2022 SISCER SAE: Unit-Level Models

Jon Wakefield and Peter Gao
Departments of Statistics and Biostatistics
University of Washington

2022-07-08

Unit level model

Areas are labeled by i and units by k.

Unit level model:

$$y_{ik} = \beta_0 + \mathbf{x}_{ik}^T \boldsymbol{\beta}_1 + \delta_i + \epsilon_{ik}, \quad \delta_i \sim_{iid} N(0, \sigma_\delta^2), \quad \epsilon_{ik} \sim_{iid} N(0, \sigma_\epsilon^2)$$

Here δ_i are area random effects and ϵ_{ik} are unit level errors.

The model assumes that conditional on $\mathbf{x}$ there is no sample selection bias.

Estimation proceeds by first estimating β_0 and the variance parameters σ_δ^2 and σ_ϵ^2.

Next, given known variance parameters, predict δ_i by calculating the EBLUP.

Nested error model of Battese et al (1988)

The area fitted values are:

$$\widehat{y}_i^{\text{EBLUP}} = \widehat{\beta}_0 + f_i \overline{y}_{iS} + (\overline{x}_i^T - f_i \overline{x}_{iS}^T)\widehat{\beta}_1 + (1 - f_i)\widehat{\delta}_i,$$

where

- $f_i = n_i/N_i$ is the domain sampling fraction.
- $\overline{y}_{iS}$ is the mean response in the sampled units.
- $\overline{x}_{iS}$ is the mean of the covariates in the sampled units.
- $\overline{x}_i$ is the mean of the covariates in the population.
- $\widehat{\delta}_i$ is the estimated random effect.

When $f_i \approx 0$,

$$\widehat{y}_i^{\text{EBLUP}} = \widehat{\beta}_0 + \overline{x}_i^T \widehat{\beta}_1 + \widehat{\delta}_i.$$

Corn and Soy Production

The `cornsoybean` and `cornsoybeanmeans` datasets contain info on corn and soy beans production in 12 Iowa counties. Code from Molina et al (2015) and data from Battese et al (1988). Ideally, we would like to use satellite imagery of the number of pixels assigned to corn and soy to estimate the hectares grown of corn.

- `SampSegments`: sample size.

- `PopnSegments`: population size.
- `MeanCornPixPerSeg`: county mean of the number of corn pixels (satellite imagery).
- `MeanSoyBeansPixPerSeg` county mean of the number of soy beans (satellite imagery) pixels.

So, `MeanCornPixPerSeg` and `MeanSoyBeansPixPerSeg` are the county means of the auxiliary variables.

Counties are the domains, and units are the segments.

```
library(sae)
data("cornsoybean")
head(cornsoybean, n = 10)
##    County CornHec SoyBeansHec CornPix SoyBeansPix
## 1       1  165.76        8.09     374          55
## 2       2   96.32      106.03     209         218
## 3       3   76.08      103.60     253         250
## 4       4  185.35        6.47     432          96
## 5       4  116.43       63.82     367         178
## 6       5  162.08       43.50     361         137
## 7       5  152.04       71.43     288         206
## 8       5  161.75       42.49     369         165
## 9       6   92.88      105.26     206         218
## 10      6  149.94       76.49     316         221
```

Auxiliary information

```
data("cornsoybeanmeans")
Xmean <- data.frame(cornsoybeanmeans[, c("CountyIndex", "MeanCornPixPerSeg",
    "MeanSoyBeansPixPerSeg")])
Popn <- data.frame(cornsoybeanmeans[, c("CountyIndex", "PopnSegments")])
head(Xmean)
##   CountyIndex MeanCornPixPerSeg MeanSoyBeansPixPerSeg
## 1           1            295.29                189.70
## 2           2            300.40                196.65
## 3           3            289.60                205.28
## 4           4            290.74                220.22
## 5           5            318.21                188.06
## 6           6            257.17                247.13
```

Fit nested error model (Battese-Harter-Fuller)

The `pbmseBHF` function:

- obtains EBLUPs under the nested error model and then
- uses a parametric bootstrap approach to estimate MSEs (a measure of the uncertainty - can think of as similar to the square of the standard error).

```
cornsoybean <- cornsoybean[-33, ]  # remove outlier
BHF <- pbmseBHF(CornHec ~ CornPix + SoyBeansPix, dom = County,
    meanxpop = Xmean, popnsize = Popn, B = 200, data = cornsoybean)
```

Results:

