BIOST 555 Assignment 3

Benjamin Stan

February 22, 2022

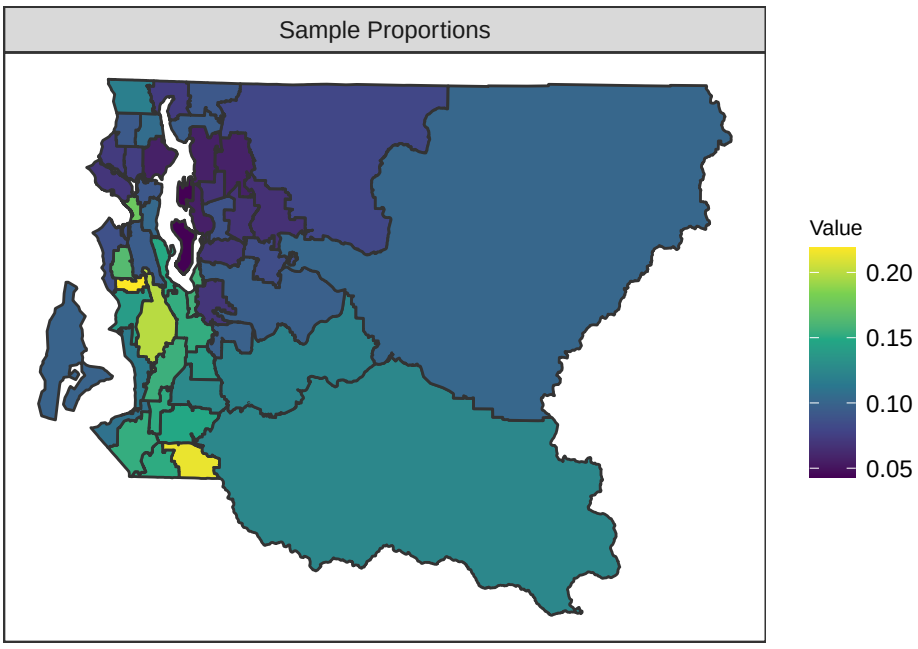
Problem 1

The binomial model below was considered.

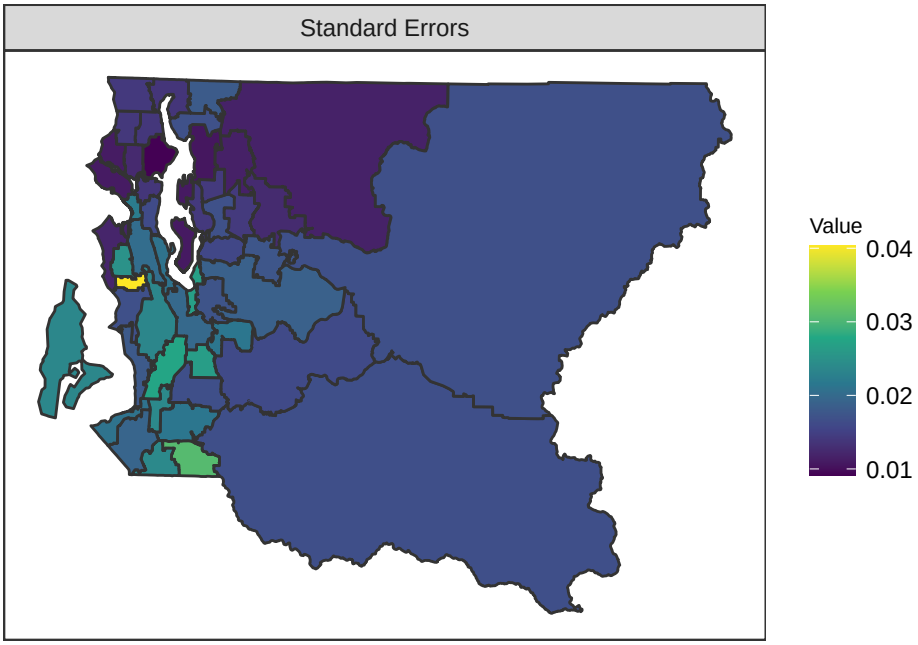
$$Y_i|p_i \sim_{iid} Binomial(n_i, p_i)$$

The map below shows the naive sample proportion ($\hat{p}_i = y_i/n_i$) in each region.

$$\begin{aligned} Y_i|p_i &\sim_{iid} Binomial(n_i, p_i) \\ \hat{p}_i &= y_i/n_i \\ \hat{se} &= \sqrt{\hat{p}_i(1-\hat{p}_i)/n_i} \end{aligned}$$



