Where would you open a chipotle?

Introduction

Chipotle Mexican Grill ($CMG) is a typical Mexican fast-food restaurant founded in 1993 in America. The restaurant was based on the idea that fast-food restaurants do not have to be like other restaurants with similar menus. The name 'Chipotle' comes from the name of the dried, smoked jalapeno peppers. The founder of Chipotle Mexican Grill is an alumnus of the Culinary Institute of America in Hyde Park named Steve Ells. After graduation, he worked as a cook at Jeremiah Tower at Stars in San Francisco. Armed with lessons and experience, Steve founded his first Chipotle restaurant in Denver, Colorado, on July 13, 1993[9] with his father's $85,000 in the capital. In 2019, Chipotle Mexican Grill had branches of as many as 2,622 restaurants in various regions. With revenues of 5.58 billion USD and is included in the S&P 500 index

Some of us have thought about opening a business, and most of us have thought about grabbing a burrito from Chipotle. To do either of these things, we need to know where the current Chipotle stores are (and are not) located. We will use data from Thinknum to find potential locations for the next Chipotle restaurant and where we might recommend opening a Chipotle.

The dataset

The dataset we are using is from thinknum(Source: https://www.thinknum.com/datasets/nyse:cmg/store). Thinknum is a website that tracks thousands of websites capturing and indexing vast amounts of public data. It has the following columns:

- Id: Restaurant ID.
- Street: The address of the restaurant.
- st: In which state the location is located.
- crty: In which country the location is located.
- lat: The latitude of the restaurant location.
- lon: The longitude of the restaurant location.
- closed: The status of the restaurant closed or not.

Let's load the datasets and packages and take a look at the first few rows.

```{r}

# Load tidyverse, leaflet, and leaflet.extras

library(tidyverse)

library(leaflet)

library(leaflet.extras)

library(sf)

library(dismo)

library(dplyr)

# Read datasets/chipotle.csv into a tibble named chipotle using read_csv

chipotle <- read_csv("chipotle.csv")

# Load south_dakota_pop.rds into an object called south_dakota_pop

south_dakota_pop <- readRDS("south_dakota_pop.rds")

# Load chipotle_sd_locations.csv that contains proposed South Dakota locations

chipotle_sd_locations <- read_csv("chipotle_sd_locations.csv")

# load the Voronoi polygon data

polys <- readRDS("voronoi_polygons.rds")

head(chipotle)
```

```

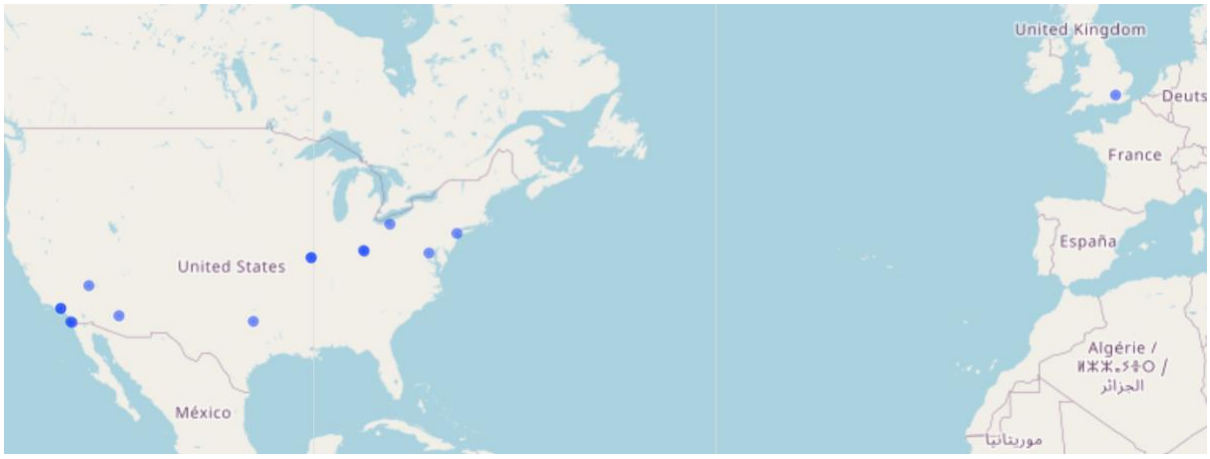
| | id | street | city | st | ctry | lat | lon | closed |
|---|---|---|---|---|---|---|---|---|
| 1 | 1358023 | 121 N. La Cienega Blvd., | Los Angeles | *NA* | United States | 34.07366 | -118.376499 | TRUE |
| 2 | 1358955 | 24369 Cedar Rd, | Lyndhurst | OH | United States | 41.50338 | -81.502983 | TRUE |
| 3 | 1359012 | 1130 West Grove Ave., | Mesa | *NA* | United States | 33.39023 | -111.855502 | TRUE |
| 4 | 1359490 | 6316 Delmar, | St. Louis | MO | United States | 38.65559 | -90.302554 | TRUE |
| 5 | 1359574 | 1464 St. Louis Galleria, | St. Louis | MO | United States | 38.63275 | -90.348553 | TRUE |
| 6 | 1359575 | 8301 Westchester, | Dallas | TX | United States | 32.86364 | -96.806573 | TRUE |
| 7 | 1359586 | 24 East 12th Street, | New York | NY | United States | 40.73406 | -73.993207 | TRUE |
| 8 | 1359597 | 11449 Princeton Pike, | Springdale | *NA* | United States | 39.28507 | -84.467213 | TRUE |
| 9 | 1359667 | 6340 W Charleston Blvd, #110 | Las Vegas | NV | United States | 36.15914 | -115.232725 | TRUE |
| 10 | 1359765 | 40 Wimbledon Hill Road, | London | UK | United Kingdom | 51.42227 | -0.209120 | TRUE |

# Empirical Analysis

First, let's make sure we don't recommend opening a location that has already been closed. This may also prevent many a disappointing Chipotle run to closed locations. Rather than looking just at the city/state pairs in the data, we can plot all closed locations to see exactly where the restaurants were located. Leaflet maps work with the %>% operator to pipe our chipotle data directly into a chain of function calls to produce an interactive map. All of the Chipotle locations have already been geocoded. The leaflet package will scan our column names for variables that are likely lat and lon, and if we use a common naming convention (e.g., lat/lon, latitude/longitude, or lat/long), the leaflet will automatically know which columns contain our coordinates. Because of their interactive features, leaflet maps can be especially helpful for exploratory data analyses. After making our first map take a minute or two to zoom in and pan the map to explore where Chipotle locations have closed.

