IDENTIFYING HIGH SUICIDE RATES AND POTENTIAL CAUSES IN COLORADO COUNTIES

Emma Collins
Math 6384 – Spatial and Functional Analysis
Fall 2019
Dr. Joshua French

Introduction

In 2017, suicide was the tenth leading cause of death for all Americans, fourteen individuals per hundred thousand. In the same year, Colorado was ranked eleventh for highest suicide rate out of all fifty states, over twenty individuals per hundred thousand (American Foundation for Suicide Prevention, 2019). Since Colorado's suicide rate is higher than the national average, we aim to detect counties in Colorado with unusually high rates of suicide and identify potential factors that may contribute to these high rates. Policy derived from our conclusions will be recommended to decrease Colorado's high suicide rate and improve mental health care.

The observational data used in the following analysis is collected from two sources. The counts of suicide are collected from the Colorado Department of Public Health and Environment between and including 2010 and 2018; if any county has less than three suicides, counts are suppressed to 0 for privacy reasons. There are 40 suppressed counts over the eight year period not included in this analysis. County demographic information is collected from the U.S. Census Bureau from 2010.

Methods

Three spatial tests will be used throughout the analysis to detect individual clusters of counties with high suicide rates using various assessments of significance. These tests are performed under the Constant Risk Hypothesis (CRH) on areal data. The CRH assumes all individuals during the study period have the same risk of event, regardless of location, where we expect more events in higher populated areas (Waller & Gotway, 2004). All methods assume the risk of suicide is the same everywhere and we expect highly populated areas, such as Denver, to have more suicides than more sparsely populated counties.

The first test is the Besag-Newell. This test is performed by designating a specific number of cases, c^*, and determining how many counties must be included to reach c^* number of cases by starting at the centroid of each county and expanding outward in a circle. This is performed for each county where significance is assessed based on the probability of observing c^* cases in fewer counties than what is expected under the CRH.

The second test is Turnbull et al.'s Cluster Evaluation Permutation Procedure (CEPP). This method depends on a specified number of persons at risk or the total populations, n^*. Similar to the Besag-Newell, the CEPP starts at each county centroid and expands outward in a circle, encompassing more counties until n^* persons are contained. This is performed for each county where significance is assessed by determining if there are an excess number of suicides in the regions of counties than what we would expect to see under the CRH.

The last test is Kulldorf's Spatial Scan test, which can be viewed as a more generalized test of the previous two. The goal is to identify the regions of counties least compatible with the null hypothesis of no unusually high suicide rates in any counties or groups of counties. This is done by examining the number of suicides and total populations of each county individually and then expanding outward until a predetermined population upper bound is reached.

All tests will be performed with a significance level of 0.05.

These tests are performed assuming the CRH; everyone has the same risk of suicide regardless of location. However, we are trying to find factors that contribute to higher rate of suicide and want to show risk is *different* depending on location. This will be done by creating a

Generalized Linear Model (GLM) using various county demographics to predict suicide counts and then using the predictions in the previous tests as expected suicide rate for each individual county as opposed to every county having the same risk. If the tests conclude there are no clusters of counties with high suicide rates, then we can conclude counties with certain demographics also exhibit higher suicide rates.

We will use a Poisson GLM. A Poisson GLM is a generalization of a linear regression. Instead of predicting the average response, Y, given a number of predictors, X_i, shown as:

$$Y = \beta_0 + \sum_{i=1}^{n} \beta_i X_i,$$

we transform our response Y using the logarithmic link function. Thus, our new model is:

$$\log Y = \beta_0 + \sum_{i=1}^{n} \beta_i X_i \quad \Leftrightarrow \quad \mathrm{Y} = \exp\{\beta_0 + \sum_{i=1}^{n} \beta_i X_i\}.$$

Poisson GLM's are commonly used to predict a response of counts, or data that follows a Poisson probability distribution.

Analysis and Results

First, we look at initial plots and summary statistics of our data. In addition to suicide counts between and including 2010 and 2018, we also have the following demographics for every Colorado county in 2010: the proportion of the county considered rural, the proportion of the county self-identified as female, the proportion of the county with a high school diploma, the proportion of the county with a bachelors degree, the proportion of the county considered in poverty, the proportion of county residents that own their house, the county income per capita, and the total population of each county. These demographics will be used to fit the Poisson GLM.

Summary statistics of the demographics give a quick insight into the diversity of Colorado. There is at least one county considered completely rural and another county not considered rural at all. The percentage of residents with high school diplomas range from 72% to 98%, while the percentage of residents with bachelor's degrees only ranges from 12% to 60%. The county with the smallest population has less than 700 residents, while the most populated as over 600,000, with a median population of about 15,000.

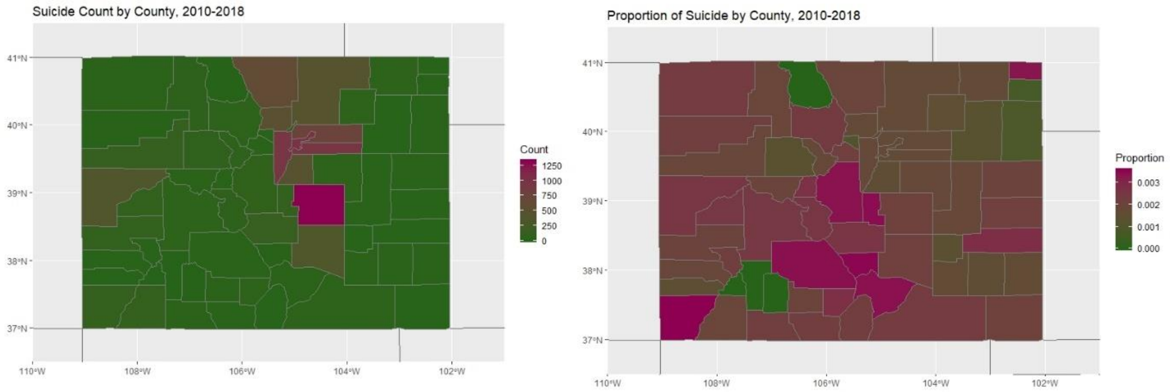

Figure 1. Suicide counts per county from 2010-2018, and proportion of suicide count per county to county population from 2010-2018.

The above figure shows the difference between total suicide count of each individual county and the proportion of suicide count to population. We would expect there to be the most suicides in the Denver metropolitan area, but when we consider population the same area appears to have a rate near the middle, while counties in the southern central area have a higher rate. Suicide counts range from 0 to over 1250, while our suicide rate ranges from 0% to over 0.3%.

First, we perform the Besag-Newell test. We predetermine our c^* to be 20, 50, and 100 by summary statistics of our suicide counts; the first quartile is 10 suicides, the median is 33, and the third quartile is 88.