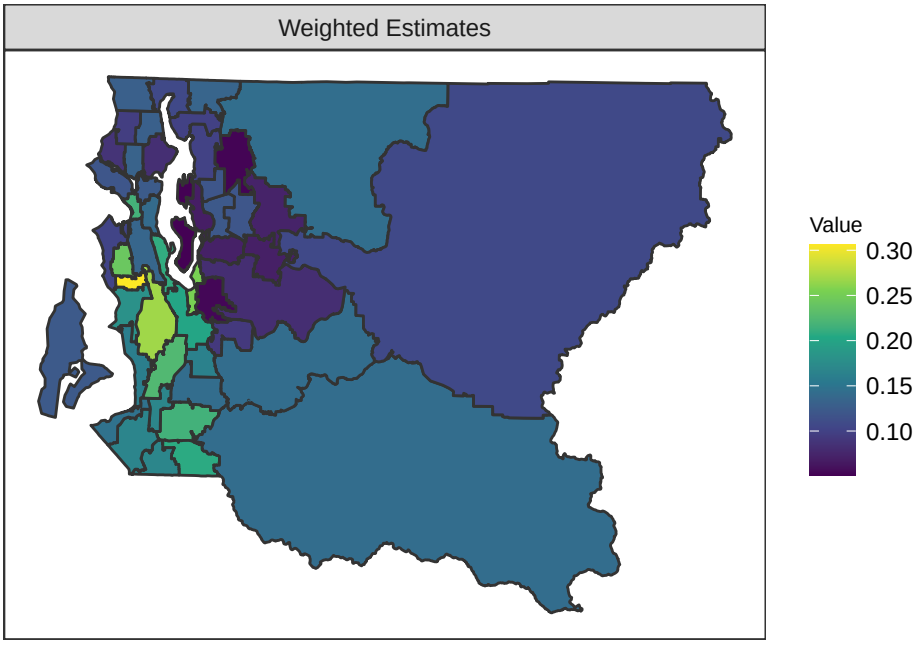
The map below shows the standard errors ($\hat{se} = \sqrt{\hat{p}_i(1-\hat{p}_i)/n_i}$) in each region.



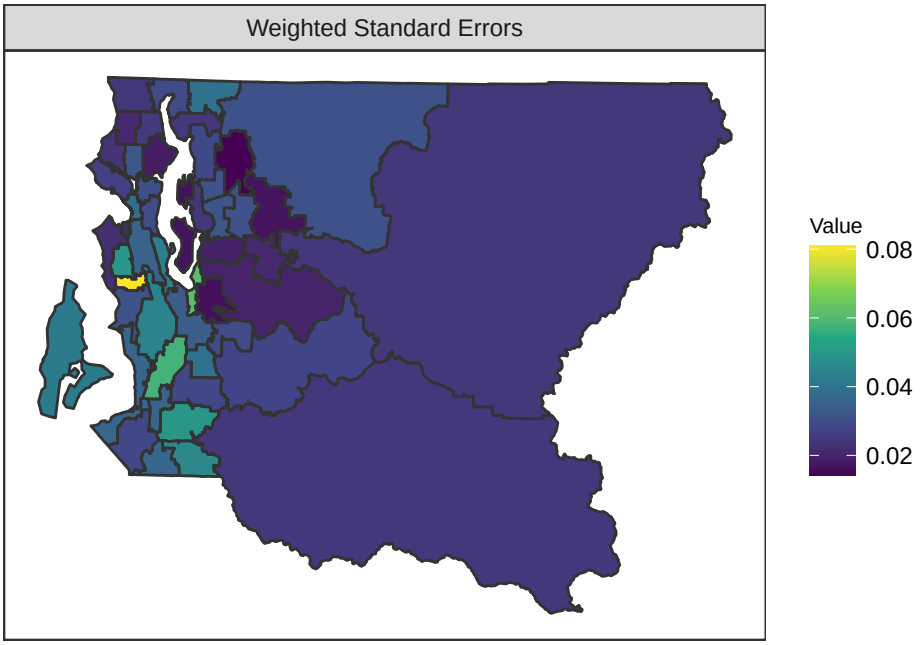
Problem 2

The map below shows the weighted estimates, $\hat{p}_i^w$, in each region.

$$\hat{p_i}^w = \frac{\sum_{k \epsilon S_i} w_k y_k}{\sum_{k \epsilon S_i} w_k}$$
$$\hat{se} = \sqrt{\hat{p_i}^w (1 - \hat{p_i}^w)/n_i}$$

