ADI Models

Paul Lantos

May 10, 2017

This document presents the scripting used for our spatial analyses.

The unadjusted model looks somewhat different than the GAM produced for our recent paper (Lantos et al, Geographic Disparities in Cytomegalovirus During Pregnancy, JPIDS 2017) because for this paper we have used spline-based models using the mgcv package. In our previous paper we used LOESS-based models using the GAM package. The odds predictions and the pattern of spatial heterogeneity are generally similar in this paper, but the exact predictions and significance contours vary slightly. Maps in this markdown document are made using R graphics (ggplot2 and viridis). For the publication maps were produced in ArcGIS by exporting the model results as .csv files and projecting in GIS. These were then exported layer-by-layer to Adobe Photoshop in order to produce drop shadows and manage figure layout.

Individual coordinates in the NC State Plane coordinate system are blurred to maintain patient confidentiality. Ages and ADI values are centered and standardized (means subtracted, then divided by standard deviation). The dataset used for this analysis contained 3504 subjects (excluding the 23 who had evidence of acute CMV seroconversion).

```
library(mgcv)
```

```
## Loading required package: nlme
```

```
## This is mgcv 1.8-15. For overview type 'help("mgcv-package")'.
```

```
library(ggplot2)
library(viridis)

setwd("~/Project Data/CMV ADI Study/R")
data <- read.csv("ADI_dataset.csv")
grid <- read.csv("ADI_grid.csv")
str(data)
```

```