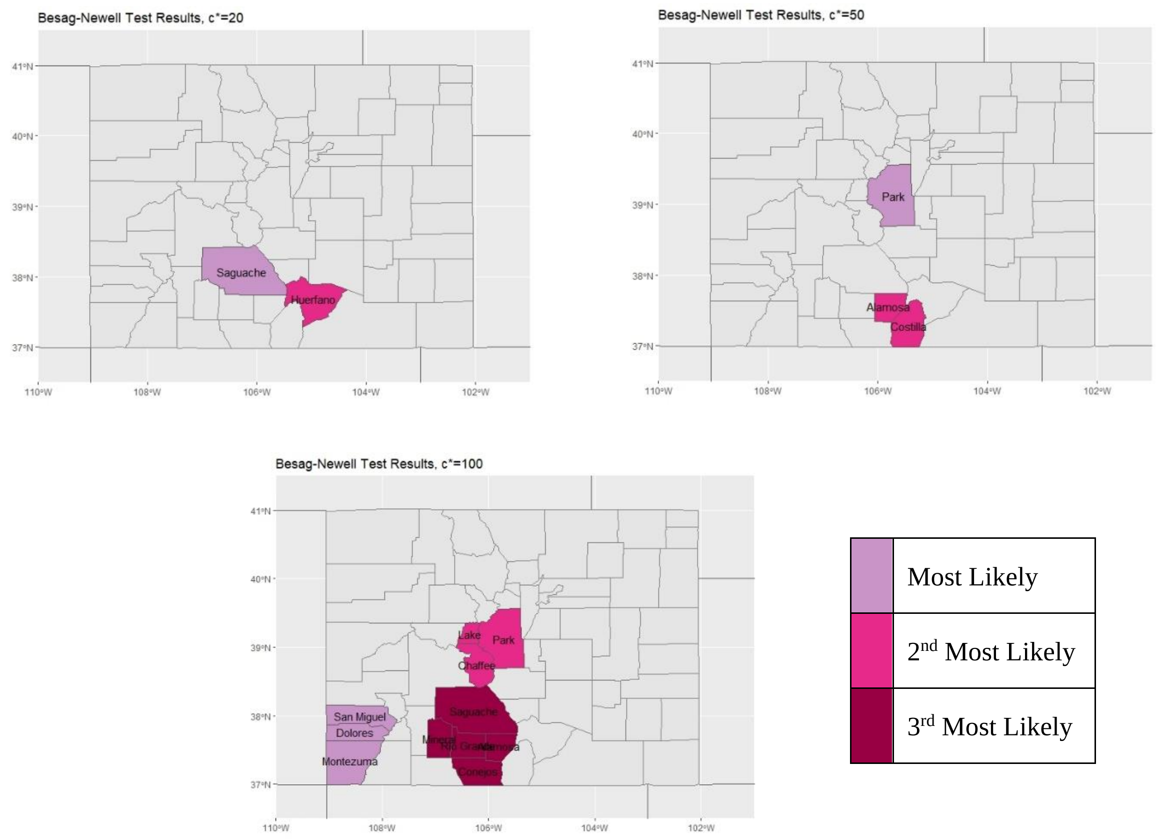

Figure 2. The most likely clusters returned from the Besag-Newell test for c=20, c*=50, and c*=100, under the CRH for suicide rates between 2010 and 2018.*

Each plot above shows the two or three most likely clusters at the various c^* levels. The first two tests only had two clusters within our significance level. The $c^* = 50$ and $c^* = 100$ tests have some overlapping results; the two most likely clusters for the $c^* = 50$ test are enveloped by the second and third most likely cluster of the $c^* = 100$ test.

The predetermined population levels for the CEPP test n^* are 20,000, 100,000, and 200,000, again chosen by considering the quantiles of the total populations.

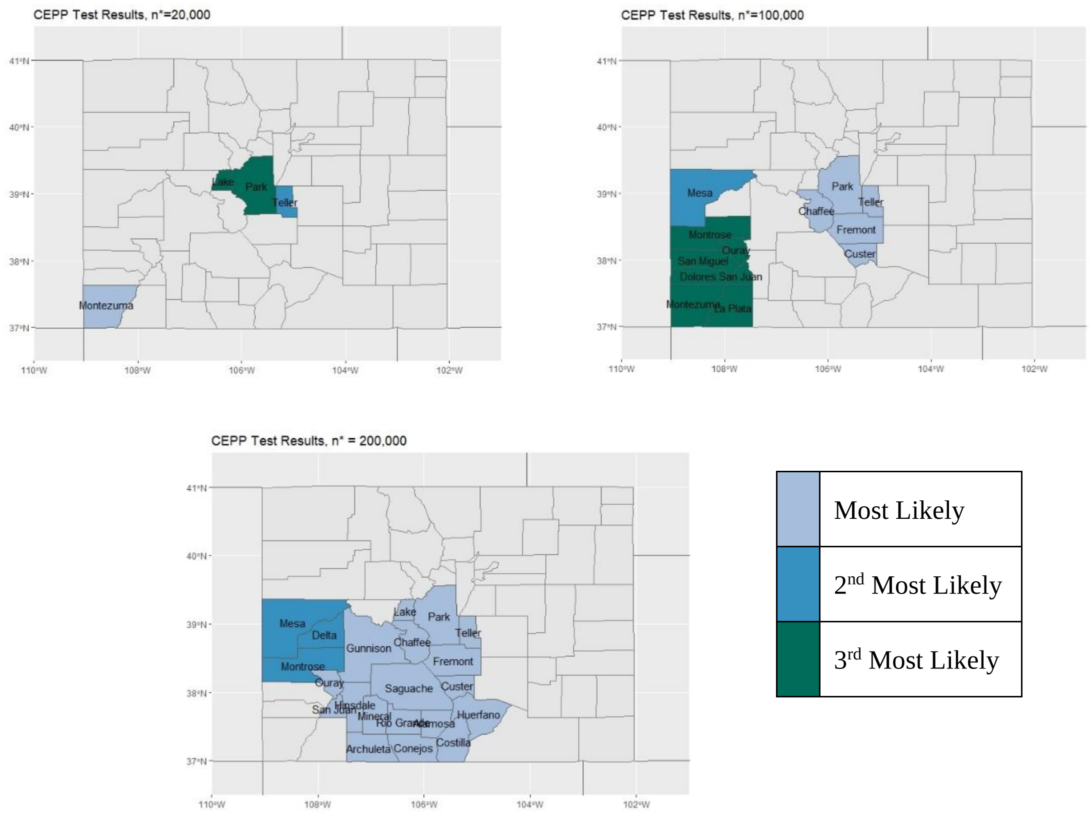

Figure 3. The most likely clusters returned from the CEPP for n=20,000, n*=100,000, and n*=200,000, under the CRH for suicide rates between 2010 and 2018.*

Similar to the Besag-Newell test, there are overlaps of the three most likely clusters for each value of n^*. The two most likely clusters for $n^* = 100{,}000$ grow to include more counties in the $n^* = 200{,}000$ test.

The spatial scan test has similar results with population percentage upper bounds at 5%, 15%, and 25%. To not reach clustering that covered half the state, we limit our highest upper bound to 25%, with 10% decreasing increments for the other two bounds.

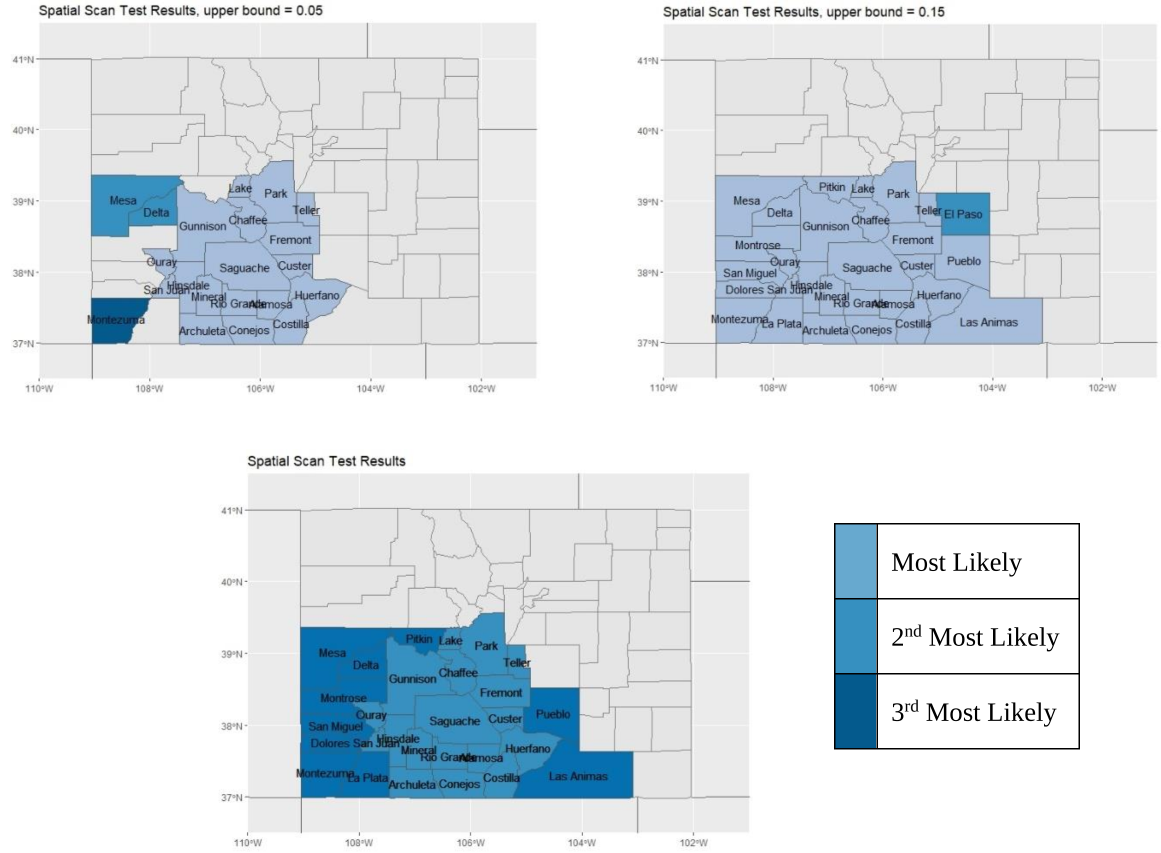

Figure 4. The most likely clusters returned from the Spatial Scan test for population upper bounds of 0.05, 0.15, and 0.25, under the CRH for suicide rates between 2010 and 2018.

