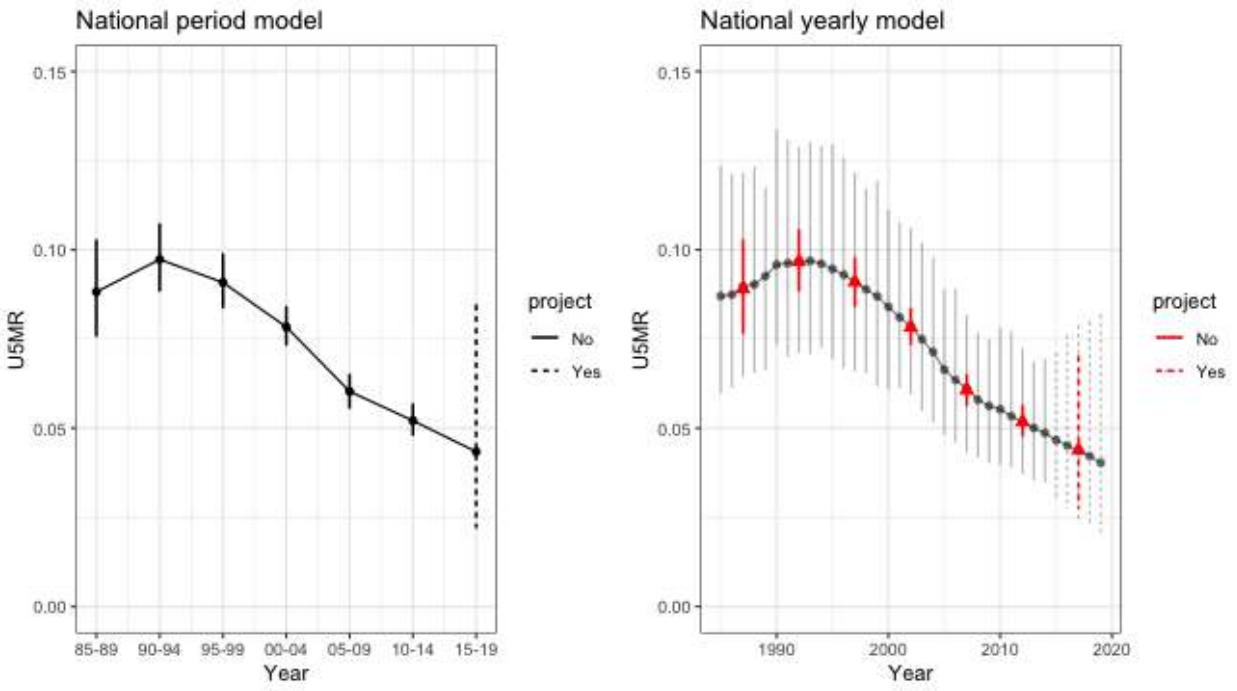
```
fit2 <- fitINLA(data = direct, geo = NULL, Amat = NULL, year_label = years,
    year_range = c(1985, 2019), rw = 2, is.yearly = TRUE, m = 5)
```

The marginal posteriors are already stored in the fitted object. We use the following function to extract and re-arrange them.

```
out1 <- getSmoothed(fit1, year_range = c(1985, 2019), year_label = years)
out2 <- getSmoothed(fit2, year_range = c(1985, 2019), year_label = years)
```

We can compare the results visually. Notice to correctly display the period estimates, the reference year in each period needs to be specified. Here we simply take the median year in each period.

```
library(ggplot2)
library(gridExtra)
years.ref <- c(1987, 1992, 1997, 2002, 2007, 2012, 2017)
g1 <- plot(out1, year_label = years, year_med = years.ref, is.subnational = FALSE) +
    ggtitle("National period model") + ylim(c(0, 0.15))
g2 <- plot(out2, is.subnational = FALSE, year_label = years, year_med = years.ref) +
    ggtitle("National yearly model") + ylim(c(0, 0.15))
grid.arrange(g1, g2, ncol = 2)
```



The national model also allows us to benchmark the estimates using other published national results. For example, we take the 2019 UN-IGME estimates and find the median under-5 mortality rates in each periods. We can then calculate the ratio of the estimates from national models to the published UN estimates.

```
data(KenData)
UN <- KenData$IGME2019
UN.period <- data.frame(period = c("85-89", "90-94", "95-99", "00-04",
    "05-09", "10-14"), median = NA)
for (i in 1:6) {
    UN.period$median[i] <- median(UN$mean[which(UN$years %in% (c(1985:1989) +
        (i - 1) * 5))])
}
ratio <- subset(out1, region == "All")$median[1:6]/UN.period$median
print(ratio)
```

```
## [1] 0.90 0.86 0.78 0.82 0.87 1.00
```

We will use this adjustment ratio to correct the bias from our direct estimates. We organize the adjustment ratios into a matrix of two columns, since the adjustment factor only varies over time. We can then perform the benchmarking to the UN estimates similar to the HIV adjustment before.