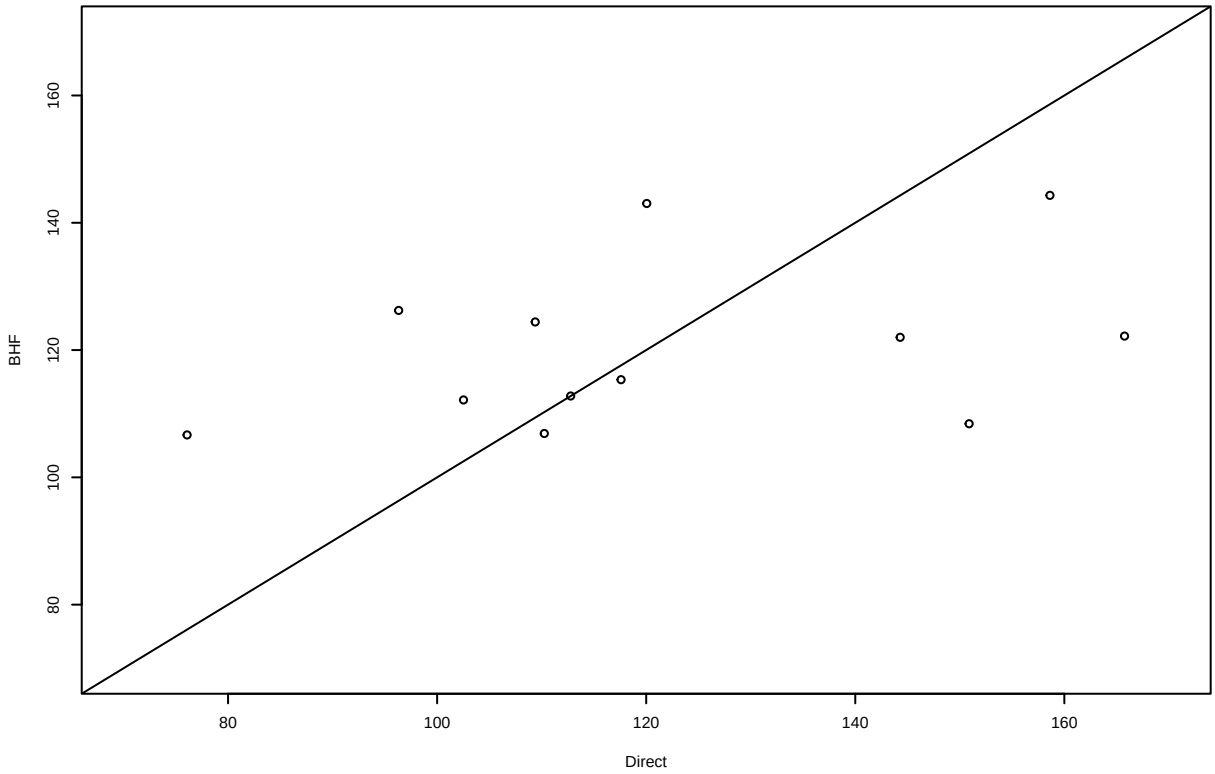
```
BHF$est$fit$summary
## Linear mixed model fit by REML ['lmerMod']
```

```
## Formula: ys ~ -1 + Xs + (1 | dom)
##
## REML criterion at convergence: 298.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.87577 -0.70965 -0.08544  0.72472  1.65661
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  dom      (Intercept) 140.0    11.83
##  Residual             147.3    12.14
## Number of obs: 36, groups:  dom, 12
##
## Fixed effects:
##                 Estimate Std. Error t value
## XsXs(Intercept) 51.07040   24.40970   2.092
## XsXsCornPix      0.32872    0.04988   6.591
## XsXsSoyBeansPix -0.13457    0.05519  -2.438
##
## Correlation of Fixed Effects:
##             XsX(I) XsXsCP
## XsXsCornPix -0.935
## XsXsSyBnsPx -0.892  0.723
```

Now we calculate the coefficients of variation (SD/Mean).

```
cv.BHF <- 100 * sqrt(BHF$mse$mse)/BHF$est$eblup$eblup
results.BHF <- data.frame(CountyIndex = BHF$est$eblup$domain,
    CountyName = cornsoybeanmeans$CountyName, SampleSize = BHF$est$eblup$sampsize,
    eblup.BHF = BHF$est$eblup$eblup, cv.BHF)
print(results.BHF, row.names = FALSE)
##  CountyIndex CountyName SampleSize eblup.BHF   cv.BHF
##            1 CerroGordo          1  122.1954 8.036594
##            2   Hamilton          1  126.2280 7.338579
##            3      Worth          1  106.6638 8.901392
##            4   Humboldt          2  108.4222 7.239690
##            5   Franklin          3  144.3072 4.513586
##            6 Pocahontas          3  112.1586 5.685806
##            7  Winnebago          3  112.7801 6.350905
##            8     Wright          3  122.0020 5.535902
##            9    Webster          4  115.3438 5.218175
##           10    Hancock          5  124.4144 4.648261
##           11    Kossuth          5  106.8883 5.016463
##           12     Hardin          5  143.0312 3.986466
```

```
par(cex = 0.5)
corn.DIR <- direct(y = CornHec, dom = County, data = cornsoybean,
    replace = T)
plot(corn.DIR$Direct, results.BHF$eblup.BHF, xlab = "Direct",
    ylab = "BHF", xlim = c(70, 170), ylim = c(70, 170))
abline(0, 1)
```