```{r}
# Create a leaflet map of all closed Chipotle stores
closed_chipotles <- chipotle %>%
  # Filter the chipotle tibble to stores with a value of t for closed
  filter(closed == TRUE) %>%
  leaflet() %>%
  # Use addTiles to plot the closed stores on the default OpenStreet Map tile
  addTiles() %>%
  # Plot the closed stores using addCircles
  addCircles()

# Print map of closed chipotles
closed_chipotles
```
```



After exploring the map, the first question that comes to mind is, why did these particular locations close? In fact, why would any Chipotle ever close? Unfortunately, questions like this defy logic, so after quickly counting up the closed locations, this notebook moves on to the more important question of "Where should the next Chipotle be opened?". Rather than counting up all of the circles on our interactive map, we can use dplyr to quickly count the number of closed stores. After we note this, we'll create a new tibble that removes the closed locations from our data to avoid confusing them for open locations in future maps.

```{r}
# Use count from dplyr to count the values for the closed variable
chipotle %>%
  count(closed)

# Create a new tibble named chipotle_open that contains only open chipotle
chipotle_open <-
  chipotle %>%
  filter(closed == FALSE) %>%
  dplyr::select(-closed)
```

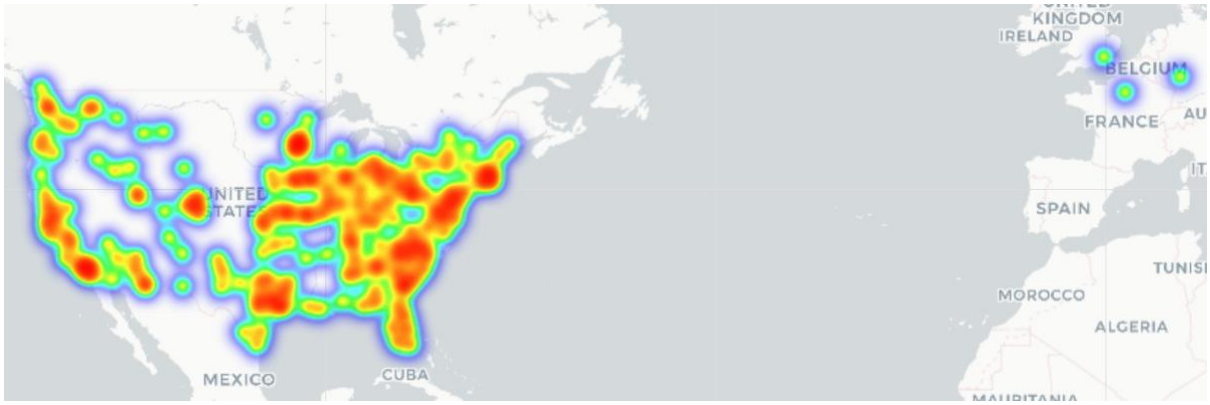
Closed	Number of Restaurant
FALSE	2469
TRUE	15