The spatial scan test with the lowest population upper bound has three most likely clusters while the other two tests have only two. However, as the population upper bound increases for each test, so do the size of the most likely clusters.

Next, we fit the Poisson GLM. After removing the proportion of county residents with a bachelor's degree to avoid collinearity, we selected a model using both AIC and BIC stepwise selection, in both directions. The AIC and BIC selection criteria chose the same model. The final model predicts the suicide counts between 2010 and 2018 with income per capita, proportion of county with a high school diploma, the proportion of county considered rural, the proportion of county residents in poverty, and the proportion of residents that own their own home. The model passes diagnostic tests to check for normal errors, homogeneous variance, and outliers or influential observations.

Now we re-test the CEPP and Spatial Scan methods using the same test parameters as before. We do not retest the Besag-Newell since there is no easy implementation to include expected counts at this time.

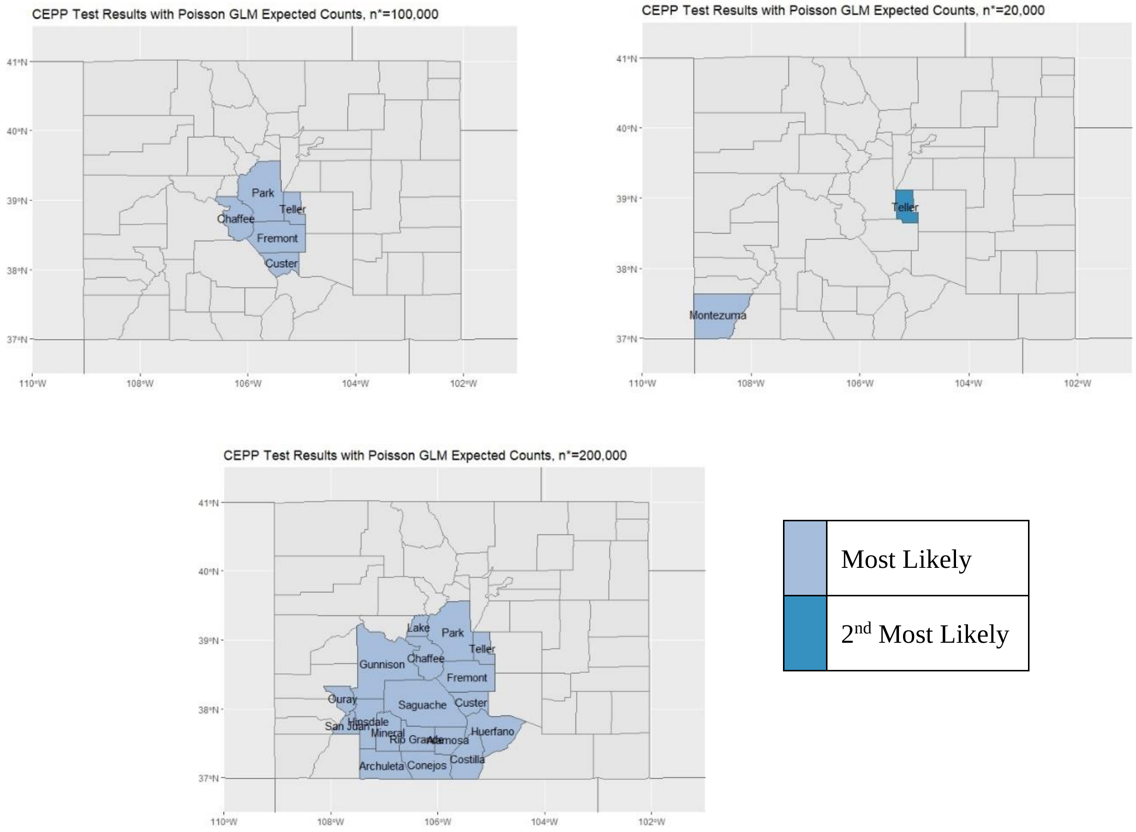

Figure 5. The most likely clusters returned from the CEPP for n=20,000, n*=100,000, and n*=200,000, using expected counts per county from the Poisson GLM for suicide rates between 2010 and 2018.*

The CEPP results with expected counts are very similar to before, only with smaller clusters in each test level. There are also only one likely cluster for $n^* = 20{,}000$ and $n^* = 200{,}000$, compared to three and two likely clusters under the CRH, respectively. There are only two likely clusters for $n^* = 100{,}000$ when there were three likely clusters under CRH.

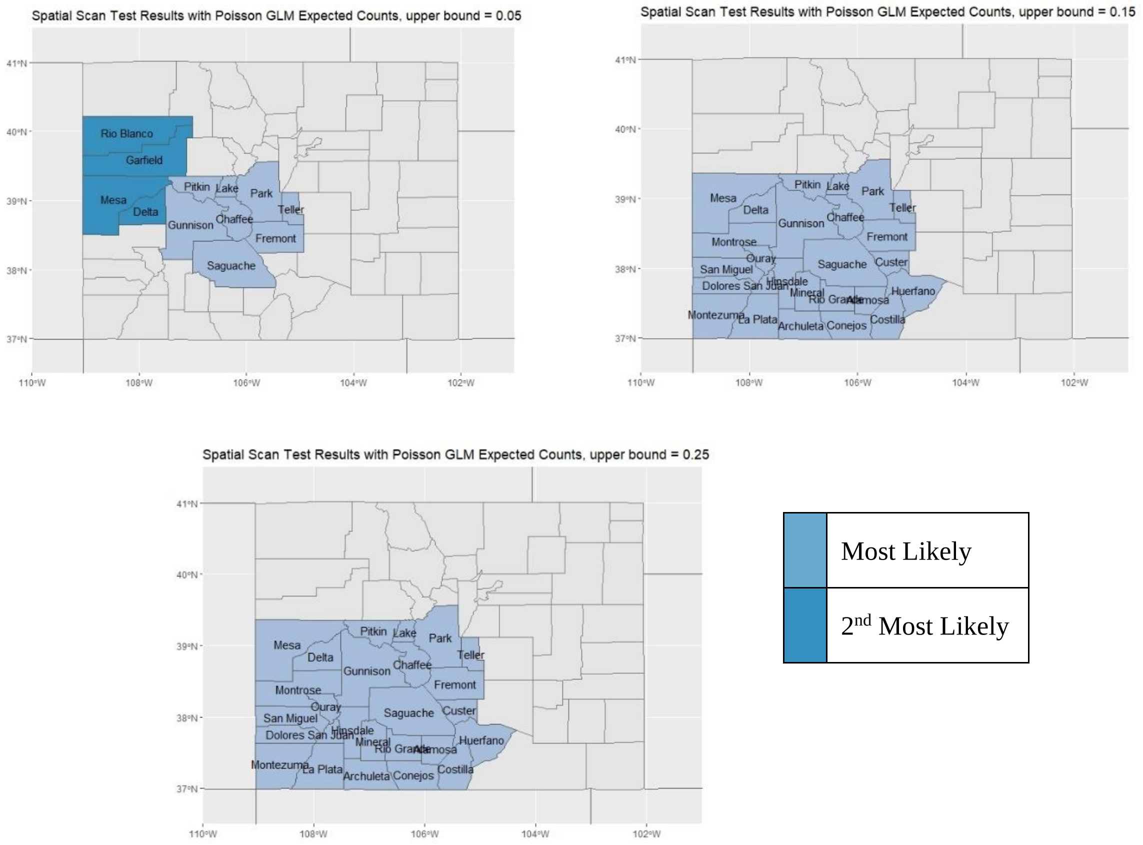

Figure 6. The most likely clusters returned from the Spatial Scan test for population upper bounds of 0.05, 0.15, and 0.25, using expected counts per county from the Poisson GLM for suicide rates between 2010 and 2018.

