Mixed Effects Models with Spatial Correlation Function Fitted Using the NLME Package

May 27, 2022

Michael C. Wimberly
Department of Geography and Environmental Sustainability, University of Oklahoma

```
library(sf)
library(tidyverse)
library(terra)
library(gstat)
library(nlme)
```

Data

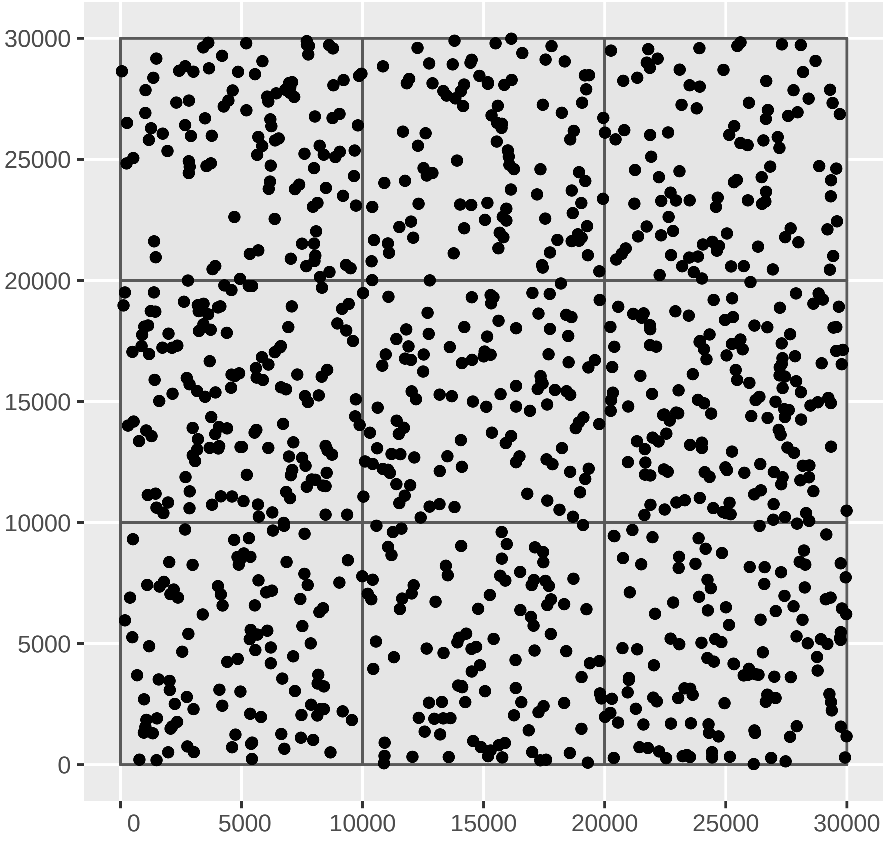
Start by generating some artificial data

1. 1000 students in nine schools
2. Each school has a different intercept, simulated as a uniform random variable between 50 and 70
3. Test scores modeled as a linear function of NDVI with slope of 10 added to the school-level intercept
4. NDVI is simulated as a Gaussian random field
5. Spatial and independent (non-spatial) errors are added to these "true" score values to generate the observed test scores
6. Spatial errors are simulated as a Gaussian random field
7. Non-spatial error are simulated as IID random variables

To start, we generate a polygon for the overall study area, and then subdivide it into nine square tiles (one for each school zone). Then we sample student homes at random locations and interect them with the school zones.