## 'data.frame':    3504 obs. of  7 variables:
##  $ long  : num  [illegible]
##  $ lat   : num  [illegible]
##  $ race  : int  1 1 1 2 1 1 1 1 1 1 ...
##  $ result: int  1 0 1 1 1 0 0 0 0 1 ...
##  $ age   : num  -0.056 0.3009 -0.0502 2.1318 0.0274 ...
##  $ adi   : num  0.0027 0.0027 0.0027 0.0053 0.006 0.0063 0.0063 0.0063 0.0063 0.0063 ...
##  $ FIPS  : num  3.71e+11 3.71e+11 3.71e+11 3.72e+11 3.72e+11 ...
```

```r
str(grid)
```

```
## 'data.frame':    7243 obs. of  5 variables:
##  $ long: num  628788 629508 630228 630948 631668 ...
##  $ lat : num  202217 202217 202217 202217 202217 ...
##  $ age : num  2.45e-11 2.45e-11 2.45e-11 2.45e-11 2.45e-11 ...
##  $ race: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ adi : int  0 0 0 0 0 0 0 0 0 0 ...
```

```r
## models
## unadjusted
gam1 <- gam(result ~ s(long , lat), family=binomial, data=data)

## ADI alone
gam2 <- gam(result ~ s(long , lat) + adi, family=binomial, data=data)

## individual predictors (age - race interaction)
gam3 <- gam(result ~ s(long , lat) + age*race, family=binomial, data=data)

## individual predictors plus ADI
gam4 <- gam(result ~ s(long , lat) + age*race + adi, family=binomial, data=data)

## includes random effects term for neighborhood (block group FIPS code)
gam5 <- gam(result ~ s(long , lat) + age*race + adi+ s(FIPS , bs="re"), family=binomial, data=data)

## compare models
anova(gam2, gam3, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: result ~ s(long, lat) + adi
## Model 2: result ~ s(long, lat) + age * race
##   Resid. Df Resid. Dev     Df Deviance  Pr(>Chi)
## 1    3488.1     4656.1
## 2    3486.6     4414.6 1.4469   241.51 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(gam3, gam4, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: result ~ s(long, lat) + age * race
## Model 2: result ~ s(long, lat) + age * race + adi
##   Resid. Df Resid. Dev      Df Deviance Pr(>Chi)
## 1    3486.6     4414.6
## 2    3493.0     4417.2 -6.3644  -2.5535   0.8879
```