The results for the Spatial Scan while considering the expected counts from the Poisson GLM also gives similar results. Arguably, the most dissimilar is the test with the population upper bound at 5%. The test under the CRH returns three likely clusters while the test with expected counts only returns two and the counties included in the test under the CRH returns more south-central counties while the test with expected counts has north-western counties. The tests with population upper bounds of 15% and 25% show very similar results. All but one or two counties are incorporated in the south-western area of the state than shown in the tests under the CRH.

It is hard to evaluate the many plots, so we compare the top cluster under the CRH and using the expected counts for each test.

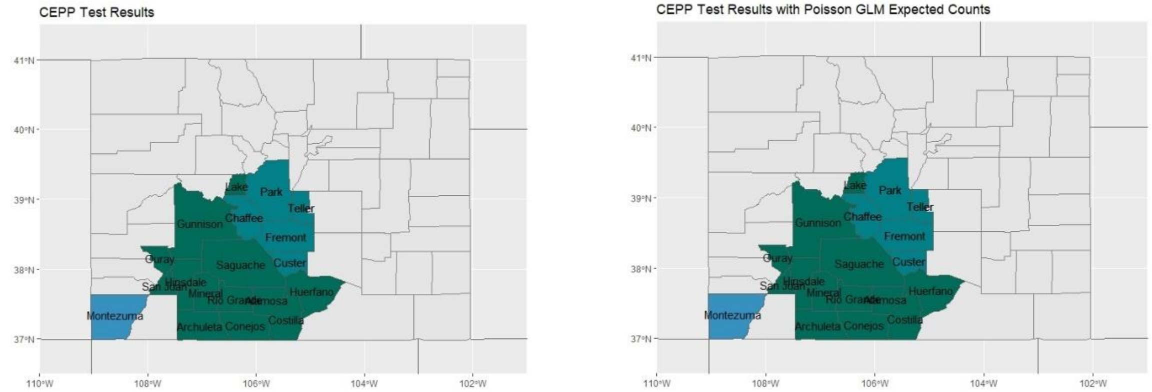

Figure 7. The most likely cluster returned for each level of n of the CEPP under the CRH (left) and using Poisson expected counts (right).*

The above plots show the most likely cluster for each test level for the CEPP under the CRH and using the expected counts from the Poisson GLM. Both tests yielded the exact same results for the most likely cluster.

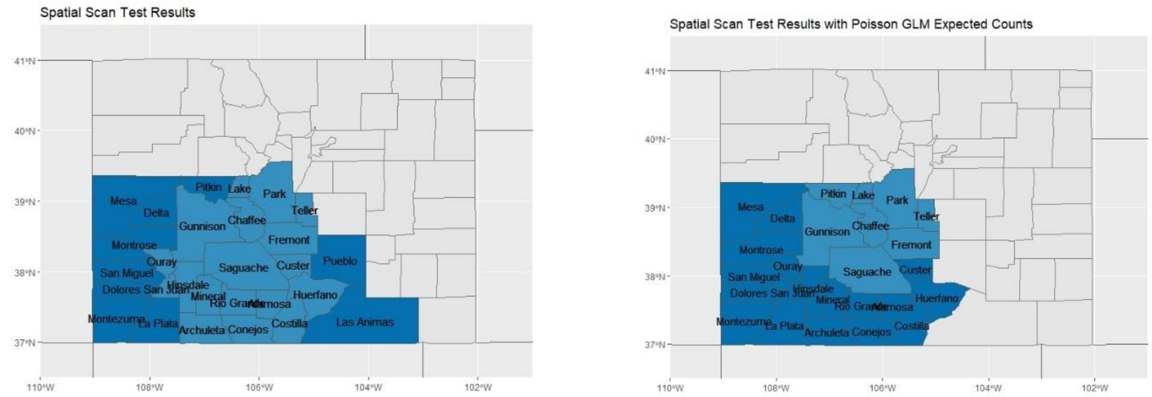

Figure 8. The most likely cluster returned for each population upper bound of the Spatial Scan test under the CRH (left) and using Poisson expected counts (right).

The above plots show the most likely clusters for each population upper bound for the Spatial Scan test under the CRH and using the Poisson GLM expected counts. The results are almost similar; the test with the expected counts returned two counties less than the test under the CRH. Otherwise, the same counties are the most likely clusters.

Conclusions and Policy Proposal

From the three different tests performed under the CRH, we can make similar conclusions. The Besag-Newell informs us there is convincing evidence that there is at least one cluster with the specified c^* that is significantly more compact, regarding the number of counties, than what is expected under the CRH. The CEPP test allows us to conclude there is significant evidence there is at least one window with more suicides than what is expected from an at-risk population of n^* under the CRH. In each application of the Spatial Scan test, we conclude the most likely cluster is significantly more unusual than what we would expect under the CRH. Thus, for all the tests, we can conclude that the clusters returned exhibit higher suicide rates than what is expected the CRH, even with their different methods of assessing significance.

We had similar results after using the expected counts from the Poisson GLM for each county in place of constant risk. While there are slight differences between the tests under constant risk and tests with different expected counts, the most likely cluster for each test returned near identical, if not completely identical results. Therefore, we cannot identify with confidence the proportion of county considered rural, proportion of residents with a high school diploma, proportion of residents that own their home, and income per capita simultaneously correspond to higher suicide rates. We also tried fitting two simple Poisson GLM, predicting suicide counts from proportion of county considered rural and the proportion of residents with high school diploma’s individually, then redid the tests with predictions from each model as the expected suicide counts per county. None of the tests with either model produced significantly different results.

Even though the test results are subject to change based on our parameter levels (c^*, n^*, and the population upper bound), multiple levels were considered and tried for each test before deciding on which values to use in the analysis. Every test returned significant clusters within the same south-western area of the state. The results may differ slightly test to test, but in practical terms we can draw the same conclusion: the southwestern counties of Colorado show evidence of clusters of suicides between 2010 and 2018 beyond what we would expect to see under the CRH or expected counts derived from a Poisson GLM.

While it would follow to have more affordable health care, so individuals can seek treatment without fear of hefty medical bills as a deterrent or potentially going into debt after seeking care, there is an even bigger issue we must address first. Colorado, and the rest of the United States, have a mental health care professional shortage. In 2016, over half of U.S. counties had no practicing psychiatrist and in a recent 3-year period, there was a 42% increase of individuals seeking treatment at emergency rooms (Weiner, 2018).

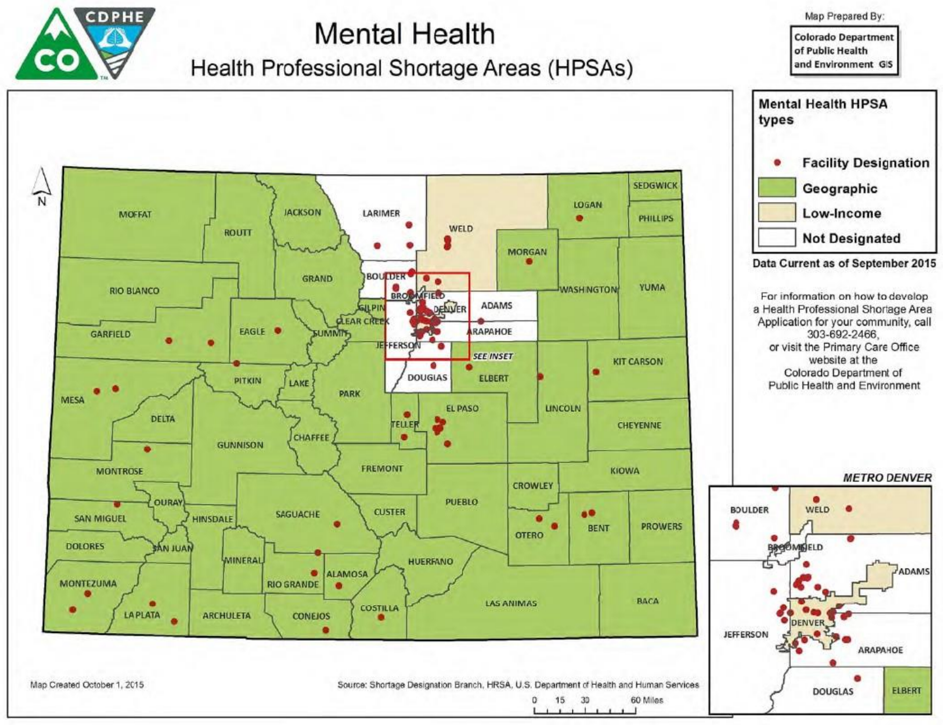

Figure 9. A map of Colorado where counties are categorized as having a mental health professional shortage (green), low income (beige), or not designated (white).