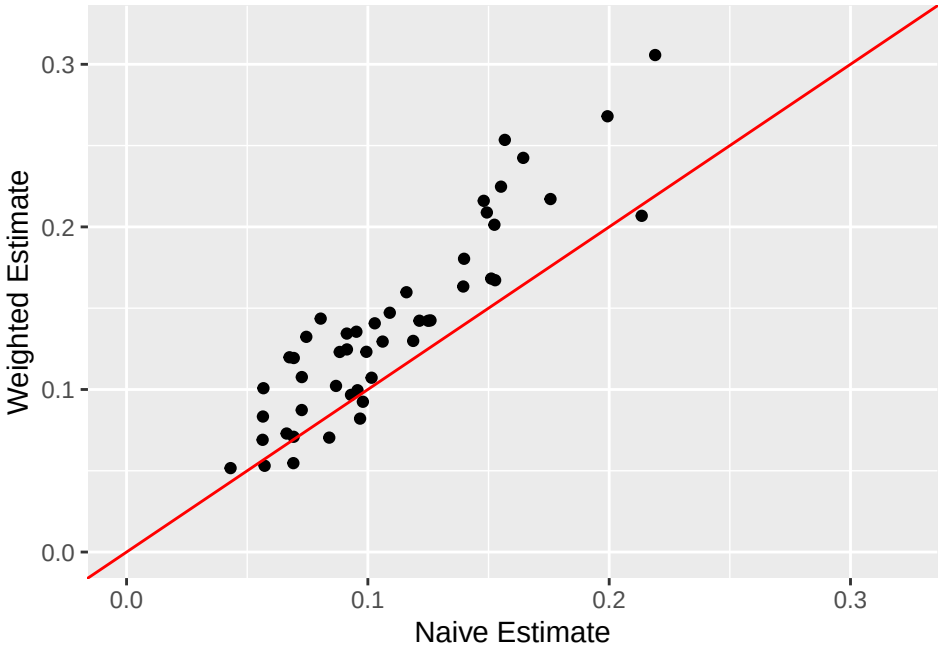


The map below shows the standard errors of the weighted estimates, $\hat{p}_i^w$, in each region.

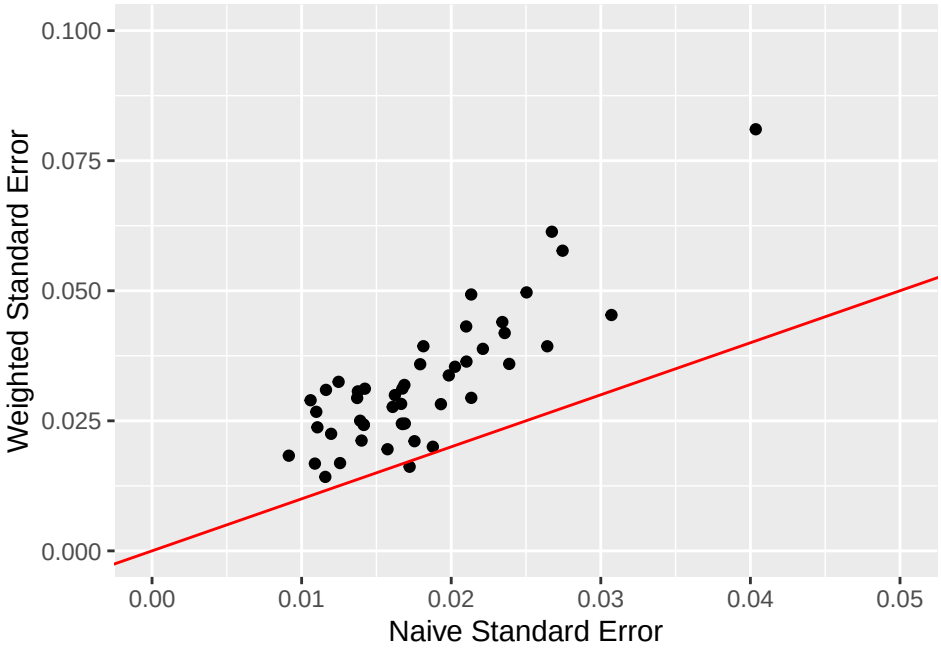


Problem 3

The naive and weighted estimates of p_i are show below.



The naive and weighted standard errors of p_i estimates are show below.



The weighted estimates of p_i tend to be larger than the corresponding naive estimates. This is particularly true for naive estimates above 0.1. Similarly, the standard errors of the weighted estimates are higher than all corresponding naive standard errors with the exception of one region. The weighted summaries are more appropriate because they incorporate information about the sampling mechanism in order to better represent the population(s) of interest.

Problem 4

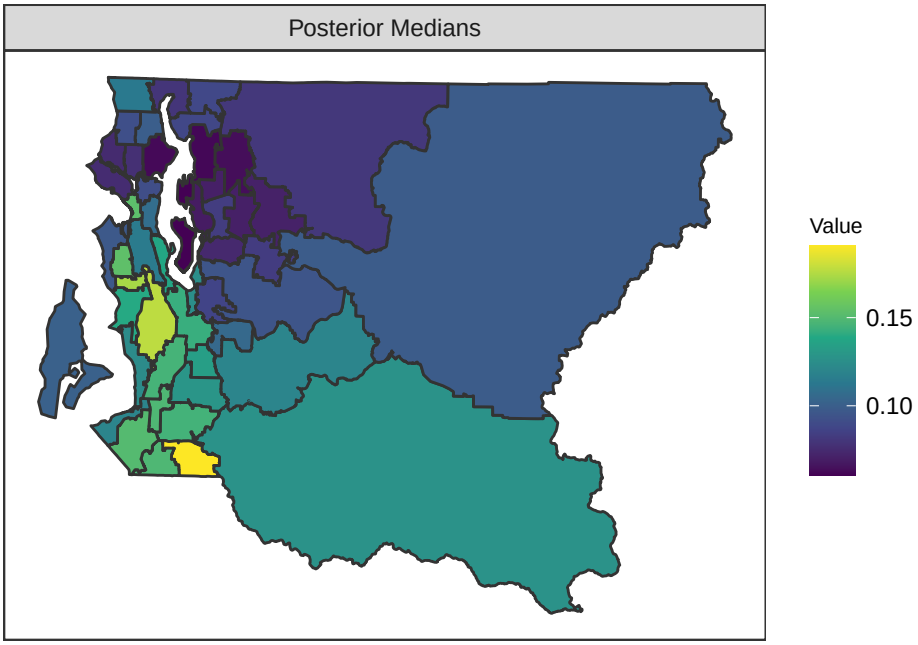
The binomial smoothing model below was considered.