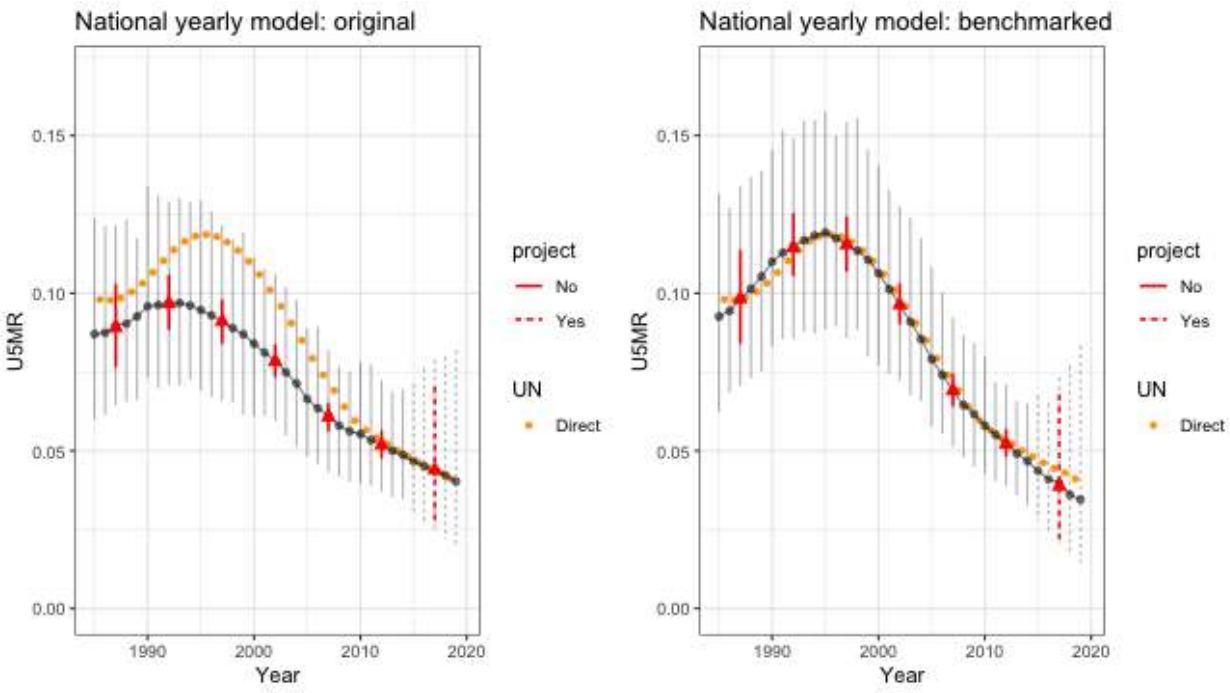
```
benchmark <- data.frame(time = c("85-89", "90-94", "95-99", "00-04", "05-09",
    "10-14"), ratio = ratio)
direct <- getAdjusted(data = direct, ratio = benchmark)
```

After benchmarking, we can fit the smoothing model again on the adjusted direct estimates, and see if they align with the UN estimates.

```
fit2.benchmark <- fitINLA(data = direct, geo = NULL, Amat = NULL, year_label = years,
    year_range = c(1985, 2019), rw = 2, is.yearly = TRUE, m = 5)
out2.benchmark <- getSmoothed(fit2.benchmark, year_range = c(1985, 2019),
    year_label = years)

g1 <- plot(out2, year_label = years, year_med = years.ref, is.subnational = FALSE,
```

```
    data.add = UN, option.add = list(point = "mean"), label.add = "UN",
    color.add = "orange") + ggtitle("National yearly model: original") +
    ylim(c(0, 0.17))
g2 <- plot(out2.benchmark, year_label = years, year_med = years.ref, is.subnational = FALSE,
    data.add = UN, option.add = list(point = "mean"), label.add = "UN",
    color.add = "orange") + ggtitle("National yearly model: benchmarked") +
    ylim(c(0, 0.17))
grid.arrange(g1, g2, ncol = 2)
```



Subnational estimates of U5MR

The syntax to fit subnational smoothing model is similar. Similar to the national model, we can choose to estimate temporal random effects on either yearly or period level. We can also choose the four types of space-time interaction terms using the `st.type` argument. The default hyper priors on the precision of random effects are now PC priors.

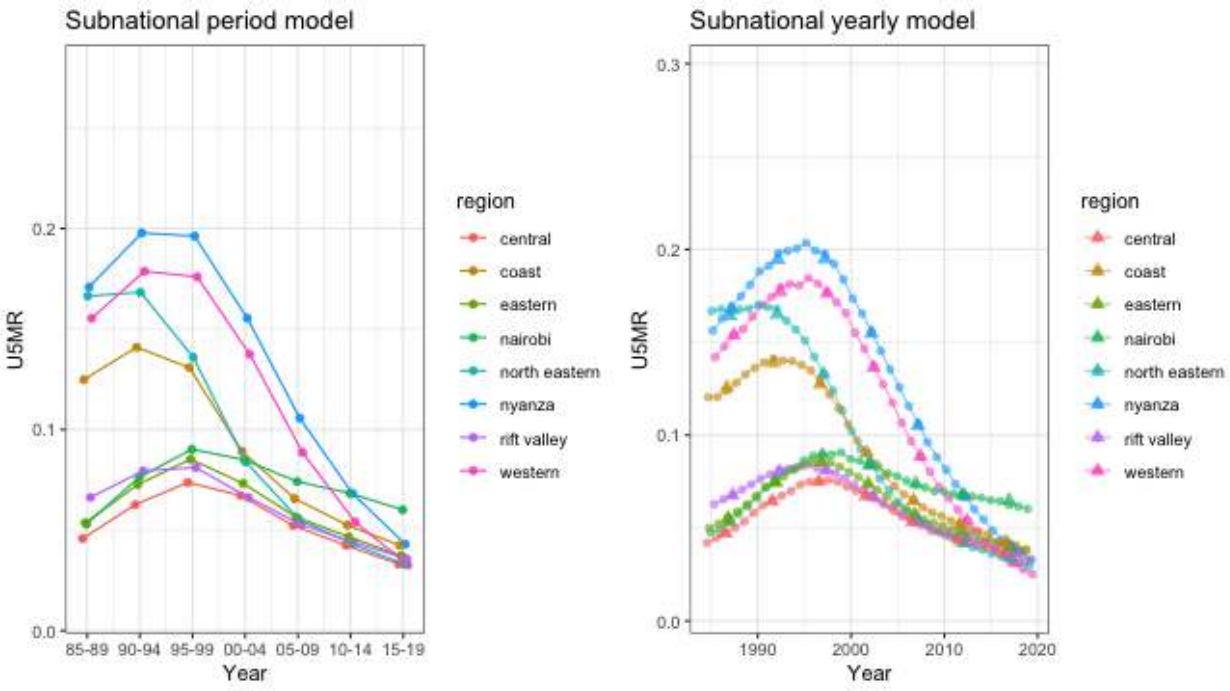
More details here

```
fit3 <- fitINLA(data = direct, geo = geo, Amat = Amat, year_label = years,
    year_range = c(1985, 2019), rw = 2, type.st = 4, is.yearly = FALSE)
out3 <- getSmoothed(fit3, Amat = Amat, year_range = c(1985, 2019), year_label = years)
```

Similarly we can also estimate the Random Walk 2 random effects on the yearly scale.