The green counties in the map above have a shortage of mental health professionals, such as psychologists and psychiatrists. The red dots indicate a mental health facility, which when outside of the Denver metropolitan area are far and few in between. When faced with mental crisis, individuals are unable to get the help they need in a reasonable amount of time, especially in rural areas. This can be addressed in two ways.

First, more universities need to increase recruiting efforts for psychiatry programs. Mental health related problems have only increased in recent years thus the psychiatrist population needs to increase to meet the rising demands. After increasing recruiting for their psychiatry program, The University of Nebraska Medical College has seen enrollment more than double in the program between 2013 and 2018 (Weiner, 2018).

Practicing mental health professionals should also consider the use of telepsychiatry. Telepsychiatry is the process of web-based video-chatting with a professional, instead of a face-to-face appointment. Improving mental health does not require a bone to be set or a shot in the arm and any necessary prescriptions can be filed electronically. This would increase the accessibility to mental health care to individuals living in rural areas.

Overall, Colorado is facing higher rates of suicide than most of the country, especially in its south-western counties, shown through various spatial scan tests. Universities in Colorado, and the rest of the country, need to improve recruiting efforts to increase the population of mental health professionals and currently practicing professionals should consider the use of telepsychiatry to increase availability to rural areas.

References

American Foundation for Suicide Prevention. (2019, April 16). *Suicide Statistics*. Retrieved from https://afsp.org/about-suicide/suicide-statistics/

State of Colorado . (2019, August 23). *Colorado Health Information Dataset*. Retrieved from Department of Public Health and Environment: https://www.colorado.gov/pacific/cdphe/data

State of Colorado. (2019, June 22). *Health Professional Shortage Area maps and data*. Retrieved from Department of Public Health and Environment: https://www.colorado.gov/pacific/cdphe/shortage-area-maps-and-data

U.S. Census Bureau. (2011). *American Community Survey Demographic and Housing Estimates*. Retrieved from https://data.census.gov/cedsci/

Waller, L. A., & Gotway, C. A. (2004). *Applied Spatial Statistics for Public Health Data.* John Wiley & Sons, Inc.

Weiner, S. (2018, February 13). *Addressing the escalating psychiatrist shortage*. Retrieved from Association of American Medical Colleges: https://www.aamc.org/news-insights/addressing-escalating-psychiatrist-shortage