$$Y_i|p_i \sim_{iid} Binomial(n_i, p_i)$$
$$\theta_i = log\left(\frac{p_i}{1-p_i}\right) = \alpha + \beta_i$$

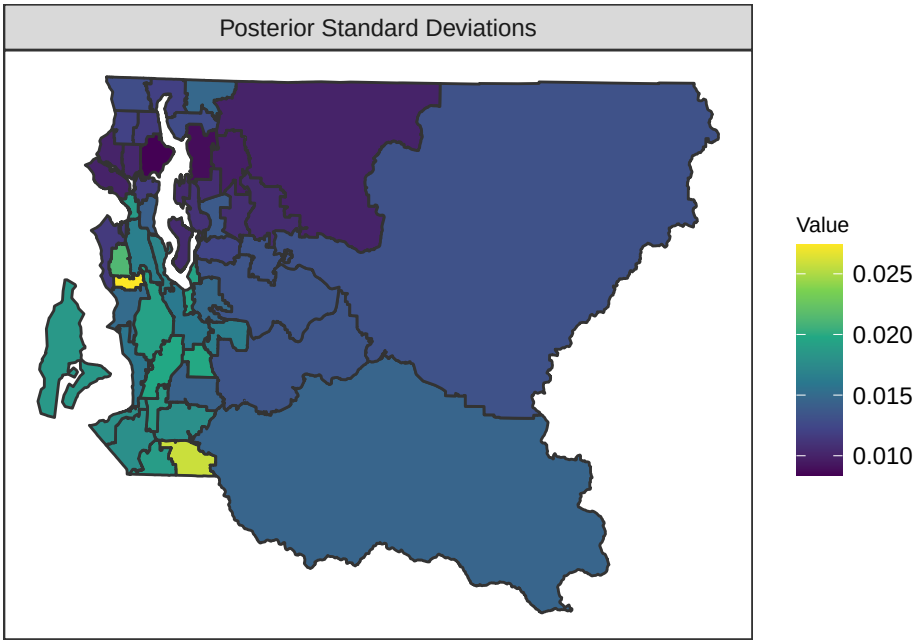
Where

- α is the intercept
- b_i are BYM2 random effects

The map of posterior medians of p_i is shown below.

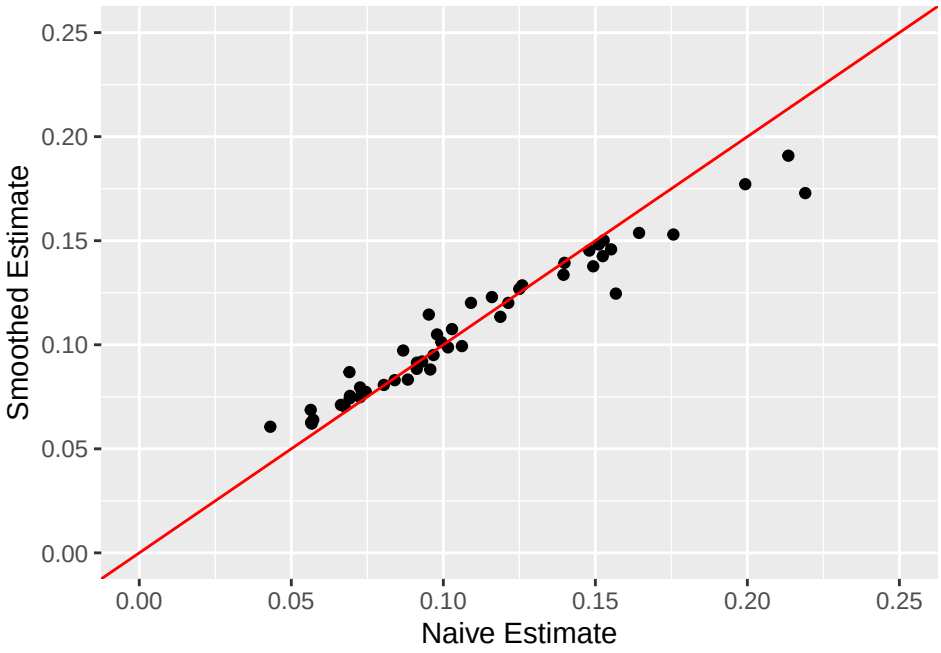


The map of posterior standard deviations of p_i is shown below.