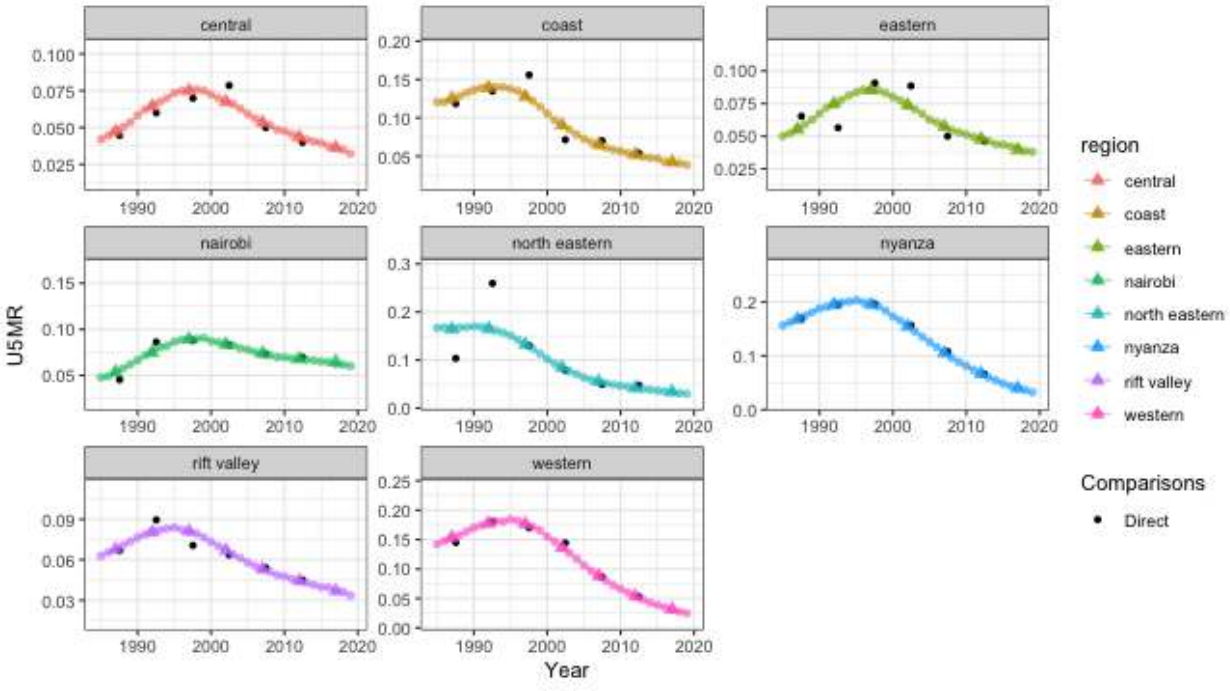
```
fit4 <- fitINLA(data = direct, geo = geo, Amat = Amat, year_label = years,
    year_range = c(1985, 2019), rw = 2, type.st = 4, is.yearly = TRUE)
out4 <- getSmoothed(fit4, Amat = Amat, year_range = c(1985, 2019), year_label = years)

g2 <- NULL
g2[[1]] <- plot(out3, is.subnational = TRUE) + ggtitle("Subnational period model")
g2[[2]] <- plot(out4, is.subnational = TRUE) + ggtitle("Subnational yearly model")
grid.arrange(grobs = g2, ncol = 2)
```



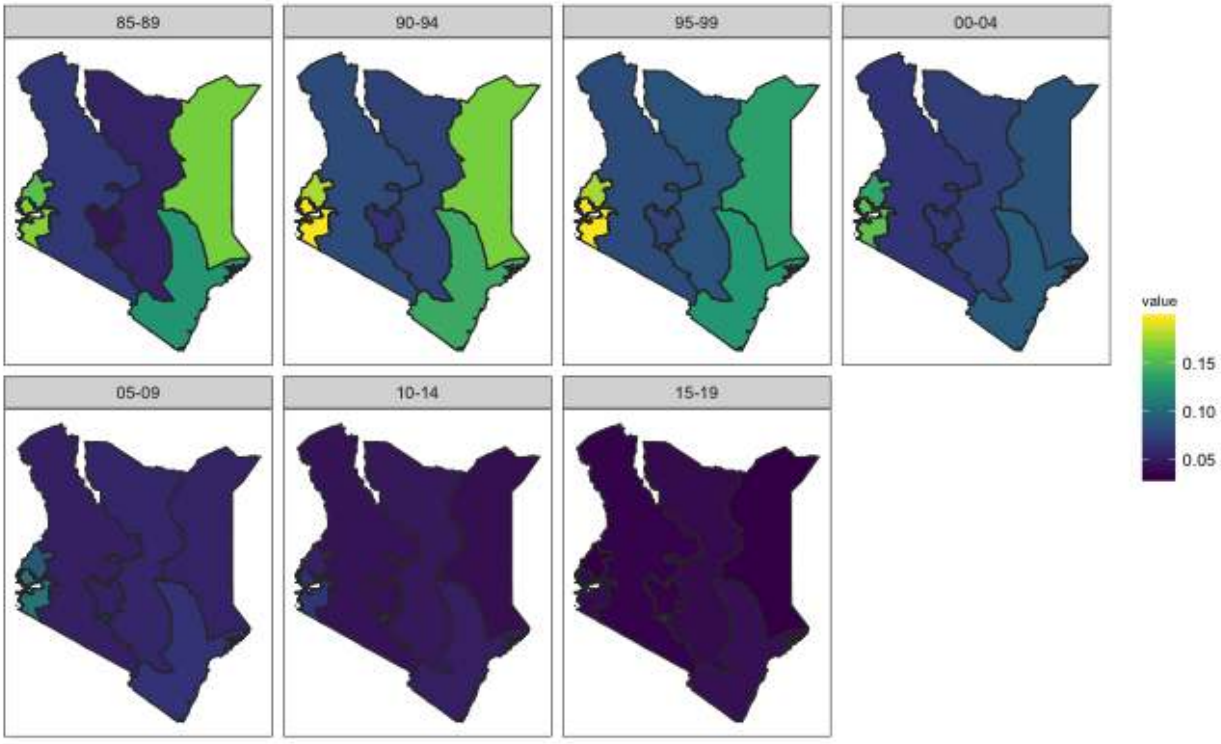
We can also add back the direct estimates for comparison.

```
plot(out4, is.subnational = TRUE, data.add = direct, option.add = list(point = "mean",
    by = "survey")) + facet_wrap(~region, scales = "free")
```