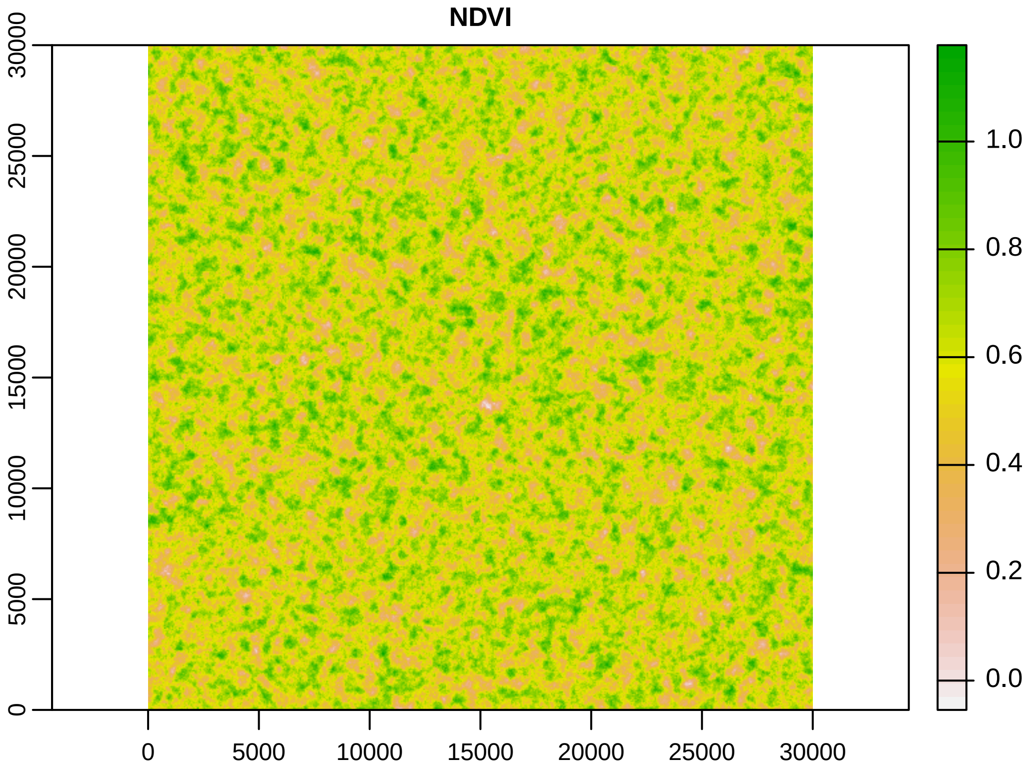
```
# Set random number seed
set.seed(22003)

# Generate bounding polygon for the study area
pol <- st_sfc(st_polygon(list(cbind(c(0,30000,30000,0,0),
                                    c(0,0,30000,30000,0)))))
h <- st_sf(pol)
# Split up into nine zones
sch_zones <- st_make_grid(h, cellsize = 10000)
sch_zones <- st_as_sf(sch_zones)
sch_zones$zone <- 1:9
# Generate 1000 student homes at random locations
```

```
homes <- st_sample(sch_zones, size = 1000)
homes <- st_as_sf(homes)
# Add the correponding school zone to each home locations
students <- st_join(homes, sch_zones)
coords <- st_coordinates(students)
students <- bind_cols(students,
                      data.frame(st_coordinates(students)))
# Map the zones and home locations
ggplot() +
  geom_sf(data = sch_zones) +
  geom_sf(data = students)
```