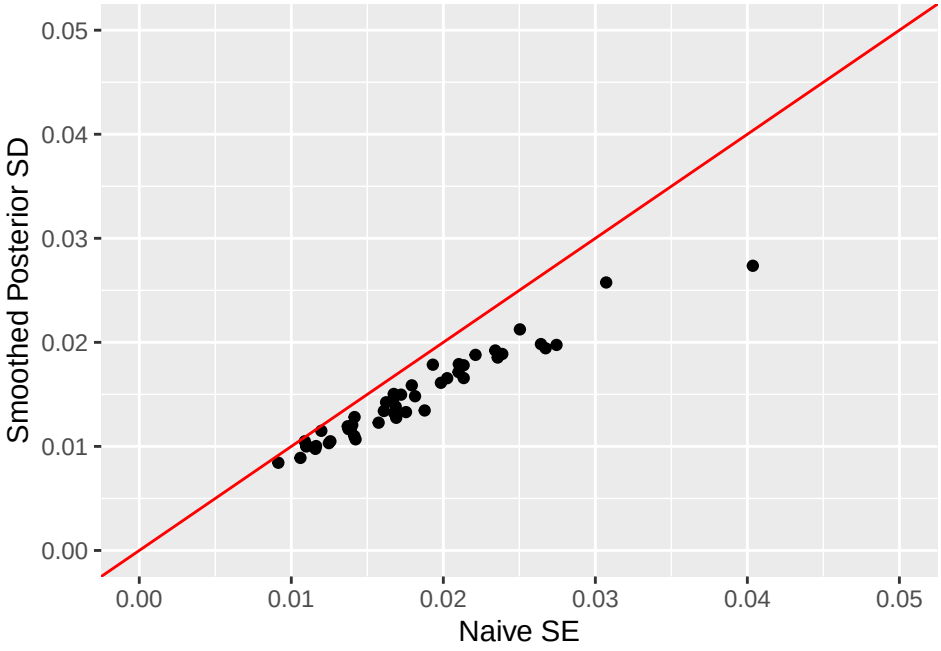


Problem 5

The naive and smoothed binomial estimates of p_i are show below.



The naive and smoothed binomial standard errors of p_i are show below.



By examining the first plot, it can be observed that the smoothed estimates are closer to their mean value than their corresponding naive estimates. For naive estimates below 0.075, the corresponding smoothed estimates tend to be larger. The two methods produced similar results between the values of 0.075 and 0.15. For naive estimates larger than 0.15, the corresponding smoothed estimates tend to be smaller. The second plot shows that the smoothed posterior standard deviations are smaller than the corresponding naive standard error for all regions. The differences between the two values increase for larger values of the naive standard error.

Problem 6

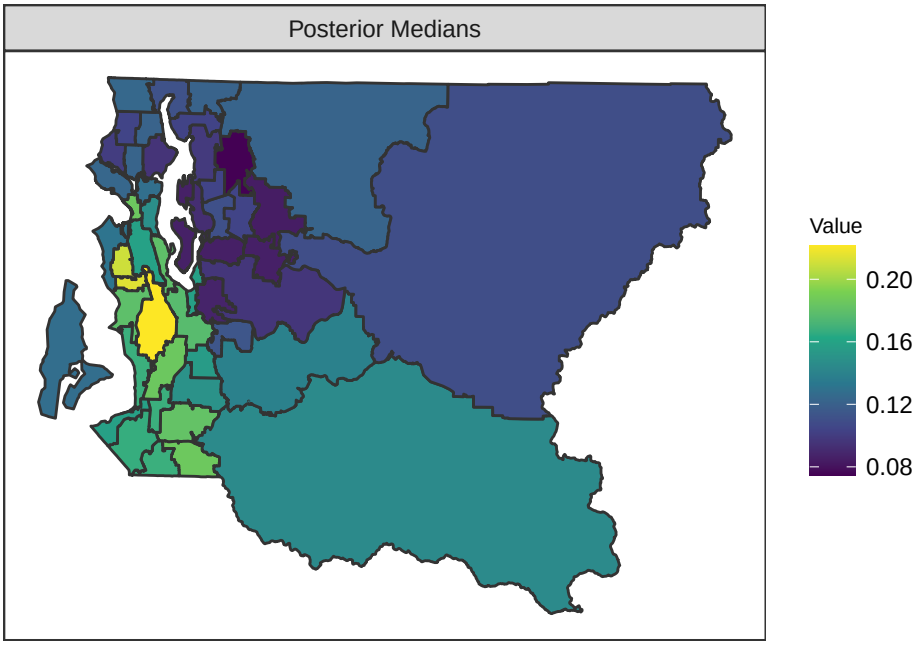
Using a definition for the transformed weighted estimates, the following model was considered.