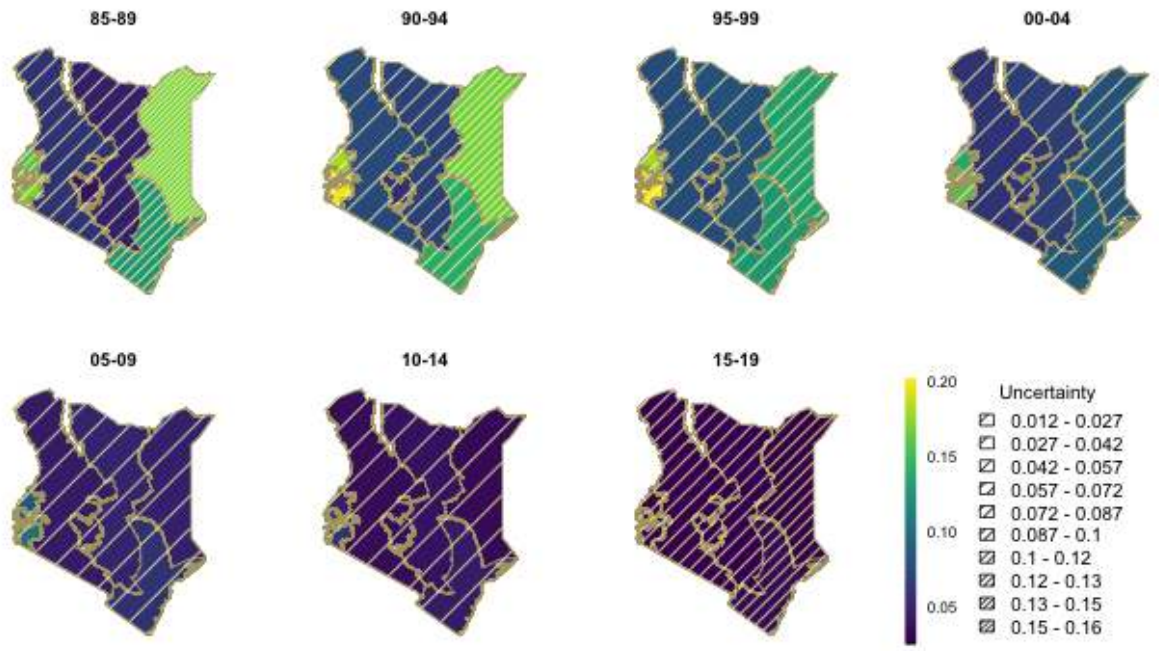
We can show the estimates over time on maps.

```
mapPlot(data = subset(out4, is.yearly == FALSE), geo = geo, variables = c("years"),
    values = c("median"), by.data = "region", by.geo = "REGNAME", is.long = TRUE,
    ncol = 4)
```



In order to also illustrate uncertainties of the estimates when presented on maps, we can use hatching to indicate the width of the 94% posterior credible intervals.

```
hatchPlot(data = subset(out4, is.yearly == FALSE), geo = geo, variables = c("years"),
    values = c("median"), by.data = "region", by.geo = "REGNAME", lower = "lower",
    upper = "upper", is.long = TRUE)
```



Comparing different models

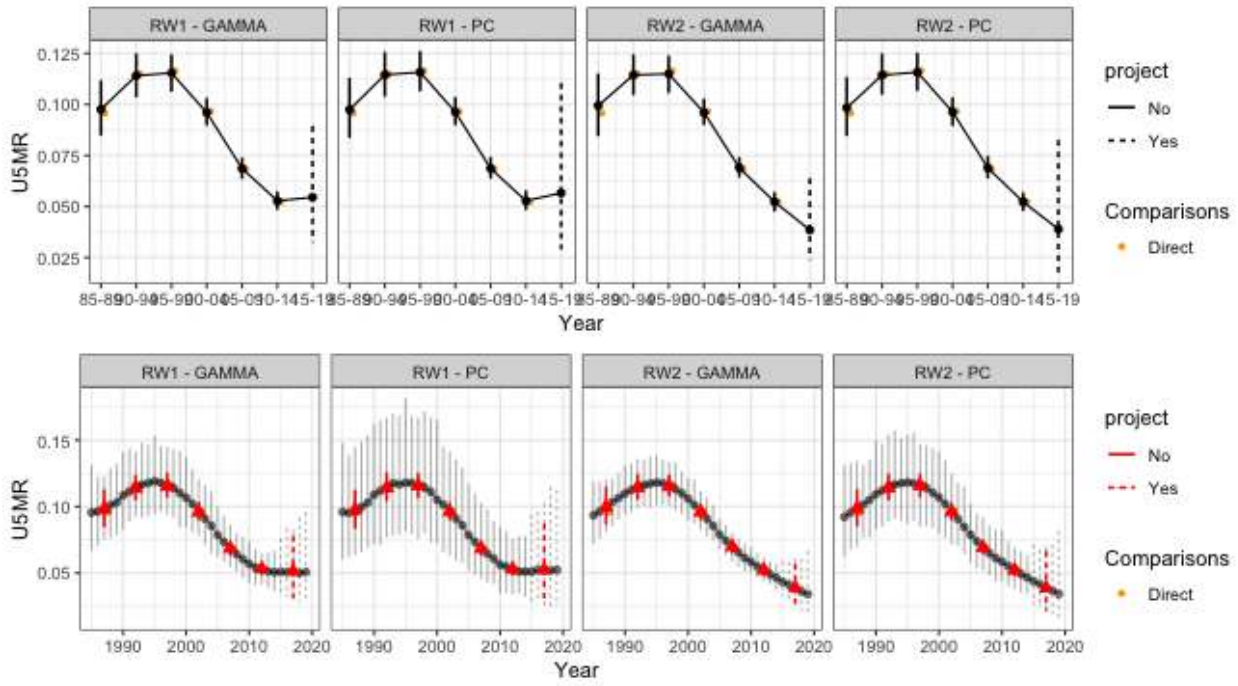
Since the year range and labels are the same as the default values, we omit them for better readability in the following comparisons of the national models.

```
index <- 1
flist <- NULL
projlist <- NULL
for (hyper in c("pc", "gamma")) {
    for (rw in c(1, 2)) {
        for (is.yearly in c(TRUE, FALSE)) {
            f <- fitINLA(data = direct, geo = NULL, Amat = NULL, year_label = years,
                rw = rw, is.yearly = is.yearly, m = 5, hyper = hyper)
            flist[[index]] <- f
            out <- getSmoothed(f)
            out$Model <- ifelse(is.yearly, "yearly", "period")
            out$prior <- paste0("RW", rw, " - ", toupper(hyper))
            projlist <- rbind(projlist, out)
            index <- index + 1
        }
    }
}
```