Code Appendix

```
# packages for spatial analysis
library("smerc")

# packages for glm
library("car")
library("StepReg")
library("leaps")
library("bestglm")

# packages for plots
library("ggplot2")
library("sf")
library("rnaturalearth")
library("rnaturalearthdata")
library("maps")
library("rgeos")
library("lwgeom")
library("geosphere")
library("RColorBrewer")

set.seed(23)

### Initial data ###
sdata <- read.csv("C:/Users/Owner/Desktop/Math 
6384/Project/COCountyData.csv")
name <- sdata$COUNTY

### Plot setup ###
display.brewer.all(type = "all", colorblindFriendly = TRUE)
mygrad = brewer.pal(9, "PuRd")
mygrad1 = brewer.pal(9, "PuBuGn")
mygrad2 = brewer.pal(9, "PuBu")
mycol = brewer.pal(8, "Dark2")
mygrad3 <- brewer.pal(11, "PiYG")

world <- ne_countries(scale = "medium", returnclass = "sf")

states <- st_as_sf(map("state", plot = FALSE, fill = TRUE))

counties <- st_as_sf(map("county", plot = FALSE, fill = TRUE))
counties <- subset(counties, grepl("colorado,", counties$ID))
counties <- cbind(counties, name, centroid(as_Spatial(counties$geometry)))

### Initial Plot of Suicide Count/Proportion ###
#Suicide Count 2010-2018
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5), aes(fill = 
as.numeric(sdata$S1018))) +
  # geom_point(data = coords, mapping = aes(x = counties.X1, y = 
counties.X2)) + remove for centroids
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  scale_fill_continuous(low = mygrad3[11], high = mygrad3[1], name = "Count")
+
  labs(title = "Suicide Count by County, 2010-2018", x = "", y = "")

#Suicide Proportion 2010-2018
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5), aes(fill =
(as.numeric(sdata$S1018)/as.numeric(sdata$Tpop)))) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  scale_fill_continuous(low = mygrad3[11], high = mygrad3[1], name =
"Proportion") +
  ggtitle("Proportion of Suicide by County, 2010-2018")



########## CLUSTERING TESTS ############

# look at summary statistics to identify potential limits (c*, n*)
summary(sdata)

#S1018, 1st Q: 10
#       Mean: 146
#       Median: 33
#       3rd Q: 85

# Population 1st Q: 5734
#            Mean: 78581
#            Median: 15084
#            3rd Q: 42663


# Prep for cluster tests
coords <- data.frame(counties$X1, counties$X2)
pop <- as.numeric(sdata$Tpop)
cases <- as.numeric(sdata$S1018)

# Besag-Newell
# addresses variability in local incidence proption by fixing lower bound for
number of
#   cases in each window by c*.
# H0: Most compact window w/ c* cases (centered at centroid) is not sig. more
compact than what
#   is expected under CRH

bnl <- bn.test(coords = coords, cases = cases, pop = pop, cstar = 20, alpha =
0.05, longlat = TRUE)
bnl
bnl_c <- counties[sort(bnl$clusters[[1]]$locids),]
bnl_c2 <- counties[sort(bnl$clusters[[2]]$locids),]
#bnl_c3 <- counties[sort(bnl$clusters[[3]]$locids),]

#BN, c* = 20, results
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
 #geom_sf(data = bnl_c3, fill = mygrad[8]) +
  geom_sf(data = bnl_c2, fill = mygrad[6]) +
  geom_sf(data = bnl_c, fill = mygrad[4]) +
  geom_text(data = bnl_c, aes(x=X1, y=X2, label=bnl_c$name)) +
  geom_text(data = bnl_c2, aes(x=X1, y=X2, label=bnl_c2$name)) +
 #geom_text(data = bnl_c3, aes(x=X1, y=X2, label=bnl_c3$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Besag-Newell Test Results, c*=20", x = "", y = "")

bnm <- bn.test(coords = coords, cases = cases, pop = pop, cstar = 50, alpha =
0.05, longlat = TRUE)
bnm
bnm_c <- counties[sort(bnm$clusters[[1]]$locids),]
bnm_c2 <- counties[sort(bnm$clusters[[2]]$locids),]
#bnm_c3 <- counties[sort(bnm$clusters[[3]]$locids),]

#BN, c* = 50, results
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  #geom_sf(data = bnm_c3, fill = mygrad[8]) +
  geom_sf(data = bnm_c2, fill = mygrad[6]) +
  geom_sf(data = bnm_c, fill = mygrad[4]) +
  geom_text(data = bnm_c, aes(x=X1, y=X2, label=bnm_c$name)) +
  geom_text(data = bnm_c2, aes(x=X1, y=X2, label=bnm_c2$name)) +
  #geom_text(data = bnm_c3, aes(x=X1, y=X2, label=bnm_c3$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Besag-Newell Test Results, c*=50", x = "", y = "")

bnh <- bn.test(coords = coords, cases = cases, pop = pop, cstar = 100, alpha
= 0.05, longlat = TRUE)
bnh
bnh_c <- counties[sort(bnh$clusters[[1]]$locids),]
bnh_c2 <- counties[sort(bnh$clusters[[2]]$locids),]
bnh_c3 <- counties[sort(bnh$clusters[[3]]$locids),]

#BN, c* = 100, results
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  geom_sf(data = bnh_c3, fill = mygrad[8]) +
  geom_sf(data = bnh_c2, fill = mygrad[6]) +
  geom_sf(data = bnh_c, fill = mygrad[4]) +
  geom_text(data = bnh_c, aes(x=X1, y=X2, label=bnh_c$name)) +
  geom_text(data = bnh_c2, aes(x=X1, y=X2, label=bnh_c2$name)) +
  geom_text(data = bnh_c3, aes(x=X1, y=X2, label=bnh_c3$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Besag-Newell Test Results, c*=100", x = "", y = "")

# Plot of most likely cluster from each BN test
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  geom_sf(data = bnh_c, fill = mygrad[8]) +
  geom_sf(data = bnm_c, fill = mygrad[7]) +
  geom_sf(data = bnl_c, fill = mygrad[6]) +
  geom_text(data = bnl_c, aes(x=X1, y=X2, label=bnl_c$name)) +
  geom_text(data = bnm_c, aes(x=X1, y=X2, label=bnm_c$name)) +
  geom_text(data = bnh_c, aes(x=X1, y=X2, label=bnh_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Besag-Newell Test Results", x = "", y = "")

# CEPP
# Chose fixed persons at risk, n*, expand window until n* is reached
# H0: No window with n* persons at risk has sig. more cases than what we
expect under CRH

ceppl <- cepp.test(coords = coords, cases = cases, pop = pop, nstar = 20000,
alpha = 0.05, longlat = TRUE)
ceppl
ceppl_c <- counties[sort(ceppl$clusters[[1]]$locids),]
ceppl_c2 <- counties[sort(ceppl$clusters[[2]]$locids),]
ceppl_c3 <- counties[sort(ceppl$clusters[[3]]$locids),]

#CEPP, n* = 20000, results
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  geom_sf(data = ceppl_c3, fill = mygrad1[8]) +
  geom_sf(data = ceppl_c2, fill = mygrad1[6]) +
  geom_sf(data = ceppl_c, fill = mygrad1[4]) +
  geom_text(data = ceppl_c3, aes(x=X1, y=X2, label=ceppl_c3$name)) +
  geom_text(data = ceppl_c2, aes(x=X1, y=X2, label=ceppl_c2$name)) +
  geom_text(data = ceppl_c, aes(x=X1, y=X2, label=ceppl_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "CEPP Test Results, n*=20,000", x = "", y = "")

ceppm <- cepp.test(coords = coords, cases = cases, pop = pop, nstar = 100000,
alpha = 0.05, longlat = TRUE)
ceppm
ceppm_c <- counties[sort(ceppm$clusters[[1]]$locids),]
ceppm_c2 <- counties[sort(ceppm$clusters[[2]]$locids),]
ceppm_c3 <- counties[sort(ceppm$clusters[[3]]$locids),]

#CEPP, n* = 100000, results
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  geom_sf(data = ceppm_c3, fill = mygrad1[8]) +
  geom_sf(data = ceppm_c2, fill = mygrad1[6]) +
  geom_sf(data = ceppm_c, fill = mygrad1[4]) +
  geom_text(data = ceppm_c3, aes(x=X1, y=X2, label=ceppm_c3$name)) +
  geom_text(data = ceppm_c2, aes(x=X1, y=X2, label=ceppm_c2$name)) +
  geom_text(data = ceppm_c, aes(x=X1, y=X2, label=ceppm_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "CEPP Test Results, n*=100,000", x = "", y = "")
```

```
cepph <- cepp.test(coords = coords, cases = cases, pop = pop, nstar = 200000,
alpha = 0.05, longlat = TRUE)
cepph
cepph_c <- counties[sort(cepph$clusters[[1]]$locids),]
cepph_c2 <- counties[sort(cepph$clusters[[2]]$locids),]
#cepph_c3 <- counties[sort(cepph$clusters[[3]]$locids),]

#CEPP, n* = 200000, results
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  #geom_sf(data = cepph_c3, fill = mygrad1[8]) +
  geom_sf(data = cepph_c2, fill = mygrad1[6]) +
  geom_sf(data = cepph_c, fill = mygrad1[4]) +
  #geom_text(data = cepph_c3, aes(x=X1, y=X2, label=cepph_c3$name)) +
  geom_text(data = cepph_c2, aes(x=X1, y=X2, label=cepph_c2$name)) +
  geom_text(data = cepph_c, aes(x=X1, y=X2, label=cepph_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "CEPP Test Results, n* = 200,000", x = "", y = "")

# Most likely cluster from each CEPP Test
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  geom_sf(data = cepph_c, fill = mygrad1[8]) +
  geom_sf(data = ceppm_c, fill = mygrad1[7]) +
  geom_sf(data = ceppl_c, fill = mygrad1[6]) +
  geom_text(data = ceppl_c, aes(x=X1, y=X2, label=ceppl_c$name)) +