```r
anova(gam4, gam5, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: result ~ s(long, lat) + age * race + adi
## Model 2: result ~ s(long, lat) + age * race + adi + s(FIPS, bs = "re")
##   Resid. Df Resid. Dev         Df    Deviance Pr(>Chi)
## 1      3493     4417.2
```

```
## 2      3493     4417.2 1.7935e-07 -2.6687e-07
```

```
## model summaries
summary(gam1)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## result ~ s(long, lat)
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.23964    0.03456   6.933 4.12e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df Chi.sq  p-value
## s(long,lat) 18.88   23.8  111.2 3.24e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.0326   Deviance explained = 2.79%
## UBRE = 0.34582  Scale est. = 1         n = 3504
```

```
summary(gam2)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## result ~ s(long, lat) + adi
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.15961    0.06064  -2.632  0.00848 **
## adi          1.18612    0.15205   7.801 6.14e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df Chi.sq p-value
## s(long,lat) 10.07  13.94  29.76 0.00792 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.0402   Deviance explained = 3.21%
## UBRE = 0.3357  Scale est. = 1         n = 3504
```

```
summary(gam3)
```

```
##
## Family: binomial
```

```
## Link function: logit
##
## Formula:
## result ~ s(long, lat) + age * race
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.63243    0.12250 -13.326  < 2e-16 ***