We compare the projections of the different models

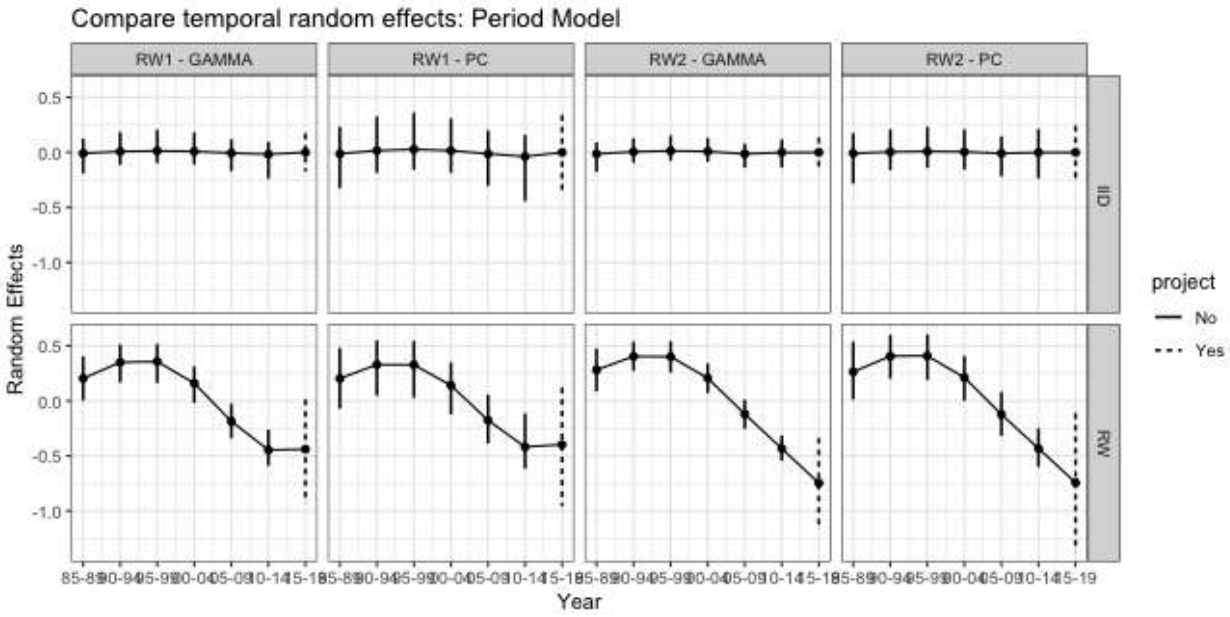
```
g1 <- plot(subset(projlist, Model == "period"), is.subnational = FALSE,
    data.add = subset(direct, region == "All"), option.add = list(point = "mean",
        by = "survey"), color.add = "orange") + facet_wrap(~prior, ncol = 4)
g2 <- plot(subset(projlist, Model == "yearly"), is.subnational = FALSE,
    data.add = subset(direct, region == "All"), option.add = list(point = "mean",
        by = "survey"), color.add = "orange") + facet_wrap(~prior, ncol = 4)
```