  geom_text(data = ceppm_c, aes(x=X1, y=X2, label=ceppm_c$name)) +
  geom_text(data = cepph_c, aes(x=X1, y=X2, label=cepph_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "CEPP Test Results", x = "", y = "")


# Spatial Scan Statistics
# Chooses population upper bound, identifies cluster(s) least compatible with
CRH
# H0: the most likely cluster is consistent with what is expected under CRH

ssl <- scan.test(coords = coords, cases = cases, pop = pop, ubpop = 0.05,
alpha = 0.05, longlat = TRUE)
ssl
ssl_c <- counties[sort(ssl$clusters[[1]]$locids),]
ssl_c2 <- counties[sort(ssl$clusters[[2]]$locids),]
ssl_c3 <- counties[sort(ssl$clusters[[3]]$locids),]

#SS, up=0.05
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  geom_sf(data = ssl_c3, fill = mygrad2[8]) +
  geom_sf(data = ssl_c2, fill = mygrad2[6]) +
  geom_sf(data = ssl_c, fill = mygrad2[4]) +
  geom_text(data = ssl_c3, aes(x=X1, y=X2, label=ssl_c3$name)) +
  geom_text(data = ssl_c2, aes(x=X1, y=X2, label=ssl_c2$name)) +
  geom_text(data = ssl_c, aes(x=X1, y=X2, label=ssl_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Spatial Scan Test Results, upper bound = 0.05", x = "", y =
"")

ssm <- scan.test(coords = coords, cases = cases, pop = pop, ubpop = 0.15,
alpha = 0.05, longlat = TRUE)
ssm
ssm_c <- counties[sort(ssm$clusters[[1]]$locids),]
ssm_c2 <- counties[sort(ssm$clusters[[2]]$locids),]
#ssm_c3 <- counties[sort(ssm$clusters[[3]]$locids),]

#SS, up=0.15
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  #geom_sf(data = ssm_c3, fill = mygrad2[8]) +
  geom_sf(data = ssm_c2, fill = mygrad2[6]) +
  geom_sf(data = ssm_c, fill = mygrad2[4]) +
  #geom_text(data = ssm_c3, aes(x=X1, y=X2, label=ssm_c3$name)) +
  geom_text(data = ssm_c2, aes(x=X1, y=X2, label=ssm_c2$name)) +
  geom_text(data = ssm_c, aes(x=X1, y=X2, label=ssm_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Spatial Scan Test Results, upper bound = 0.15", x = "", y =
"")

ssh <- scan.test(coords = coords, cases = cases, pop = pop, ubpop = 0.25,
alpha = 0.05, longlat = TRUE)
ssh
ssh_c <- counties[sort(ssh$clusters[[1]]$locids),]
ssh_c2 <- counties[sort(ssh$clusters[[2]]$locids),]
#ssh_c3 <- counties[sort(ssh$clusters[[3]]$locids),]

#SS, up=0.25
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  #geom_sf(data = ssh_c3, fill = mygrad2[8]) +
  geom_sf(data = ssh_c2, fill = mygrad2[6]) +
  geom_sf(data = ssh_c, fill = mygrad2[4]) +
  #geom_text(data = ssh_c3, aes(x=X1, y=X2, label=ssh_c3$name)) +
  geom_text(data = ssh_c2, aes(x=X1, y=X2, label=ssh_c2$name)) +
  geom_text(data = ssh_c, aes(x=X1, y=X2, label=ssh_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Spatial Scan Test Results, upper bound = 0.25", x = "", y =
"")

# most likely cluster from each test level
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  geom_sf(data = ssh_c, fill = mygrad2[8]) +
  geom_sf(data = ssm_c, fill = mygrad2[7]) +
  geom_sf(data = ssl_c, fill = mygrad2[6]) +
  geom_text(data = ssl_c, aes(x=X1, y=X2, label=ssl_c$name)) +
  geom_text(data = ssm_c, aes(x=X1, y=X2, label=ssm_c$name)) +
  geom_text(data = ssh_c, aes(x=X1, y=X2, label=ssh_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Spatial Scan Test Results", x = "", y = "")

########### BUILDING GLM #############

### collinearity check

# pairwise correlation (first column is county name)

cor(sdata[,2:11])
# obvious correlation between population and the two different suicide
counts.  Exclude Suicide 10

cor(sdata[, c(2, 4:11)])
# high correlation between suicide counts and population, remove population

cor(sdata[, 4:11])
# high-ish correlation between HS and Bachelors, but we'll leave it in for
now

# VIF
vifmod <- glm(S1018 ~ IPC + Rural + HS + Bachelors + Poverty + Female +
HomeOwn,
              data = sdata, family = "poisson", offset = log(Tpop))
vif(vifmod)
# Bachelors and HS again are pretty correlated.  Remove HS

vifmod2 <- update(vifmod, .~.-HS)
vif(vifmod2)
# IPC is pretty high, let's see what happens if we take the full model and
just remove IPC

vifmod3 <- update(vifmod, .~.-IPC)
vif(vifmod3)
# Bachelors and HS stay low.  Let's just try removing Bachelors just to see
if that makes any change

vifmod4 <- update(vifmod, .~.-Bachelors)
vif(vifmod4)
# All other VIF's are acceptable.  Let's remove bachelors, keep in IPC and HS

### Model Selection

# We'll use AIC stepwise selection, BIC stepwise selection

# fit full model first.

fullmod <- glm(S1018 ~ IPC + Rural + HS + Poverty + Female + HomeOwn, data =
sdata,
               family = "poisson", offset = log(Tpop))

AICmod <- step(fullmod, direction = "both", trace = FALSE)
summary(AICmod)
# model chosen by AIC is all predictors except female

BICmod <- step(fullmod, direction = "both", trace = FALSE, k = log(64))
summary(BICmod)
# model chosen by BIC is also all predictors except female

### Final model

finalmod <- BICmod
summary(finalmod)

### Model assumptions

# normality
qqnorm(residuals(finalmod), pch = 16)
qqline(residuals(finalmod))
#doesn't look great, but it's not awful bad

# homogeneous variance
plot(fitted(finalmod), residuals(finalmod), main="Resid V Fitted")
abline(h=0)
# The lower values are a little concerning with a slight downward pattern...

plot(fitted(finalmod), sqrt(abs(residuals(finalmod))), main="Sqrt(Resid) V
Fitted")

# outliers/infulential obsv
outlierTest(finalmod)
influencePlot(finalmod)
# potential observations, let's see what happens when they're removed

block <- row.names(sdata)

# remove 17
mod1 <- glm(S1018 ~ IPC + Rural + HS + Poverty + HomeOwn, data = sdata,
family = "poisson", offset = log(Tpop),
            subset = (block != 17))
compareCoefs(finalmod, mod1)
# some fair change in coefficients, but nothing awful (no sign changes)

# remove 19
mod2 <- glm(S1018 ~ IPC + Rural + HS + Poverty + HomeOwn, data = sdata,
family = "poisson", offset = log(Tpop),
            subset = (block != 19))
compareCoefs(finalmod, mod2)

# remove 21
mod3 <- glm(S1018 ~ IPC + Rural + HS + Poverty + HomeOwn, data = sdata,
family = "poisson", offset = log(Tpop),
            subset = (block != 21))
```

```
compareCoefs(finalmod, mod3)

# remove 40
mod4 <- glm(S1018 ~ IPC + Rural + HS + Poverty + HomeOwn, data = sdata,
family = "poisson", offset = log(Tpop),
            subset = (block != 40))
compareCoefs(finalmod, mod4)

# remove 50
mod5 <- glm(S1018 ~ IPC + Rural + HS + Poverty + HomeOwn, data = sdata,
family = "poisson", offset = log(Tpop),
            subset = (block != 50))
compareCoefs(finalmod, mod5)

# nothing warrents removal.

# Will use for expected number of cases for tests

glm_pred <- predict.glm(finalmod, type = "response")

############ POST-GLM CLUSTERING TESTS ##############

# CEPP
# Chose fixed persons at risk, n*, expand window until n* is reached
# H0: No window with n* persons at risk has sig. more cases than what we
expect under CRH

cepplp <- cepp.test(coords = coords, cases = cases, pop = pop, ex = glm_pred,
nstar = 20000, alpha = 0.05, longlat = TRUE)
cepplp
cepplp_c <- counties[sort(cepplp$clusters[[1]]$locids),]
cepplp_c2 <- counties[sort(cepplp$clusters[[2]]$locids),]
#cepplp_c3 <- counties[sort(cepplp$clusters[[3]]$locids),]

# CEPP, GLM Expected Counts, n*=20000
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  #geom_sf(data = cepplp_c3, fill = mygrad1[8]) +
  geom_sf(data = cepplp_c2, fill = mygrad1[6]) +
  geom_sf(data = cepplp_c, fill = mygrad1[4]) +
  #geom_text(data = cepplp_c3, aes(x=X1, y=X2, label=cepplp_c3$name)) +
  geom_text(data = cepplp_c2, aes(x=X1, y=X2, label=cepplp_c2$name)) +
  geom_text(data = cepplp_c, aes(x=X1, y=X2, label=cepplp_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "CEPP Test Results with Poisson GLM Expected Counts,
n*=20,000", x = "", y = "")


ceppmp <- cepp.test(coords = coords, cases = cases, pop = pop, ex = glm_pred,
nstar = 100000, alpha = 0.05, longlat = TRUE)
ceppmp
ceppmp_c <- counties[sort(ceppmp$clusters[[1]]$locids),]
#ceppmp_c2 <- counties[sort(ceppmp$clusters[[2]]$locids),]
```

```
#ceppmp_c3 <- counties[sort(ceppmp$clusters[[3]]$locids),]

# CEPP, GLM Expected Counts, n*=100000
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  #geom_sf(data = ceppmp_c3, fill = mygrad1[8]) +
  #geom_sf(data = ceppmp_c2, fill = mygrad1[6]) +
  geom_sf(data = ceppmp_c, fill = mygrad1[4]) +