## age         -0.46281    0.12436  -3.721 0.000198 ***
## race         1.33157    0.08253  16.134  < 2e-16 ***
## age:race     0.35280    0.07821   4.511 6.46e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df Chi.sq p-value
## s(long,lat) 9.641  13.39  13.21    0.45
##
## R-sq.(adj) =  0.106   Deviance explained = 8.23%
## UBRE = 0.26767  Scale est. = 1         n = 3504
```

```
summary(gam4) ## race*age interaction and ADI significant in model that includes both
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## result ~ s(long, lat) + age * race + adi
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.70668    0.11981 -14.245  < 2e-16 ***
## age         -0.43336    0.12363  -3.505 0.000456 ***
## race         1.26432    0.08535  14.814  < 2e-16 ***
## adi          0.50948    0.16050   3.174 0.001502 **
## age:race     0.35120    0.07785   4.511 6.44e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df Chi.sq p-value
## s(long,lat) 4.354  6.022  4.437   0.617
##
## R-sq.(adj) =  0.106   Deviance explained = 8.17%
## UBRE = 0.26595  Scale est. = 1         n = 3504
```

```
summary(gam5)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## result ~ s(long, lat) + age * race + adi + s(FIPS, bs = "re")
```

```
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.70665    0.12601 -13.544  < 2e-16 ***
## age         -0.43336    0.12363  -3.505 0.000456 ***
## race         1.26432    0.08535  14.814  < 2e-16 ***
## adi          0.50948    0.16050   3.174 0.001502 **
## age:race     0.35120    0.07785   4.511 6.44e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                   edf Ref.df Chi.sq p-value
## s(long,lat) 4.354e+00  6.022  4.437   0.617
## s(FIPS)     8.755e-07  1.000  0.000   0.305
##
## R-sq.(adj) =  0.106   Deviance explained = 8.17%
## UBRE = 0.26595  Scale est. = 1         n = 3504
```

```
## Predict odds ratio models to grid, compute areas of significant spatial effect
## Unadjusted model
gridlen = length(grid[,1])