$$\begin{aligned}
\hat{\theta}_i &= log\left[\frac{\hat{p}_i^w}{(1-\hat{p}_i^w)}\right] \\
\hat{\theta}_i|\theta_i &\sim_{iid} N(\theta_i, \hat{V}_i) \\
\theta_i &= \alpha + b_i \\
\theta_i &= \alpha + x_i^T\beta + b_i
\end{aligned}$$

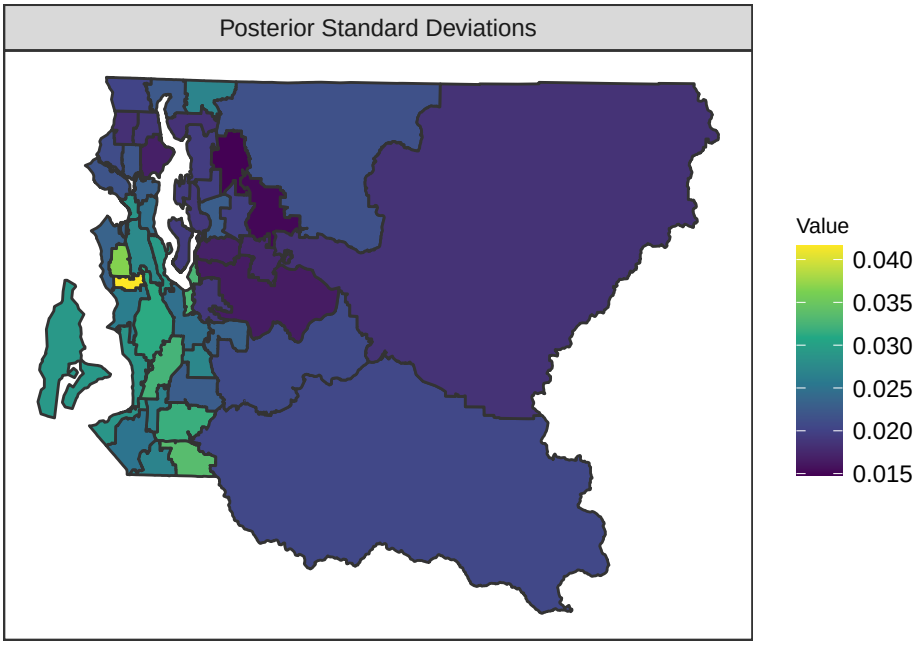
Where

- $\hat{V}_i$ are the estimated design-based variances of $\hat{\theta}_i$
- α is the intercept
- b_i are BYM2 random effects

The map of posterior medians of p_i is shown below.

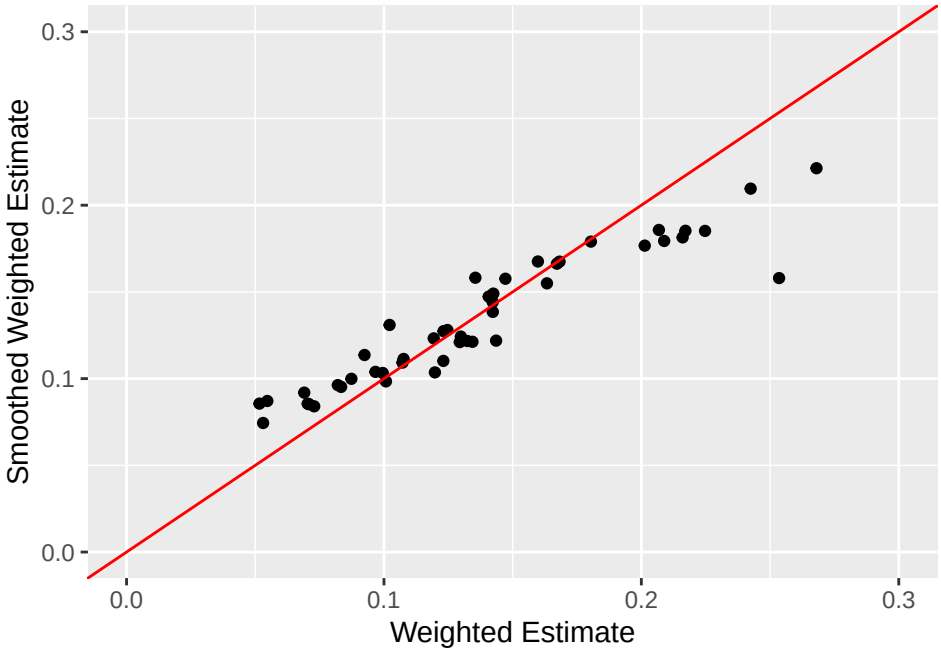


The map of posterior standard deviations of p_i is shown below.