Where's the closest Chipotle? Perhaps, more interesting is a slightly different question, where aren't there Chipotles (in the US)? By mapping all of the Chipotle locations on an interactive leafletmap, we can explore patterns in the geographic distribution of the chain's locations. Since there are thousands of store locations, many of which are clustered closely together, we will start with a heatmap. Heatmaps are a popular option for mapping large points because they leverage a color scheme to represent the data rather than plotting each point individually. This enables users to quickly identify variation in the density of points and prevents tightly clustered points from overlapping. Zooming and panning the map, the heatmap will adjust based on the current view of the map. Are there any Chipotle deserts in the United States?

```{r}
```

```
# Pipe chipotle_open into a chain of leaflet functions
chipotle_heatmap <- chipotle_open %>%
  leaflet() %>%
  # Use addProviderTiles to add the CartoDB provider tile
  addProviderTiles("CartoDB") %>%
  # Use addHeatmap with a radius of 8
  addHeatmap(radius = 8)

# Print heatmap
chipotle_heatmap
```



Using the greyscale CartoDB provider tile with a colorful heatmap palette quickly revealed both the presence and absence of Chipotle stores throughout the United States. Using a greyscale base map is often useful for exploratory data analysis as it makes patterns of Chipotle clusters and Chipotle deserts stand out on the map. For example, panning and zooming the map reveals that Chipotles are often located on horizontal or vertical lines. Zooming in further indicates that stores are often located near interstate highways (check out Utah for an example). Let's take a closer look at where there are no Chipotle stores by quantifying the Chipotle deserts count the number of Chipotle locations in each US state.

```{r}

# Create a new tibble called chipotles_by_stat to store the results

chipotles_by_state <- chipotle_open %>%

  # Filter the data to only Chipotles in the Unites Status

  filter(ctry == "United States") %>%

  # Count the number of stores in chipotle_open by st

  count(st) %>%

  # Arrange the number of stores by state in ascending order

  arrange(n)
```

```

```{r}

# Print the state counts

head(chipotle_by_state)

```

State	Number of Chipotle Restaurants
Mississippi(MS)	1
North Dakota(ND)	1
Vermont(VT)	1
Wyoming(WY)	2
Montana(MT)	3
Maine(ME)	5

"`{r}

Print the state counts

tail(chipotle_by_state)

```

| State | Number of Chipotle Restaurants |
|---|---|
| Illinois(IL) | 135 |
| New York(NY) | 142 |
| Florida(FL) | 154 |
| Ohio(OH) | 177 |
| Texas(TX) | 204 |
| California(CA) | 417 |

The chipotle_by_state tibble had 48 rows, but there are 50 fifty states in the US. Why don't we have fifty rows? Perhaps, there are two (unfortunate) states that do not have a single Chipotle. Let's take a look using a couple of handy features that are included in base R.
"`{r}
# Print the state.abb vector
state.abb
```
'AL' 'AK' 'AZ' 'AR' 'CA' 'CO' 'CT' 'DE' 'FL' 'GA' 'HI' 'ID' 'IL' 'IN' 'IA' 'KS' 'KY' 'LA' 'ME' 'MD' 'MA' 'MI' 'MN' 'MS' 'MO' 'MT' 'NE' 'NV' 'NH' 'NJ' 'NM' 'NY' 'NC' 'ND' 'OH' 'OK' 'OR' 'PA' 'RI' 'SC' 'SD' 'TN' 'TX' 'UT' 'VT' 'VA' 'WA' 'WV' 'WI' 'WY'

"`{r}

Use the %in% operator to determine which states are in chipotles_by_state
state.abb %in% chipotle_by_state$st
```

```
TRUE FALSE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE FALSE TRUE
TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
TRUE TRUE FALSE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

"`{r}

```
# Use the %in% and ! operators to determine which states are not in chipotles_by_state
!state.abb %in% chipotle_by_state$st
```
```
FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE
FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE
FALSE FALSE FALSE FALSE FALSE FALSE
```

"`{r}

```
# Create a states_wo_chipotles vector
states_wo_chipotles <- state.abb[!state.abb %in% chipotle_by_state$st]

# Print states with no Chipotles
states_wo_chipotles
```
```
'AK' 'HI' 'SD'
```

There's no Chipotle restaurant in Alaska, Hawaii, and South Dakota.

48 + 3 = 51 states!?! Let's take a closer look at the values in our state variable that are not in the state.abb vector.