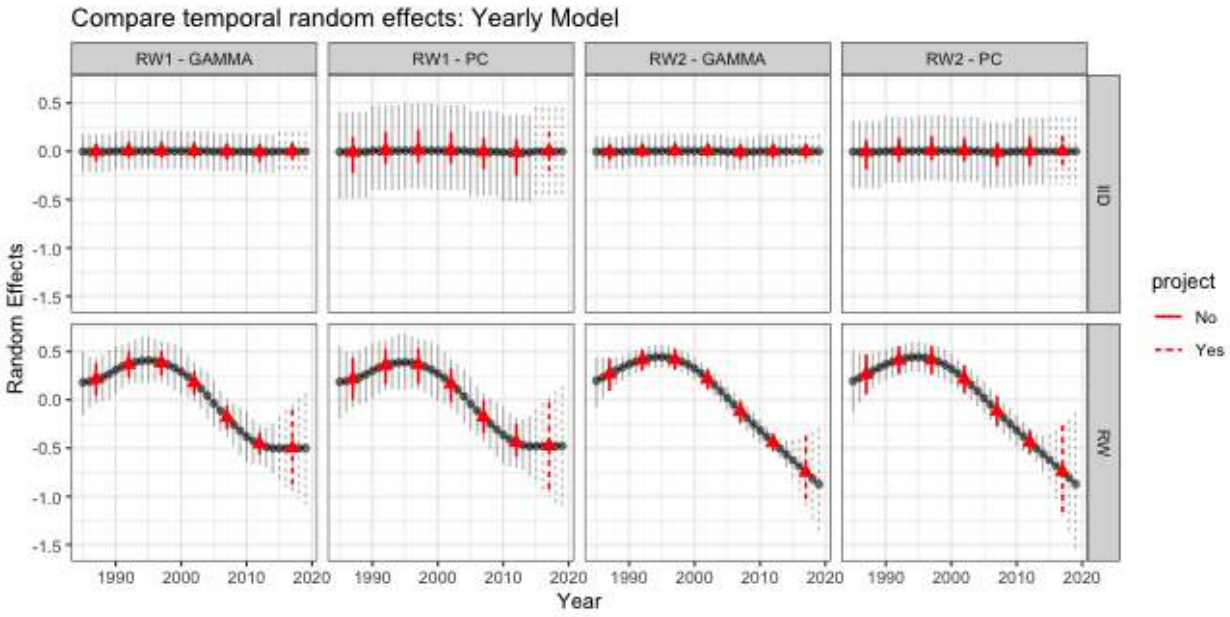
```
grid.arrange(g1, g2, ncol = 1)
```



We can also compare the specific model fits of the different models by extracting the posterior marginal distributions of the random effects.

```
random.time <- NULL
index <- 1
for (hyper in c("pc", "gamma")) {
    for (rw in c(1, 2)) {
        for (is.yearly in c(TRUE, FALSE)) {
            random <- getDiag(flist[[index]], field = "time")
            random$Model <- ifelse(is.yearly, "yearly", "period")
            random$prior <- paste0("RW", rw, " - ", toupper(hyper))
            random.time <- rbind(random.time, random)
            index <- index + 1
        }
    }
}
g1 <- plot(subset(random.time, Model == "period"), is.subnational = FALSE) +
    facet_grid(label ~ prior) + ggtitle("Compare temporal random effects: Period Model") +
    ylab("Random Effects")
g2 <- plot(subset(random.time, Model == "yearly"), is.subnational = FALSE) +
    facet_grid(label ~ prior) + ggtitle("Compare temporal random effects: Yearly Model") +
    ylab("Random Effects")
grid.arrange(g1, g2, ncol = 1)
```

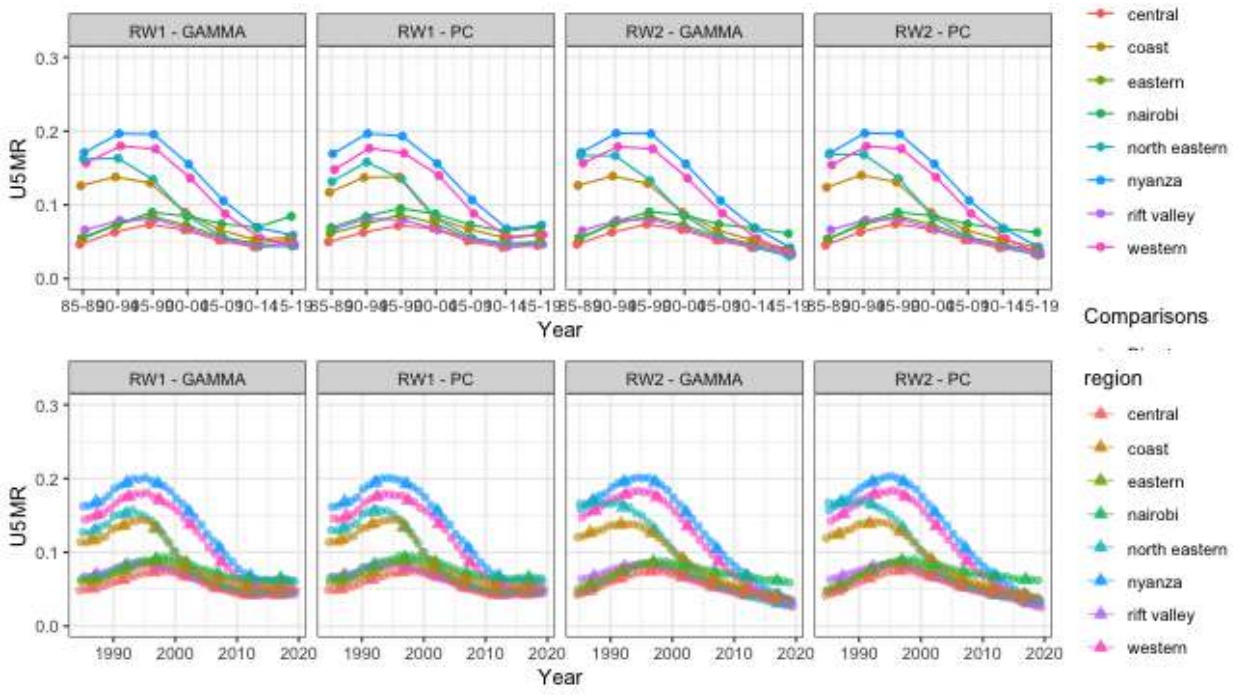




Similarly, we compare the estimates under different priors for the subnational models.

```
index <- 1
flist <- NULL
projlist <- NULL
for (hyper in c("pc", "gamma")) {
    for (rw in c(1, 2)) {
        for (is.yearly in c(TRUE, FALSE)) {
            f <- fitINLA(data = direct, geo = geo, Amat = Amat, year_label = years,
                rw = rw, is.yearly = is.yearly, m = 5, hyper = hyper, type.st = 4)
            flist[[index]] <- f
            out <- getSmoothed(f, Amat = Amat)
            out$Model <- ifelse(is.yearly, "yearly", "period")
            out$prior <- paste0("RW", rw, " - ", toupper(hyper))
            projlist <- rbind(projlist, out)
```