m.1 = gam(result ~ s(long , lat), family = binomial, data = data)
m.0 = gam(result ~ 1, family = binomial, data = data)

synoutput = matrix(ncol = 4,nrow = gridlen,0)
synoutput = grid[,1:2]
synrefnull = mean(predict.gam(m.0 , grid))
synoutput[,3] = matrix(exp(predict.gam(m.1 , grid) - synrefnull))[,1]  ## odds ratio (local / average od

Rglobstat = (data.frame(anova(m.0, m.1, test = "Chisq"))[2,5])

ID = length(data$result)
coords = data[,1:2]
m.data = data
permresults = matrix(ncol=1000 , nrow = gridlen, 0)
permresults[,1] = predict.gam(m.1 , grid)
permrank = matrix(ncol = 1000, nrow = gridlen, 0)
devrank = matrix(ncol = 1, nrow = 1000, 0)
devdata = data.frame(anova(m.0, m.1, test="Chisq"))
devrank[1,1] = devdata[2,4]
i = 2
j = 2
while (j<1001)
{
  index = sample(ID, replace=F)
  m.data[,1:2]=coords[index,]


  m.gam = gam(result ~ s(long , lat), family = binomial(logit) , data = m.data)
  permresults[,i] = predict.gam(m.gam , grid)

  devdata = data.frame(anova(m.0 , m.gam , test="Chisq"))
```

```
  devrank[i,1] = devdata[2,4]

  i=i+1
  if (j%%10==0) print(i)
  j=j+1
}
```

```
## [1] 11
## [1] 21
## [1] 31
## [1] 41
## [1] 51
## [1] 61
## [1] 71
## [1] 81
## [1] 91
## [1] 101
## [1] 111
## [1] 121
## [1] 131
## [1] 141
## [1] 151
## [1] 161
## [1] 171
## [1] 181
## [1] 191
## [1] 201
## [1] 211
## [1] 221
## [1] 231
## [1] 241
## [1] 251
## [1] 261
## [1] 271
## [1] 281
## [1] 291
## [1] 301
## [1] 311
## [1] 321
## [1] 331
## [1] 341
## [1] 351
## [1] 361
## [1] 371
## [1] 381
## [1] 391
## [1] 401
## [1] 411
## [1] 421
## [1] 431
## [1] 441
## [1] 451
## [1] 461
## [1] 471
```

```