```{r}

chipotles_by_state$st[!chipotles_by_state$st %in% state.abb]

```

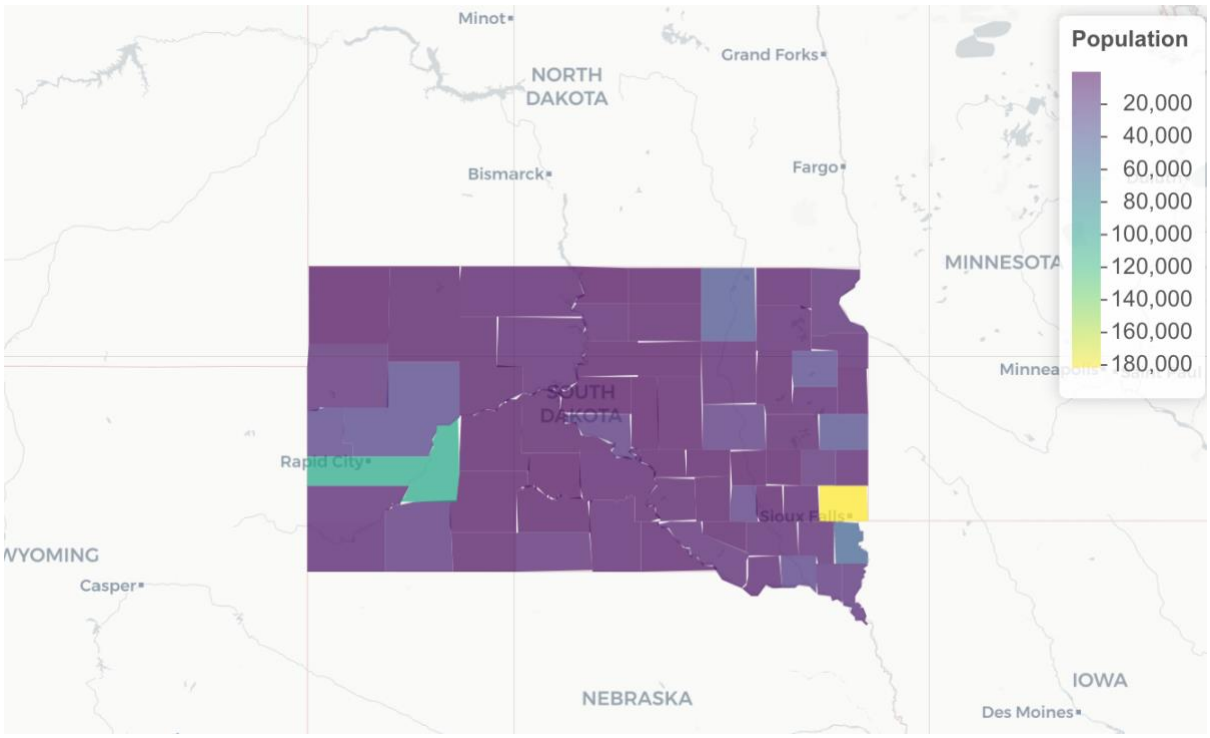
Washington D.C it's a district, not a state.

Now let's focus on the only state in the contiguous United States that does not have a Chipotle: South Dakota. If we were to open a Chipotle location in South Dakota, how might we select proposed locations? In the following chunks of code, we look at several maps to explore how the location of current Chipotles and geographic, transportation, and governmental features of the state may inform this decision. First, let's look at how South Dakota's population is distributed across the state using data from the US Census.

```{r}
# Create color palette to color map by county population estimate
pal <- colorNumeric(palette = "viridis", domain = south_dakota_pop$estimate)
```

```
sd_pop_map <- south_dakota_pop %>%
  leaflet() %>%
  addProviderTiles("CartoDB") %>%
  # Add county boundaries with addPolygons and color by population estimate
  addPolygons(stroke = FALSE, fillOpacity = 0.7, color = ~pal(estimate)) %>%
  # Add a legend using addLegend
  addLegend(pal = pal, values = ~estimate, title = "Population")

# Print map of South Dakota population by county
sd_pop_map
```



Minnehaha and Pennington's counties stand out on the population map. These counties are home to Sioux Falls and Rapid City, respectively. Let's take a closer look at each of the two largest cities in South Dakota to consider what features of the base map may be important when selecting the location of a Chipotle? Sioux Falls has a larger population, but Rapid City is proximate to Badlands National park, which has a million visitors a year. Additionally, we should note that I-90, a major interstate in America, runs through both cities. Let's plot a proposed Chipotle location in each city to further our exploration.