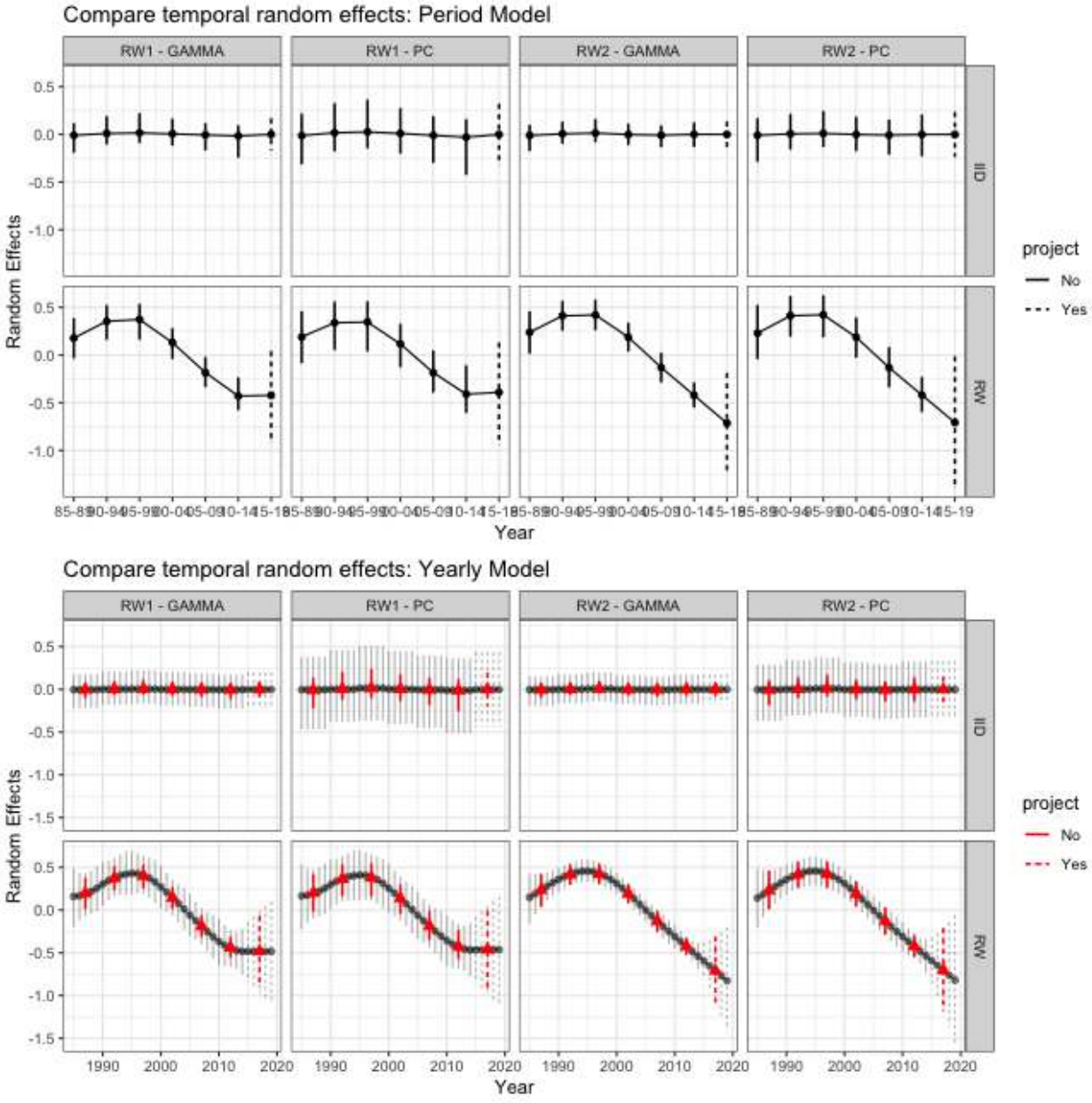
```
            index <- index + 1
        }
    }
}
g1 <- plot(subset(projlist, Model == "period"), is.subnational = TRUE,
    data.add = subset(direct, region == "All"), option.add = list(point = "mean",
        by = "survey"), color.add = "orange") + facet_wrap(~prior, ncol = 4) +
    ylim(c(0, 0.3))
g2 <- plot(subset(projlist, Model == "yearly"), is.subnational = TRUE) +
    facet_wrap(~prior, ncol = 4) + ylim(c(0, 0.3))
grid.arrange(g1, g2, ncol = 1)
```



We compare the posterior marginal distributions of the temporal random effects under different priors for the subnational models.

```
random.time <- NULL
index <- 1
for (hyper in c("pc", "gamma")) {
    for (rw in c(1, 2)) {
        for (is.yearly in c(TRUE, FALSE)) {
            random <- getDiag(flist[[index]], field = "time")
            random$Model <- ifelse(is.yearly, "yearly", "period")
            random$prior <- paste0("RW", rw, " - ", toupper(hyper))
            random.time <- rbind(random.time, random)
            index <- index + 1
        }
    }
}
g1 <- plot(subset(random.time, Model == "period"), is.subnational = FALSE) +
    facet_grid(label ~ prior) + ggtitle("Compare temporal random effects: Period Model") +
    ylab("Random Effects")
g2 <- plot(subset(random.time, Model == "yearly"), is.subnational = FALSE) +
```

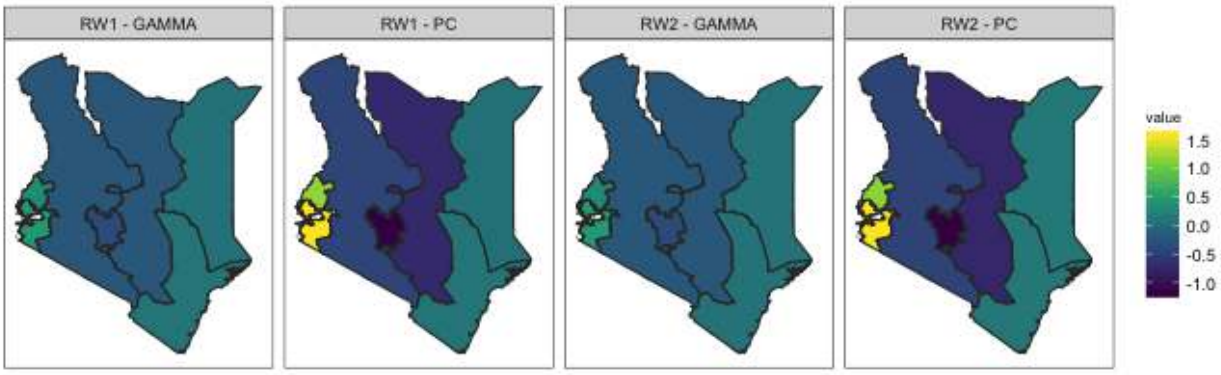
```
    facet_grid(label ~ prior) + ggtitle("Compare temporal random effects: Yearly Model") +
    ylab("Random Effects")
grid.arrange(g1, g2, ncol = 1)
```



We compare the posterior marginal distributions of the structured spatial random effects under different priors for the subnational models.