## [1] 481
## [1] 491
## [1] 501
## [1] 511
## [1] 521
## [1] 531
## [1] 541
## [1] 551
## [1] 561
## [1] 571
## [1] 581
## [1] 591
## [1] 601
## [1] 611
## [1] 621
## [1] 631
## [1] 641
## [1] 651
## [1] 661
## [1] 671
## [1] 681
## [1] 691
## [1] 701
## [1] 711
## [1] 721
## [1] 731
## [1] 741
## [1] 751
## [1] 761
## [1] 771
## [1] 781
## [1] 791
## [1] 801
## [1] 811
## [1] 821
## [1] 831
## [1] 841
## [1] 851
## [1] 861
## [1] 871
## [1] 881
## [1] 891
## [1] 901
## [1] 911
## [1] 921
## [1] 931
## [1] 941
## [1] 951
## [1] 961
## [1] 971
## [1] 981
## [1] 991
## [1] 1001
```

```
tempdev = rank(devrank)
devglobstat=((1000-tempdev[1]) / 1000)
k = 1
while (k <= gridlen)
{
  permrank[k,]=rank(permresults[k,])
  k = k + 1
}
synoutput[,4] = permrank[,1]
unadj.model = synoutput
unadj.model[,5:6] = predict.gam(m.1, grid, se.fit = TRUE)
unadj.model = unadj.model[,-5]
names(unadj.model) = c("long" , "lat" , "OR" , "Rank", "SE")



## Adjusted model
gridlen = length(grid[,1])

m.1 = gam(result ~ s(long , lat) + age*race + adi, family = binomial, data = data)
m.0 = gam(result ~ age*race + adi, family = binomial, data = data)

synoutput = matrix(ncol = 4,nrow = gridlen,0)
synoutput = grid[,1:2]
synrefnull = mean(predict.gam(m.0 , grid))
synoutput[,3] = matrix(exp(predict.gam(m.1 , grid) - synrefnull))[,1]  ## odds ratio (local / average o

Rglobstat = (data.frame(anova(m.0, m.1, test = "Chisq"))[2,5])

ID = length(data$result)
coords = data[,1:2]
m.data = data
permresults = matrix(ncol=1000 , nrow = gridlen, 0)
permresults[,1] = predict.gam(m.1 , grid)
permrank = matrix(ncol = 1000, nrow = gridlen, 0)
devrank = matrix(ncol = 1, nrow = 1000, 0)
devdata = data.frame(anova(m.0, m.1, test="Chisq"))
devrank[1,1] = devdata[2,4]
i = 2
j = 2
while (j<1001)
{
  index = sample(ID, replace=F)
  m.data[,1:2]=coords[index,]