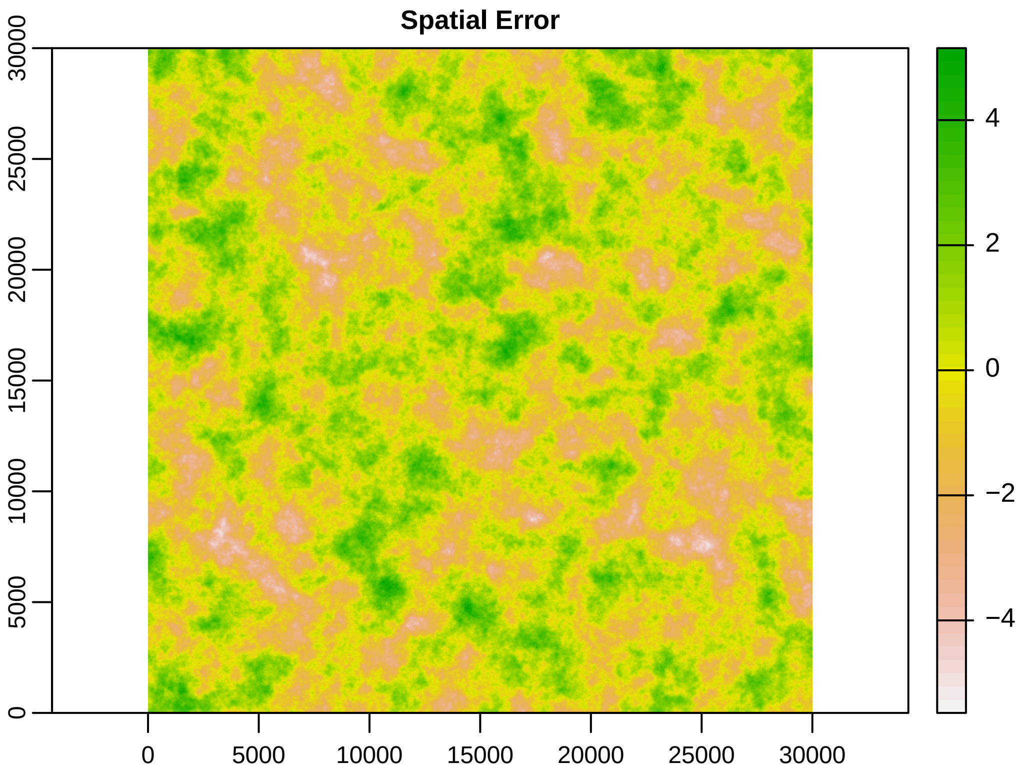


Next, we generate artificial data NDVI data and the spatial error term as spatially correlated gaussian random fields. We also generate another spatially correlated "garbage" variable that is independent of the test scores.

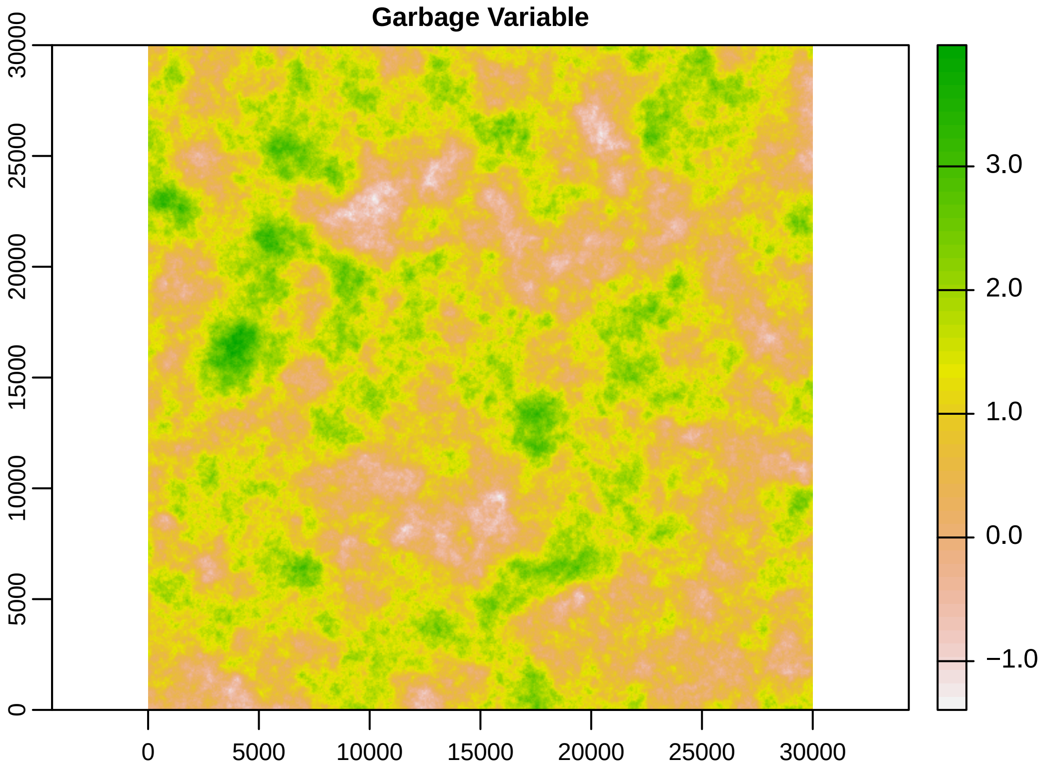
```
# Generate an empty grid for the random fields
blankraster <- rast(vect(h), resolution = 60)

# Model for NDVI (short range correlation)
gmodel_1 <- gstat(formula=z~1, locations=~x+y, dummy=T, beta=1,
                  model=vgm(psill=0.5, range=500, model='Sph'),
                  nmax=20)
# Model for spatially correlated error (longer-range correlation)
gmodel_2 <- gstat(formula=z~1, locations=~x+y, dummy=T, beta=1,
                  model=vgm(psill=0.5, range=2000, model='Sph'),
                  nmax=20)

# Model for spatially correlated "garbage" variable (longer-range correlation)
gmodel_3 <- gstat(formula=z~1, locations=~x+y, dummy=T, beta=1,
                  model=vgm(psill=0.5, range=3000, model='Sph'),
                  nmax=20)


# Generate spatially correlated random field for ndvi and the
# spatially correlated error
# Notes - these take a while to run
# For some reason, the interpolate() function doesn't work if I try to do
# just one simulation (nsim = 1), so I'm doing two for each variable
ndvi_interp <- interpolate(blankraster,
                           gmodel_1,
                           nsim = 2,
                           xyNames = c("x", "y"),
                           debug.level=0)
error_interp <- interpolate(blankraster,
                            gmodel_2,
                            nsim = 2,
                            xyNames = c("x", "y"),
                            debug.level=0)
garbage_interp <- interpolate(blankraster,
                              gmodel_3,
                              nsim = 2,
                              xyNames = c("x", "y"),
                              debug.level=0)

# Transform ndvi and spatial error terms to more "realistic" values
ndvi_interp <- (ndvi_interp + 2) * 0.2
error_interp <- (error_interp - 1) * 2

plot(ndvi_interp[[1]], main = "NDVI")
```



```
plot(error_interp[[1]], main = "Spatial Error")
```