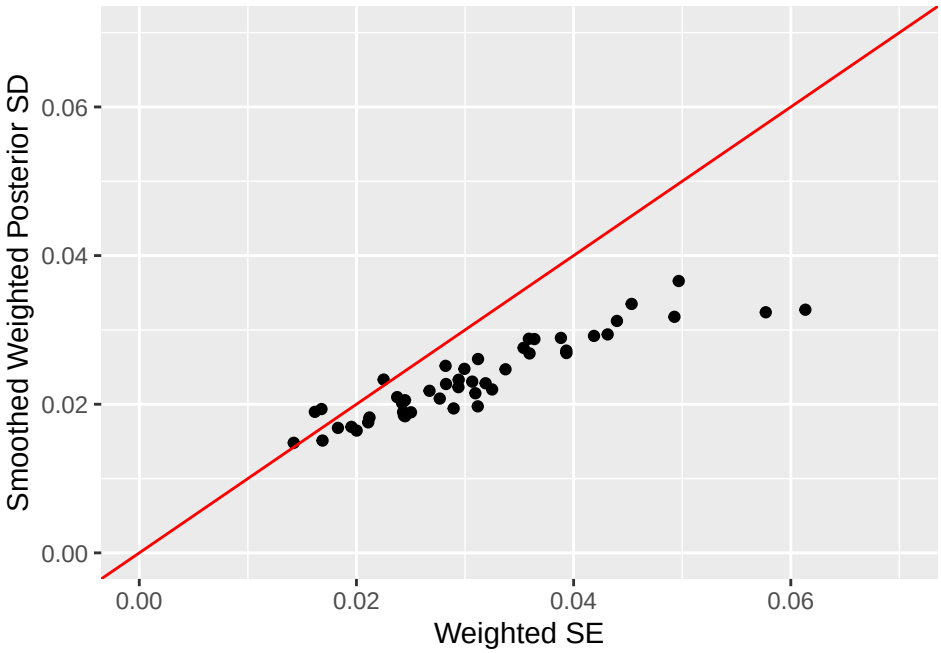


Problem 7

The weighted and smoothed weighted estimates of p_i are show below.



The weighted and smoothed weighted standard errors of p_i are show below.



By examining the plot of estimates, it appears that the smoothed weighted estimates are closer to the mean value than the corresponding weighted estimates. For weighted estimate values below the weighted estimate mean of 0.14, the corresponding smoothed weighted estimates tend to be larger. For weighted estimate values above 0.14, the the corresponding smoothed weighted estimates tend to be smaller. When comparing the posterior standard deviations of the smoothed weighted estimates to the standard errors of the weighted estimates, the former values are smaller for all but four regions. The differences between the two values increase for larger values of the weighted SE. Between these two methods, the weighted smoothed estimates would be recommended because they capture the spatial relationship between neighboring regions. This removes the notion that each region is a standalone entity and allows for a reduction in variance across estimates.

Problem 8

The smoking prevalence across King County varies greatly across HRAs. The SeaTac/Tukwila region has the highest estimated prevalence (using smoothed weighted methods) of 22%, and Redmond has the lowest estimated prevalence of 7.4%. Regions in the southern and western portions of the county tend to have higher prevalence of smoking. Redmond also has the lowest posterior estimate of standard deviation at 1.5%, while North Highline has the highest posterior standard deviation at 4.2%.