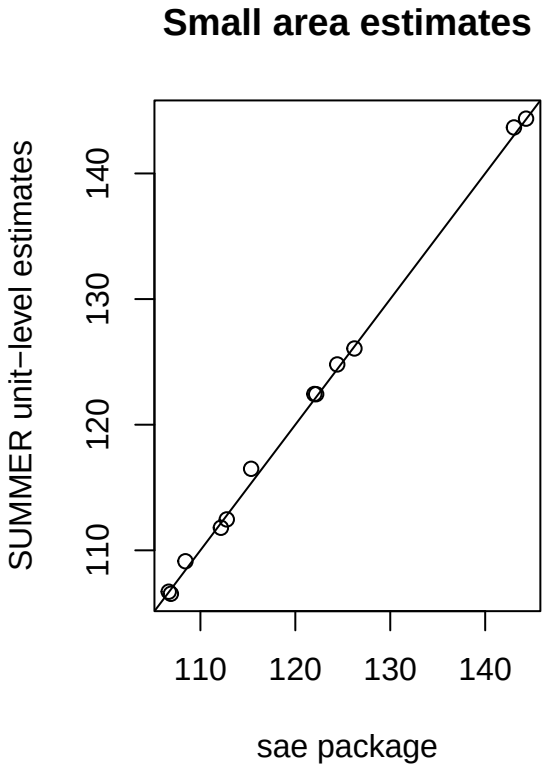
`SUMMER::smoothUnit` provides the ability to fit unit level models with unit level covariates for Gaussian response variables.

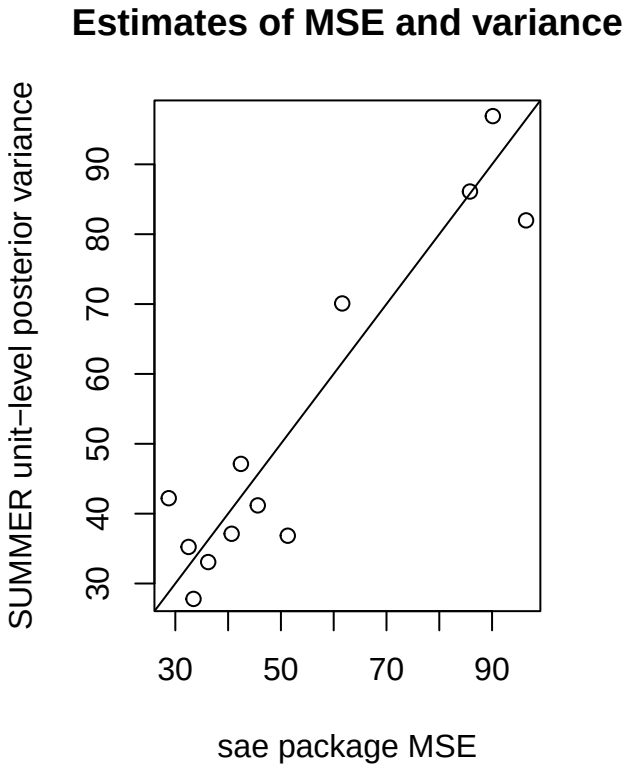
Note that in order to align the `SUMMER` estimates with those from the `sae` package, we specify a relatively flat prior on the variance of the area-specific random effect (`pc.u = 100, pc.alpha = 0.01` specifies a penalized complexity prior such that $P(\sigma_u > 100) = 0.01$ where σ_u is the standard deviation of the area-specific random effects).

```
library(survey)
library(SUMMER)
cornsoybean$id <- 1:dim(cornsoybean)[1]
Xsummer <- Xmean
colnames(Xsummer) = c("County", "CornPix", "SoyBeansPix")
des0 <- svydesign(ids = ~1, data = cornsoybean)
summer.bhf.unit <- smoothUnit(formula = CornHec ~ CornPix + SoyBeansPix,
    family = "gaussian", domain = ~County, design = des0, X.pop = Xsummer,
    pc.u = 1000, pc.alpha = 0.01, CI = 0.95)
```

Below, we plot comparisons of the `sae` and `SUMMER` results.

```
par(mfrow = c(1, 2))
range1 <- range(c(BHF$est$eblup$eblup, summer.bhf.unit$median))
plot(BHF$est$eblup$eblup, summer.bhf.unit$model.est$median, xlab = "sae package",
    ylab = "SUMMER unit-level estimates", main = "Small area estimates",
    xlim = range1, ylim = range1)
abline(0, 1)
range2 <- range(c(BHF$mse$mse, summer.bhf.unit$var))
plot(BHF$mse$mse, summer.bhf.unit$model.est$var, xlab = "sae package MSE",
    ylab = "SUMMER unit-level posterior variance", main = "Estimates of MSE and variance",
    xlim = range2, ylim = range2)
abline(0, 1)
```


Small area estimates
SUMMER unit−level estimates
110
120
130
140
sae package


Estimates of MSE and variance
SUMMER unit−level posterior variance
30
40
50
60
70
80
90
sae package MSE