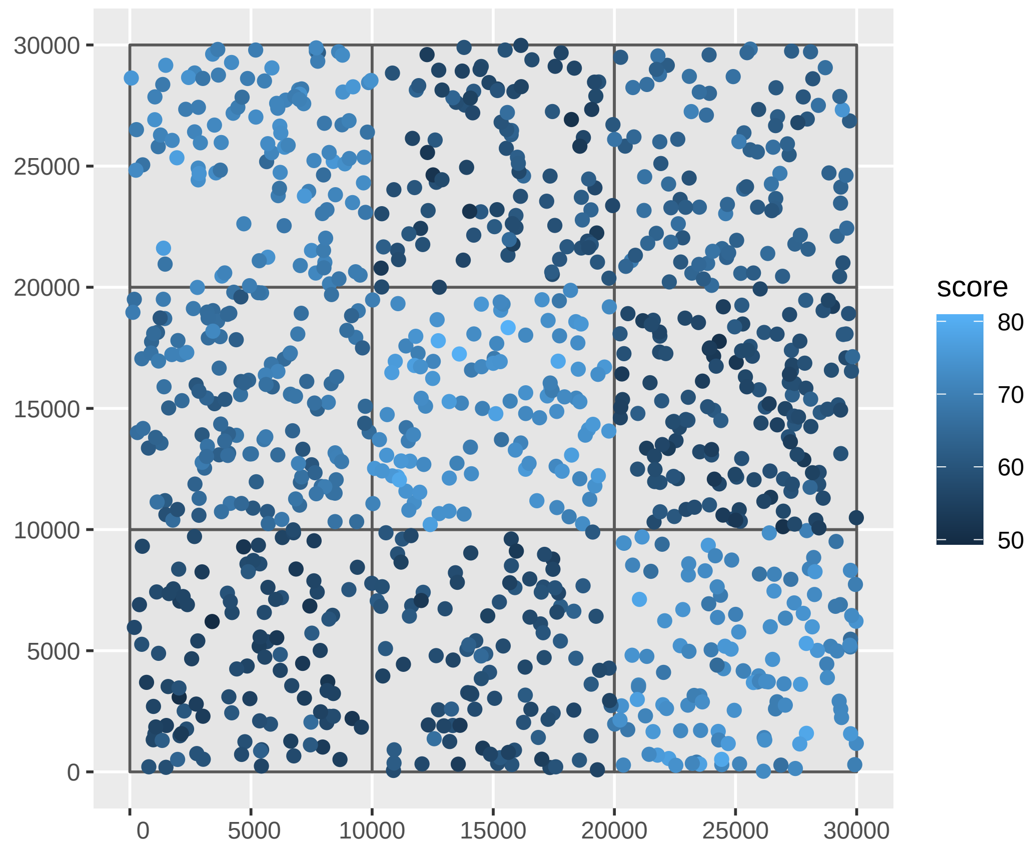


```
plot(garbage_interp[[1]], main = "Garbage Variable")
```



Extract ndvi, spatial error, and garbage variable values for each of the home locations. Then we generate the artificial scores by adding the group level intercept to ndvi x 10 (which we are summing to be the "true" ndvi effect in this example) and then adding noise from the spatial and non-spatial random errors. The map of the resulting scores shows the strong school-level random effect along with other, finer-scale spatial variaiblity resulting from the ndvi effects and the spatial error term.

```
# Random generate intercepts for each zone
zoneint <- runif(9, 50, 70)

# Extract ndvi, spatial error, and "garbage" variables for each home location
rsample <- terra::extract(c(ndvi_interp[[1]], error_interp[[1]], garbage_interp[[1]]),
                          vect(students))

names(rsample)[2:4] <- c("ndvi", "spaterr", "garbage")

students <- bind_cols(students, rsample[,2:4])

students <- students %>%
  mutate(groupint = zoneint[zone],
         inderr = rnorm(n(), 0, 2),
         score = groupint + ndvi * 10 + spaterr + inderr)

ggplot() +
  geom_sf(data = sch_zones) +
  geom_sf(data = students, aes(color = score), size = 2)
```