  m.gam = gam(result ~ s(long , lat) + age*race + adi, family = binomial(logit) , data = m.data)
  permresults[,i] = predict.gam(m.gam , grid)

  devdata = data.frame(anova(m.0 , m.gam , test="Chisq"))
  devrank[i,1] = devdata[2,4]

  i=i+1
```

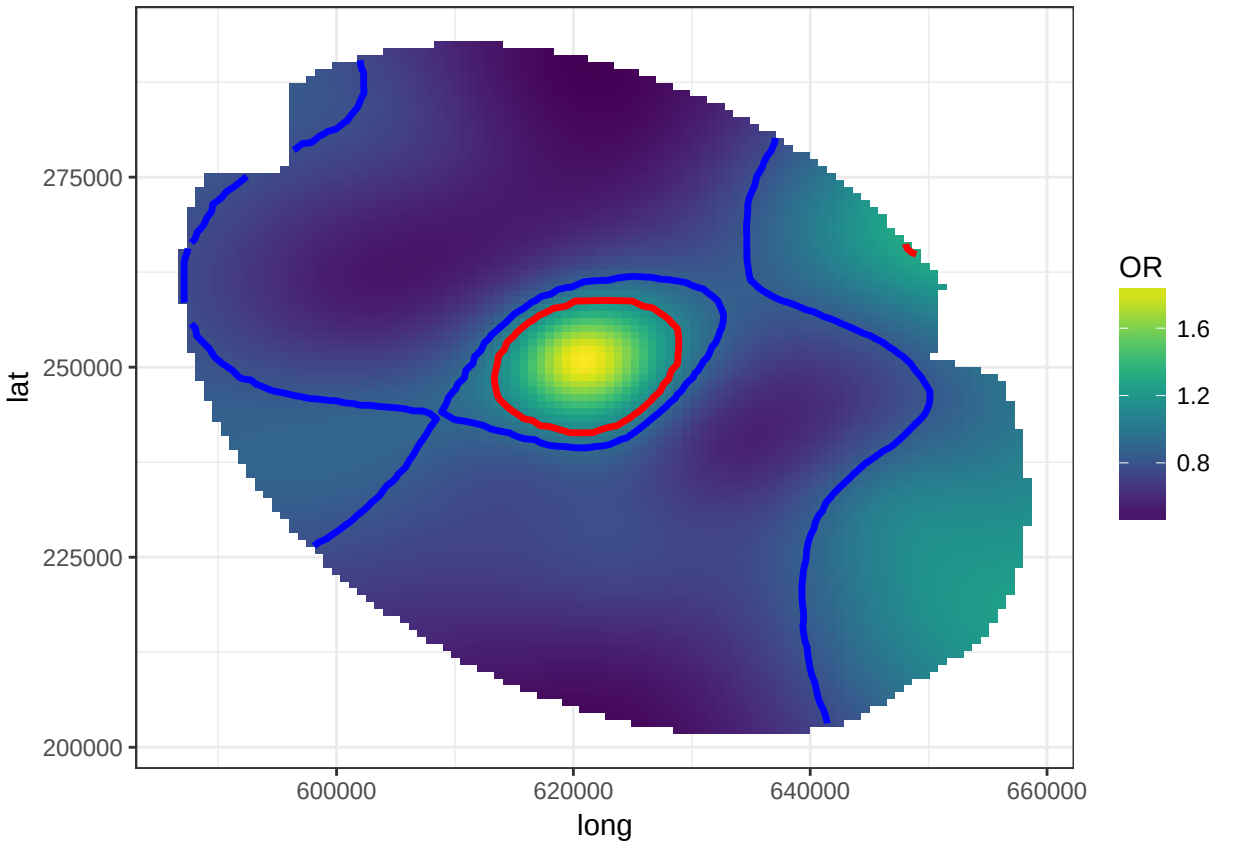
```
  if (j%%10==0) print(i)
  j=j+1
}
```

```
## [1] 11
## [1] 21
## [1] 31
## [1] 41
## [1] 51
## [1] 61
## [1] 71
## [1] 81
## [1] 91
## [1] 101
## [1] 111
## [1] 121
## [1] 131
## [1] 141
## [1] 151
## [1] 161
## [1] 171
## [1] 181
## [1] 191
## [1] 201
## [1] 211
## [1] 221
## [1] 231
## [1] 241
## [1] 251
## [1] 261
## [1] 271
## [1] 281
## [1] 291
## [1] 301
## [1] 311
## [1] 321
## [1] 331
## [1] 341
## [1] 351
## [1] 361
## [1] 371
## [1] 381
## [1] 391
## [1] 401
## [1] 411
## [1] 421
## [1] 431
## [1] 441
## [1] 451
## [1] 461
## [1] 471
## [1] 481
## [1] 491
## [1] 501
```

```
## [1] 511
## [1] 521
## [1] 531
## [1] 541
## [1] 551
## [1] 561
## [1] 571
## [1] 581
## [1] 591
## [1] 601
## [1] 611
## [1] 621
## [1] 631
## [1] 641
## [1] 651
## [1] 661
## [1] 671
## [1] 681
## [1] 691
## [1] 701
## [1] 711
## [1] 721
## [1] 731
## [1] 741
## [1] 751
## [1] 761
## [1] 771
## [1] 781
## [1] 791
## [1] 801
## [1] 811
## [1] 821
## [1] 831
## [1] 841
## [1] 851
## [1] 861
## [1] 871
## [1] 881
## [1] 891
## [1] 901
## [1] 911
## [1] 921
## [1] 931
## [1] 941
## [1] 951
## [1] 961
## [1] 971
## [1] 981
## [1] 991
## [1] 1001
```

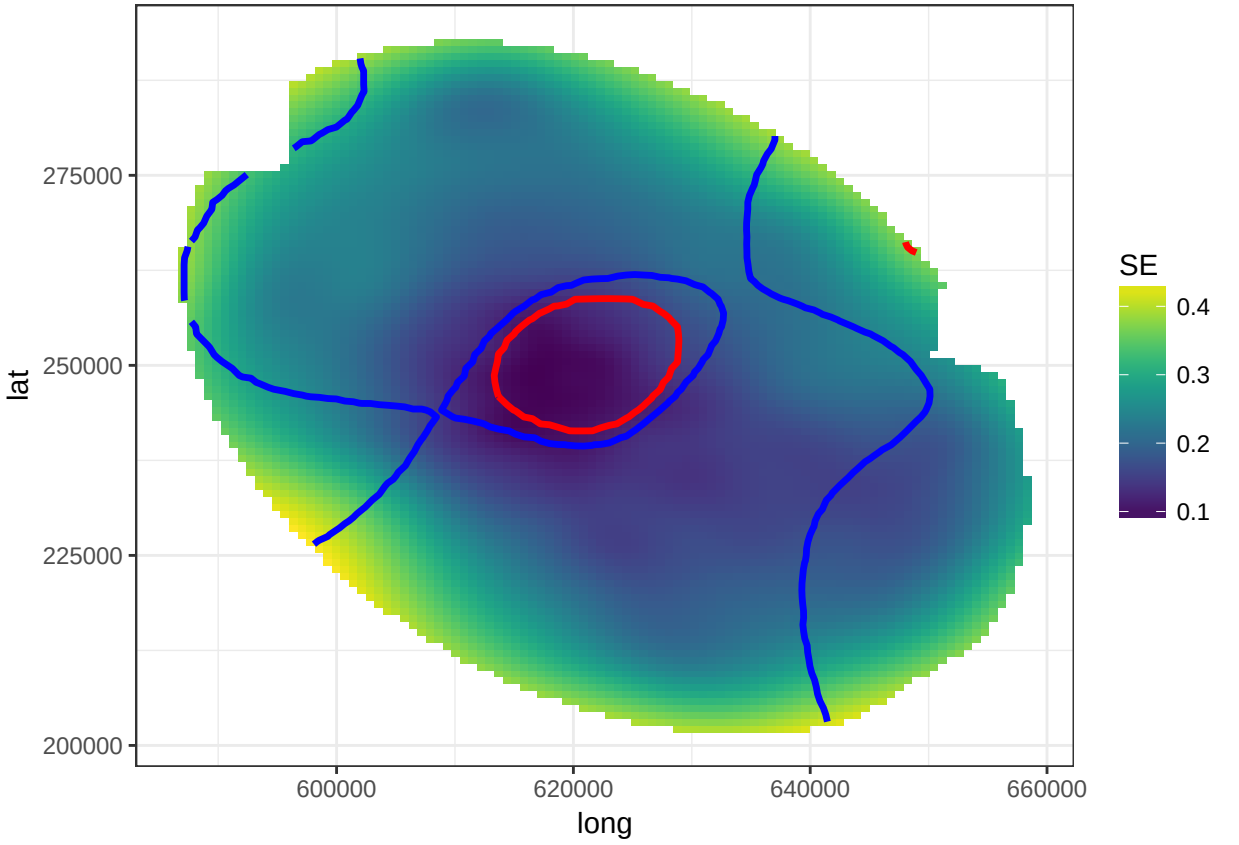
```
tempdev = rank(devrank)
devglobstat=((1000-tempdev[1]) / 1000)
k = 1
```

```
while (k <= gridlen)
{
  permrank[k,]=rank(permresults[k,])
  k = k + 1
}
synoutput[,4] = permrank[,1]
adj.model = synoutput
adj.model[,5:6] = predict.gam(m.1, grid, se.fit = TRUE)
adj.model = adj.model[,-5]