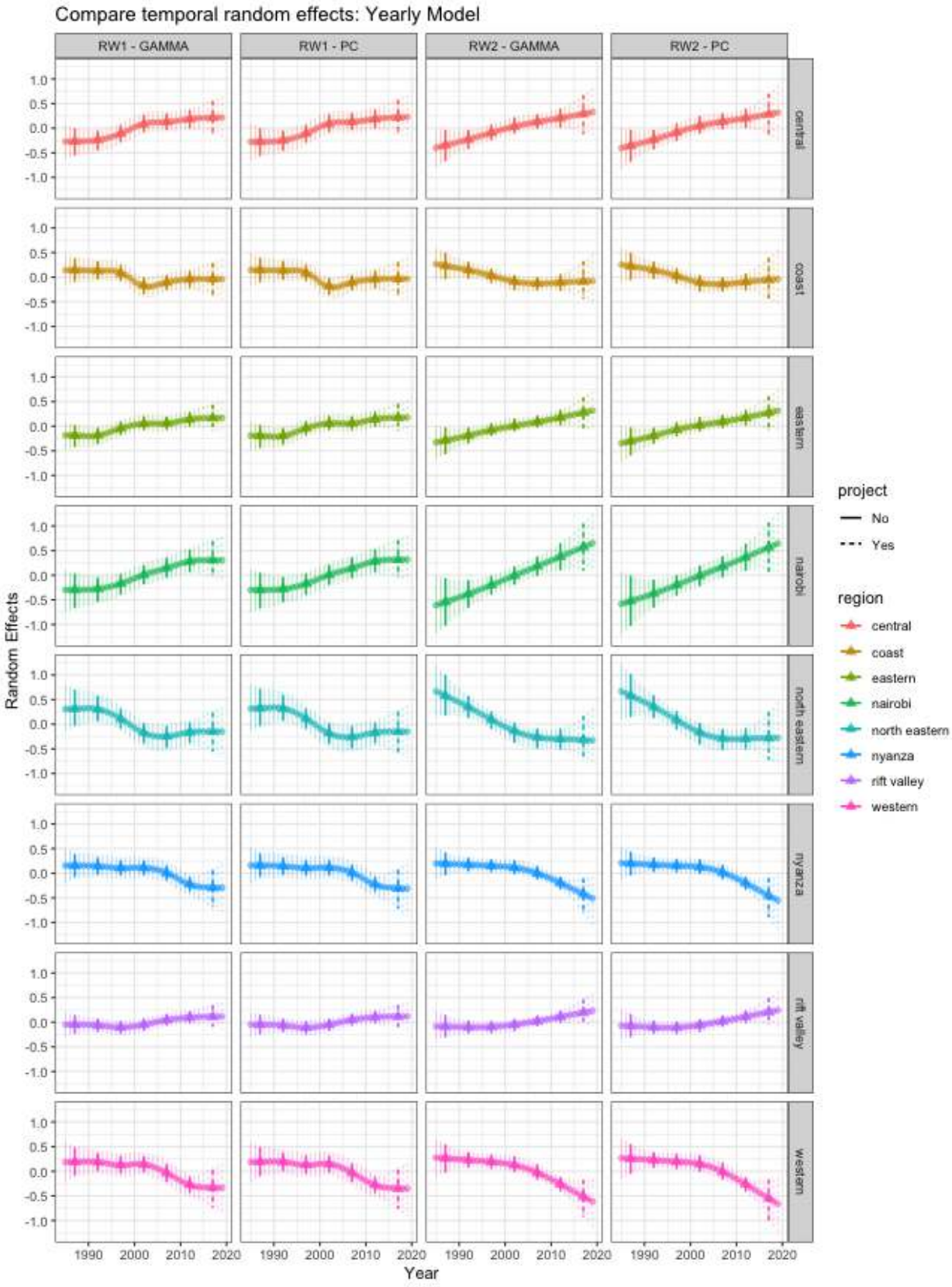
```
random.space <- NULL
index <- 1
for (hyper in c("pc", "gamma")) {
    for (rw in c(1, 2)) {
        for (is.yearly in c(TRUE, FALSE)) {
            random <- getDiag(flist[[index]], field = "space", Amat = Amat)
            random$Model <- ifelse(is.yearly, "yearly", "period")
```

```
            random$prior <- paste0("RW", rw, " - ", toupper(hyper))
            random.space <- rbind(random.space, random)
            index <- index + 1
        }
    }
}
mapPlot(subset(random.space, Model == "yearly" & label == "Besag"), geo = geo,
    variables = c("prior"), values = c("median"), by.data = "region", by.geo = "REGNAME",
    is.long = TRUE, ncol = 4)
```



We compare the posterior marginal distributions of the space-time interaction effects under different priors for the subnational models.

```
random.st <- NULL
index <- 1
for (hyper in c("pc", "gamma")) {
    for (rw in c(1, 2)) {
        for (is.yearly in c(TRUE, FALSE)) {
            random <- getDiag(flist[[index]], field = "spacetime", Amat = Amat)
            random$Model <- ifelse(is.yearly, "yearly", "period")
            random$prior <- paste0("RW", rw, " - ", toupper(hyper))
            random.st <- rbind(random.st, random)
            index <- index + 1
        }
    }
}
plot(subset(random.st, Model == "yearly"), is.subnational = TRUE, plot.CI = TRUE) +
    facet_grid(region ~ prior) + ggtitle("Compare temporal random effects: Yearly Model") +
    ylab("Random Effects")
```


Compare temporal random effects: Yearly Model
RW1 - GAMMA
RW1 - PC
RW2 - GAMMA
RW2 - PC
central
coast
eastern
nairobi
north eastern
nyanza
rift valley
western
Random Effects
Year
project
No
Yes
region
central
coast
eastern
nairobi
north eastern
nyanza
rift valley
western