Now, we try to estimate the estimate the “true” ndvi effects from the artificial data using different models. The simple linear model (lm1) is technically wrong because it doesn’t account for the school level effects of the spatially correlated error. However, it does give us a reasonable estimate of the ndvi effect in this case. However, note that lm2 finds a statisically significant effect of the garbage variable on test scores, resulting in a type I error. This occurs because the naive linear model greater underestimates the standard errors of the fixed effect coefficients. The estimation of the ndvi effects from the lme model (lme1) is more defensible. Note also that the lme model does not result in a type I error for the garbage variable (lme2).

```
lm1 <- lm(score ~ ndvi, data = students)
summary(lm1)
```

```
##
## Call:
## lm(formula = score ~ ndvi, data = students)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.8855  -5.9555  -0.6898   6.1974  14.2504
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  57.9380     0.9463  61.228  < 2e-16 ***
## ndvi         10.4549     1.5283   6.841 1.37e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 6.75 on 998 degrees of freedom
## Multiple R-squared:  0.04479,    Adjusted R-squared:  0.04383
## F-statistic: 46.79 on 1 and 998 DF,  p-value: 1.373e-11
```

```
lm2 <- lm(score ~ garbage, data = students)
summary(lm2)
```

```
##
## Call:
## lm(formula = score ~ garbage, data = students)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.345  -5.828  -1.027   6.214  16.755
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  63.4709     0.3917 162.031   <2e-16 ***
## garbage       0.7618     0.3207   2.375   0.0177 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 6.887 on 998 degrees of freedom
## Multiple R-squared:  0.005622,   Adjusted R-squared:  0.004626
## F-statistic: 5.643 on 1 and 998 DF,  p-value: 0.01771
```

```
lme1 = lme(fixed = score ~ ndvi, data = students, random = ~1 | zone,
          corr = corSpatial(form = ~X + Y, type ="spherical", nugget = F), method = "REML")
summary(lme1)
```

```
## Linear mixed-effects model fit by REML
##   Data: students
##        AIC      BIC    logLik
##   4701.998 4726.527 -2345.999
##
## Random effects:
##  Formula: ~1 | zone
##         (Intercept) Residual
## StdDev:    6.651586 2.472516
##
## Correlation Structure: Spherical spatial correlation
##  Formula: ~X + Y | zone
##  Parameter estimate(s):
##    range
## 234.7121
## Fixed effects:  score ~ ndvi
##                 Value Std.Error  DF  t-value p-value
## (Intercept) 58.48295 2.2445713 990 26.05529       0
## ndvi         9.51921 0.5639481 990 16.87958       0
##  Correlation:
##      (Intr)
## ndvi -0.152
##
```

```
## Standardized Within-Group Residuals:
##          Min           Q1          Med           Q3          Max 
## -2.848333765 -0.656662863  0.007418181  0.645141389  2.944567671 
## 
## Number of Observations: 1000
## Number of Groups: 9
```

```
lme2 = lme(fixed = score ~ garbage, data = students, random = ~1 | zone,
          corr = corSpatial(form = ~X + Y, type ="spherical", nugget = F), method = "REML")
summary(lme2)
```

```
## Linear mixed-effects model fit by REML
##   Data: students 
##        AIC      BIC    logLik
##   4962.407 4986.935 -2476.203
## 
## Random effects:
##  Formula: ~1 | zone
##         (Intercept) Residual
## StdDev:    6.692645  2.80077
## 
## Correlation Structure: Spherical spatial correlation
##  Formula: ~X + Y | zone 
##  Parameter estimate(s):
##    range 
## 29.81371 
## Fixed effects:  score ~ garbage 
##                Value Std.Error  DF   t-value p-value
## (Intercept) 64.24327 2.2368784 990 28.720054   0.000
## garbage     -0.02078 0.1353857 990 -0.153485   0.878
##  Correlation: 
##         (Intr)
## garbage -0.061
## 
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max 
## -3.0201632 -0.6337005  0.0003846  0.6418003  3.8809565 
## 
## Number of Observations: 1000
## Number of Groups: 9
```