names(adj.model) = c("long" , "lat" , "OR" , "Rank", "SE")


### Plots -- SE is in units of fitted values

### Unadjusted model

## Odds Ratio Map
ggplot(unadj.model) +
  geom_raster(aes(long, lat, fill = OR))+
  scale_fill_viridis()+
  geom_contour(aes(long, lat, z=Rank), breaks=25, color="Blue", cex=1.2, lty=1)+
  geom_contour(aes(long, lat, z=Rank), breaks=975, color="Red", cex=1.2, lty=1)+
  theme_bw()
```



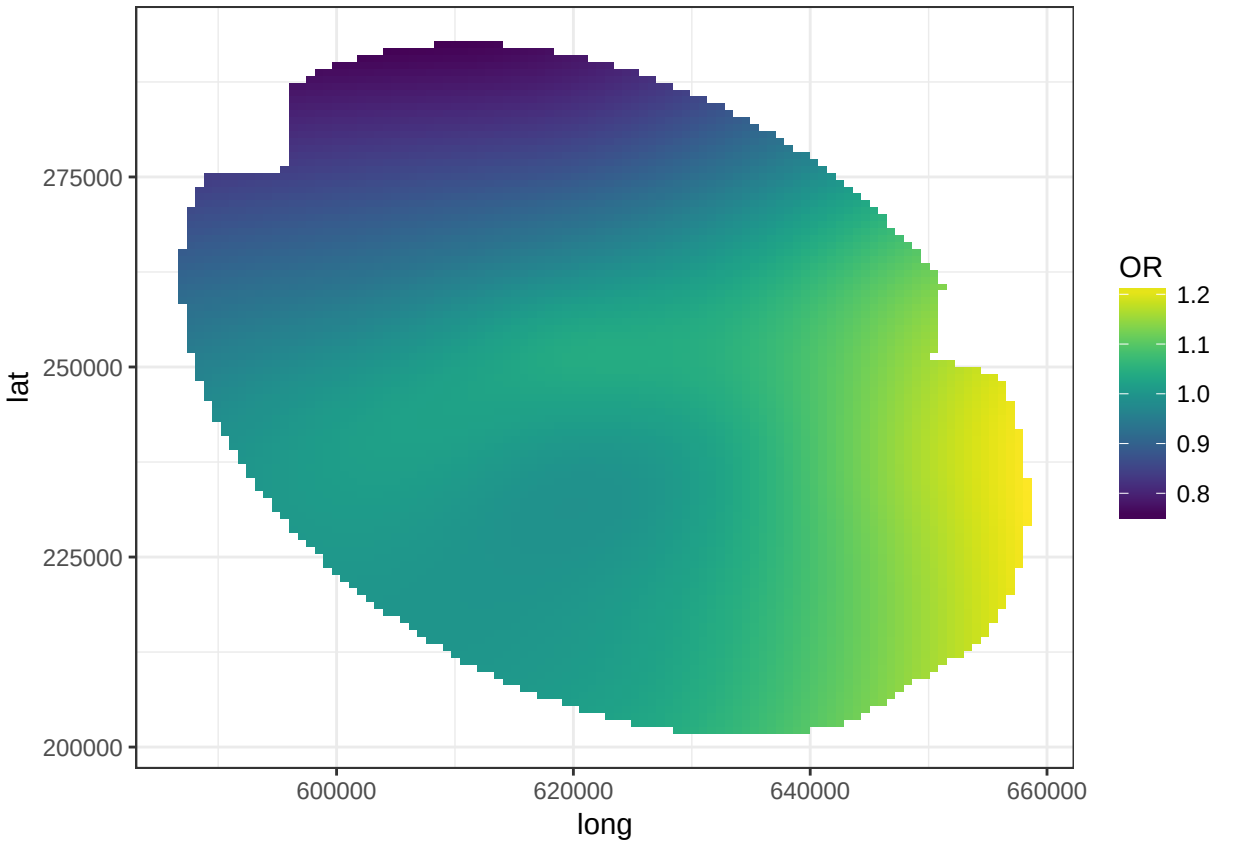

```
## Standard Error Map
ggplot(unadj.model) +
```

```
  geom_raster(aes(long, lat, fill = SE))+
  scale_fill_viridis()+
  geom_contour(aes(long, lat, z=Rank), breaks=25, color="Blue", cex=1.2, lty=1)+
  geom_contour(aes(long, lat, z=Rank), breaks=975, color="Red", cex=1.2, lty=1)+
  theme_bw()
```




```
### Adjusted for age*race interaction and ADI

## Odds Ratio Map
ggplot(adj.model) +
  geom_raster(aes(long, lat, fill = OR))+
  scale_fill_viridis()+
  geom_contour(aes(long, lat, z=Rank), breaks=25, color="Blue", cex=1.2, lty=1)+
  geom_contour(aes(long, lat, z=Rank), breaks=975, color="Red", cex=1.2, lty=1)+
  theme_bw()
```

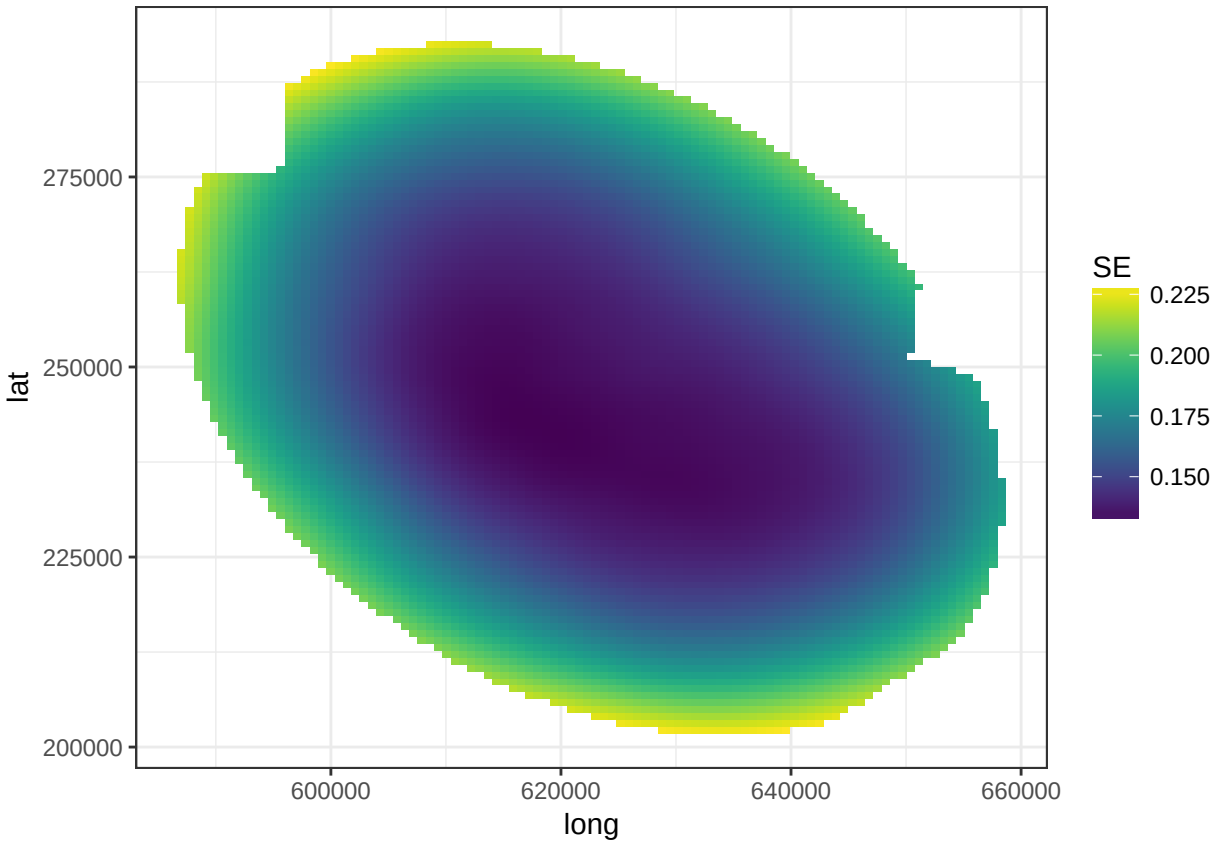
```
## Warning: Not possible to generate contour data

## Warning: Not possible to generate contour data
```




```
## Standard Error Map
ggplot(adj.model) +
  geom_raster(aes(long, lat, fill = SE))+
  scale_fill_viridis()+
  geom_contour(aes(long, lat, z=Rank), breaks=25, color="Blue", cex=1.2, lty=1)+
  geom_contour(aes(long, lat, z=Rank), breaks=975, color="Red", cex=1.2, lty=1)+
  theme_bw()
```

```
## Warning: Not possible to generate contour data
```

```
## Warning: Not possible to generate contour data
```


275000
250000
225000
200000
lat
600000
620000
640000
660000
long
SE
0.225
0.200
0.175
0.150