Appendix

```
## Set working directory and load libraries
setwd("/Users/bstan/Documents/UW/Courses/BIOST 555")
rm(list = ls())
library(tidyverse)
library(tidyr)
library(tinytex)
library(knitr)
library(rgdal)
library(sp)
library(splancs)
library(spdep)
```

```r
library(SpatialEpi)
library(RColorBrewer)
library(ggplot2)
library(ggridges)
library(INLA)
library(SUMMER)
library(survey)

## Problem 1
data(BRFSS)
data(KingCounty)
BRFSS = subset(BRFSS, !is.na(BRFSS$smoker1))
BRFSS = subset(BRFSS, !is.na(BRFSS$hracode))
nb.r = poly2nb(KingCounty, queen = F, row.names = KingCounty$HRA2010v2_)
mat = nb2mat(nb.r, style = "B", zero.policy = TRUE)
colnames(mat) = rownames(mat)

design_unweighted = svydesign(ids = ~1,
                    strata = ~strata,
                    data = BRFSS)
results_unweighted = svyby(~smoker1, ~hracode, design_unweighted, svymean)
# Confirm estimate and SE calculation
hra_counts = BRFSS %>% group_by(hracode) %>%
  summarize(count=n(),
            prop=sum(smoker1)/count)
results_unweighted = left_join(results_unweighted,hra_counts,by="hracode")
results_unweighted = results_unweighted %>% mutate(se_manual = sqrt(smoker1*(1-smoker1)/count))
head(results_unweighted)
# Map estimates
mapPlot(data = results_unweighted, geo = KingCounty, variables = c("smoker1"),
        labels = c("Sample Proportions"), by.data = "hracode", by.geo = "HRA2010v2_")
# Map SE
mapPlot(data = results_unweighted, geo = KingCounty, variables = c("se"),
        labels = c("Standard Errors"), by.data = "hracode", by.geo = "HRA2010v2_")

## Problem 2
design_weighted = svydesign(ids = ~1,
                    weights = ~rwt_llcp,
                    strata = ~strata,
                    data = BRFSS)
results_weighted = svyby(~smoker1, ~hracode, design_weighted, svymean)
head(results_weighted)
# Map estimates
mapPlot(data = results_weighted, geo = KingCounty, variables = c("smoker1"),
        labels = c("Weighted Estimates"), by.data = "hracode", by.geo = "HRA2010v2_")
# Map SE
mapPlot(data = results_weighted, geo = KingCounty, variables = c("se"),
        labels = c("Weighted Standard Errors"), by.data = "hracode", by.geo = "HRA2010v2_")

## Problem 3
unweight_weight_df = left_join(results_weighted,
                               results_unweighted,
                               by="hracode",
```

```
                                suffix = c("_w","_uw"))
# Plot estimates
ggplot(unweight_weight_df, aes(x = smoker1_uw, y = smoker1_w)) +
  geom_point() +
  labs(x = "Naive Estimate", y = "Weighted Estimate") +
  geom_abline(intercept = 0,slope = 1, color = "red") +
  xlim(0, 0.32) +
  ylim(0, 0.32)
# Plot SE
ggplot(unweight_weight_df, aes(x = se_uw, y = se_w)) +
  geom_point() +
  labs(x = "Naive Standard Error", y = "Weighted Standard Error") +
  geom_abline(intercept = 0,slope = 1, color = "red") +
  xlim(0, 0.05) +
  ylim(0, 0.1)

## Problem 4
smoothed = smoothSurvey(data = BRFSS, geo = KingCounty, Amat = mat,
                        responseType = "binary", responseVar = "smoker1",
                        strataVar = NULL, weightVar = NULL, regionVar = "hracode",
                        clusterVar = NULL, CI = 0.95)
smooth_post = smoothed$smooth
smooth_post = smooth_post %>% mutate(sd=sqrt(var))
# Map posterior medians
mapPlot(data = smooth_post, geo = KingCounty, variables = c("median"),
        labels = c("Posterior Medians"), by.data = "region", by.geo = "HRA2010v2_")
# Map posterior standard deviations
mapPlot(data = smooth_post, geo = KingCounty, variables = c("sd"),
        labels = c("Posterior Standard Deviations"), by.data = "region", by.geo = "HRA2010v2_")

## Problem 5
unweight_smooth_df = left_join(results_unweighted,
                               smooth_post,
                               by=c("hracode"="region"),
                               suffix = c("_naive","_smooth"))
# Plot estimates
ggplot(unweight_smooth_df, aes(x = smoker1, y = median)) +
  geom_point() +
  labs(x = "Naive Estimate", y = "Smoothed Estimate") +
  geom_abline(intercept = 0,slope = 1, color = "red") +
  xlim(0, 0.25) +
  ylim(0, 0.25)
# Plot SE
ggplot(unweight_smooth_df, aes(x = se, y = sd)) +
  geom_point() +
  labs(x = "Naive SE", y = "Smoothed Posterior SD") +
  geom_abline(intercept = 0,slope = 1, color = "red") +
  xlim(0, 0.05) +
  ylim(0, 0.05)

## Problem 6
smoothed_weight = smoothSurvey(data = BRFSS, geo = KingCounty, Amat = mat,
                               responseType = "binary",responseVar = "smoker1",
```

```
                                strataVar = "strata", weightVar = "rwt_llcp",
                                regionVar = "hracode", clusterVar = "~1", CI = 0.95)
smooth_weight_post = smoothed_weight$smooth
smooth_weight_post = smooth_weight_post %>% mutate(sd=sqrt(var))
# Map posterior medians
mapPlot(data = smooth_weight_post, geo = KingCounty, variables = c("median"),
        labels = c("Posterior Medians"), by.data = "region", by.geo = "HRA2010v2_")
# Map posterior standard deviations
mapPlot(data = smooth_weight_post, geo = KingCounty, variables = c("sd"),
        labels = c("Posterior Standard Deviations"), by.data = "region", by.geo = "HRA2010v2_")

## Problem 7
weight_smoothweight_df = left_join(results_weighted,
                                smooth_weight_post,
                                by=c("hracode"="region"),
                                suffix = c("_w","_sw"))
# Plot estimates
ggplot(weight_smoothweight_df, aes(x = smoker1, y = median)) +
  geom_point() +
  labs(x = "Weighted Estimate", y = "Smoothed Weighted Estimate") +
  geom_abline(intercept = 0,slope = 1, color = "red") +
  xlim(0, 0.3) +
  ylim(0, 0.3)
# Plot SE
ggplot(weight_smoothweight_df, aes(x = se, y = sd)) +
  geom_point() +
  labs(x = "Weighted SE", y = "Smoothed Weighted Posterior SD") +
  geom_abline(intercept = 0,slope = 1, color = "red") +
  xlim(0, 0.07) +
  ylim(0, 0.07)

## Problem 8
weight_smoothweight_df %>% arrange(median) %>% head(1)
weight_smoothweight_df %>% arrange(-median) %>% head(1)
weight_smoothweight_df %>% arrange(sd) %>% head(1)
weight_smoothweight_df %>% arrange(-sd)
```