  #geom_text(data = ceppmp_c3, aes(x=X1, y=X2, label=ceppmp_c3$name)) +
  #geom_text(data = ceppmp_c2, aes(x=X1, y=X2, label=ceppmp_c2$name)) +
  geom_text(data = ceppmp_c, aes(x=X1, y=X2, label=ceppmp_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "CEPP Test Results with Poisson GLM Expected Counts, 
n*=100,000", x = "", y = "")


cepphp <- cepp.test(coords = coords, cases = cases, pop = pop, ex = glm_pred, 
nstar = 200000, alpha = 0.05, longlat = TRUE)
cepphp
cepphp_c <- counties[sort(cepphp$clusters[[1]]$locids),]
#cepphp_c2 <- counties[sort(cepphp$clusters[[2]]$locids),]
#cepphp_c3 <- counties[sort(cepphp$clusters[[3]]$locids),]

# CEPP, GLM Expected Counts, n*=100000
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  #geom_sf(data = ceppmp_c3, fill = mygrad1[8]) +
  #geom_sf(data = ceppmp_c2, fill = mygrad1[6]) +
  geom_sf(data = cepphp_c, fill = mygrad1[4]) +
  #geom_text(data = ceppmp_c3, aes(x=X1, y=X2, label=ceppmp_c3$name)) +
  #geom_text(data = ceppmp_c2, aes(x=X1, y=X2, label=ceppmp_c2$name)) +
  geom_text(data = cepphp_c, aes(x=X1, y=X2, label=cepphp_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "CEPP Test Results with Poisson GLM Expected Counts, 
n*=200,000", x = "", y = "")


#Top clusters from each test level
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  geom_sf(data = cepphp_c, fill = mygrad1[8]) +
  geom_sf(data = cepplp_c, fill = mygrad1[6]) +
  geom_sf(data = ceppmp_c, fill = mygrad1[7]) +
  geom_text(data = cepplp_c, aes(x=X1, y=X2, label=cepplp_c$name)) +
  geom_text(data = ceppmp_c, aes(x=X1, y=X2, label=ceppmp_c$name)) +
  geom_text(data = cepphp_c, aes(x=X1, y=X2, label=cepphp_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "CEPP Test Results with Poisson GLM Expected Counts", x = "", 
y = "")

# Spatial Scan Statistics
# Chooses population upper bound, identifies cluster(s) least compatible with
CRH
# H0: the most likely cluster is consistent with what is expected under CRH

sslp <- scan.test(coords = coords, cases = cases, pop = pop, ex = glm_pred,
ubpop = 0.05, alpha = 0.05, longlat = TRUE)
sslp
sslp_c <- counties[sort(sslp$clusters[[1]]$locids),]
sslp_c2 <- counties[sort(sslp$clusters[[2]]$locids),]
#sslp_c3 <- counties[sort(sslp$clusters[[3]]$locids),]

# SS test with GLM counts, up = 0.05
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  #geom_sf(data = sslp_c3, fill = mygrad2[8]) +
  geom_sf(data = sslp_c2, fill = mygrad2[6]) +
  geom_sf(data = sslp_c, fill = mygrad2[4]) +
  #geom_text(data = sslp_c3, aes(x=X1, y=X2, label=sslp_c3$name)) +
  geom_text(data = sslp_c2, aes(x=X1, y=X2, label=sslp_c2$name)) +
  geom_text(data = sslp_c, aes(x=X1, y=X2, label=sslp_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Spatial Scan Test Results with Poisson GLM Expected Counts,
upper bound = 0.05", x = "", y = "")

ssmp <- scan.test(coords = coords, cases = cases, pop = pop, ex = glm_pred,
ubpop = 0.15, alpha = 0.05, longlat = TRUE)
ssmp
ssmp_c <- counties[sort(ssmp$clusters[[1]]$locids),]
#ssmp_c2 <- counties[sort(ssmp$clusters[[2]]$locids),]
#ssmp_c3 <- counties[sort(ssmp$clusters[[3]]$locids),]

# SS test with GLM counts, up = 0.15
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  #geom_sf(data = sslp_c3, fill = mygrad2[8]) +
  #geom_sf(data = sslp_c2, fill = mygrad2[6]) +
  geom_sf(data = ssmp_c, fill = mygrad2[4]) +
  #geom_text(data = sslp_c3, aes(x=X1, y=X2, label=sslp_c3$name)) +
  #geom_text(data = sslp_c2, aes(x=X1, y=X2, label=sslp_c2$name)) +
  geom_text(data = ssmp_c, aes(x=X1, y=X2, label=ssmp_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Spatial Scan Test Results with Poisson GLM Expected Counts,
upper bound = 0.15", x = "", y = "")

sshp <- scan.test(coords = coords, cases = cases, pop = pop, ex = glm_pred,
ubpop = 0.25, alpha = 0.05, longlat = TRUE)
sshp
sshp_c <- counties[sort(sshp$clusters[[1]]$locids),]
#sshp_c2 <- counties[sort(sshp$clusters[[2]]$locids),]
#sshp_c3 <- counties[sort(sshp$clusters[[3]]$locids),]

ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  #geom_sf(data = sslp_c3, fill = mygrad2[8]) +
  #geom_sf(data = sslp_c2, fill = mygrad2[6]) +
  geom_sf(data = sshp_c, fill = mygrad2[4]) +
  #geom_text(data = sslp_c3, aes(x=X1, y=X2, label=sslp_c3$name)) +
  #geom_text(data = sslp_c2, aes(x=X1, y=X2, label=sslp_c2$name)) +
  geom_text(data = sshp_c, aes(x=X1, y=X2, label=sshp_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Spatial Scan Test Results with Poisson GLM Expected Counts,
upper bound = 0.25", x = "", y = "")

# top clusters from each Spatial Scan Test
ggplot(data = world) +
  geom_sf(data = states, fill = NA) +
  geom_sf(data = counties, color = gray(.5)) +
  geom_sf(data = sshp_c, fill = mygrad2[8]) +
  geom_sf(data = ssmp_c, fill = mygrad2[7]) +
  geom_sf(data = sslp_c, fill = mygrad2[6]) +
  geom_text(data = sslp_c, aes(x=X1, y=X2, label=sslp_c$name)) +
  geom_text(data = ssmp_c, aes(x=X1, y=X2, label=ssmp_c$name)) +
  geom_text(data = sshp_c, aes(x=X1, y=X2, label=sshp_c$name)) +
  coord_sf(xlim = c(-110, -101), ylim = c(36.5, 41.5), expand = FALSE) +
  labs(title = "Spatial Scan Test Results with Poisson GLM Expected Counts",
x = "", y = "")


#### What if? #####

# Rural areas tend to have higher rates of suicide.  What happens if we fit
Pois GLM with rural only?

testglm <- glm(S1018 ~ Rural, data = sdata, family = "poisson", offset =
log(Tpop))
rural_pred <- predict(testglm, type = "response")

cepplr <- cepp.test(coords = coords, cases = cases, pop = pop, ex =
rural_pred, nstar = 20000, alpha = 0.05, longlat = TRUE)
cepplr
sort(cepplr$clusters[[1]]$locids)
sort(cepplp$clusters[[1]]$locids)

ceppmr <- cepp.test(coords = coords, cases = cases, pop = pop, ex =
rural_pred, nstar = 100000, alpha = 0.05, longlat = TRUE)
ceppmr
sort(ceppmr$clusters[[1]]$locids)
sort(ceppmp$clusters[[1]]$locids)

cepphr <- cepp.test(coords = coords, cases = cases, pop = pop, ex =
rural_pred, nstar = 200000, alpha = 0.05, longlat = TRUE)
cepphr
sort(cepphr$clusters[[1]]$locids)
sort(cepphp$clusters[[1]]$locids)
```

```
sslr <- scan.test(coords = coords, cases = cases, pop = pop, ex = rural_pred,
ubpop = 0.05, alpha = 0.05, longlat = TRUE)
sslr
sort(sslr$clusters[[1]]$locids)
sort(sslp$clusters[[1]]$locids)

ssmr <- scan.test(coords = coords, cases = cases, pop = pop, ex = rural_pred,
ubpop = 0.25, alpha = 0.05, longlat = TRUE)
ssmr
sort(ssmr$clusters[[1]]$locids)
sort(ssmp$clusters[[1]]$locids)

sshr <- scan.test(coords = coords, cases = cases, pop = pop, ex = rural_pred,
ubpop = 0.25, alpha = 0.05, longlat = TRUE)
sshr
sort(sshr$clusters[[1]]$locids)
sort(sshp$clusters[[1]]$locids)

# No significant changes when comparing most likely cluster.  CEPP tests are
exactly the same,
#    Spatial scan only slightly different - only includes more counties

### What about fitting suicide counts to just HS degrees?

testglm2 <-  glm(S1018 ~ HS, data = sdata, family = "poisson", offset =
log(Tpop))
HS_pred <- predict(testglm2, type = "response")

cepplh <- cepp.test(coords = coords, cases = cases, pop = pop, ex = hs_pred,
nstar = 20000, alpha = 0.05, longlat = TRUE)
cepplh
sort(cepplh$clusters[[1]]$locids)
sort(cepplp$clusters[[1]]$locids)

ceppmh <- cepp.test(coords = coords, cases = cases, pop = pop, ex = hs_pred,
nstar = 100000, alpha = 0.05, longlat = TRUE)
ceppmh
sort(ceppmh$clusters[[1]]$locids)
sort(ceppmp$clusters[[1]]$locids)

cepphh <- cepp.test(coords = coords, cases = cases, pop = pop, ex = hs_pred,
nstar = 200000, alpha = 0.05, longlat = TRUE)
cepphh
sort(cepphh$clusters[[1]]$locids)
sort(cepphp$clusters[[1]]$locids)

sslh <- scan.test(coords = coords, cases = cases, pop = pop, ex = hs_pred,
ubpop = 0.05, alpha = 0.05, longlat = TRUE)
sslh
sort(sslh$clusters[[1]]$locids)
sort(sslp$clusters[[1]]$locids)

ssmh <- scan.test(coords = coords, cases = cases, pop = pop, ex = hs_pred,
ubpop = 0.25, alpha = 0.05, longlat = TRUE)
```

```
ssmh
sort(ssmh$clusters[[1]]$locids)
sort(ssmp$clusters[[1]]$locids)

sshh <- scan.test(coords = coords, cases = cases, pop = pop, ex = hs_pred,
ubpop = 0.25, alpha = 0.05, longlat = TRUE)
sshh
sort(sshh$clusters[[1]]$locids)
sort(sshp$clusters[[1]]$locids)

# Again, no significantly changing results when comparing most likely cluster
for each test
# CEPP the same, Spatial scan the same or includes previous counties and 1-3
less counties
```