```{r}
# Limit chipotle store data to locations in states boardering South Dakota
chipotle_market_research <- chipotle_open %>%
  filter(st %in% c("IA", "MN", "MT", "ND", "NE", "WY")) %>%
  dplyr::select(city, st, lat, lon) %>%
  mutate(status = "open") %>%
  # bind the data on proposed SD locations onto the open store data
  bind_rows(chipotle_sd_locations)
```

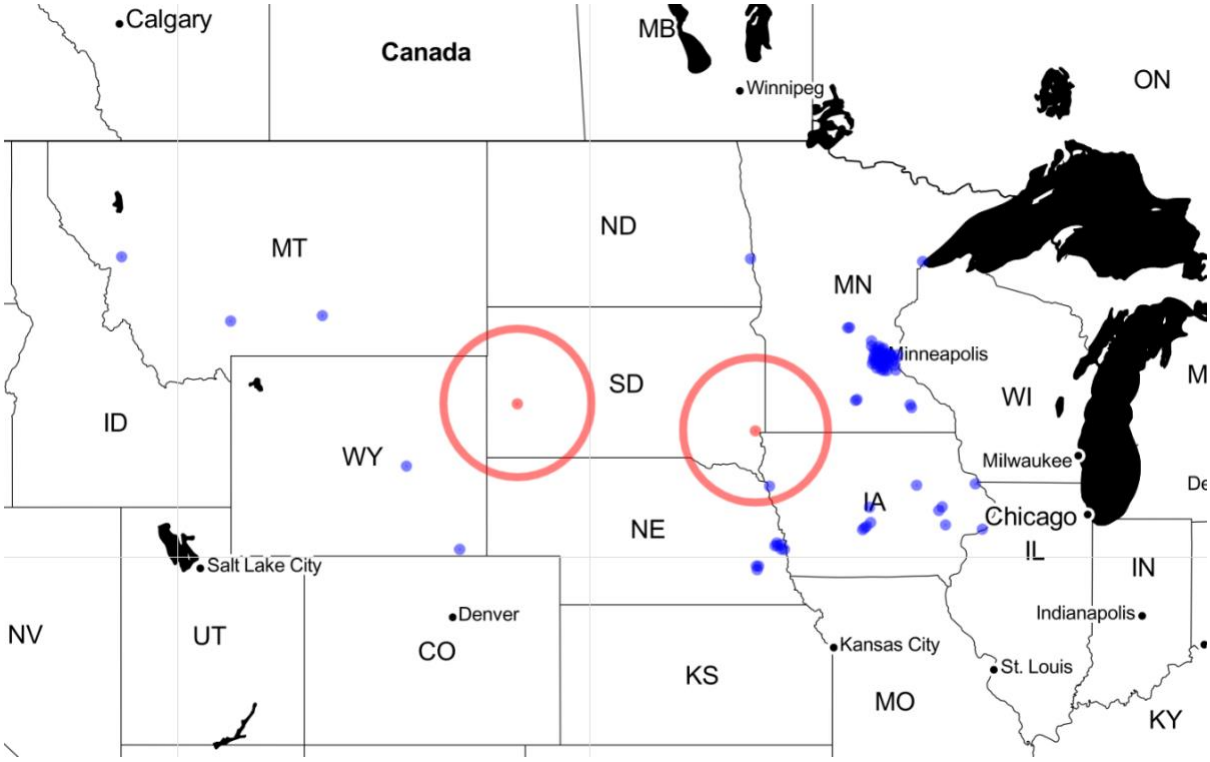
print the market research data
chipotle_market_research
```

| | city | st | lat | lon | status |
|---|---|---|---|---|---|
| 1 | Lincoln | NE | 40.81311 | -96.64009 | open |
| 2 | Brooklyn Park | MN | 45.09414 | -93.38321 | open |
| 3 | Eagan | MN | 44.83598 | -93.15196 | open |
| 4 | Champlin | MN | 45.15832 | -93.39078 | open |
| 5 | Woodbury | MN | 44.94463 | -92.90468 | open |
| 6 | Columbia Heights | MN | 45.06271 | -93.24830 | open |
| 7 | Fargo | ND | 46.85503 | -96.86172 | open |
| 8 | Iowa City | IA | 41.65975 | -91.53489 | open |
| 9 | Minnetonka | MN | 44.91803 | -93.50246 | open |

Let's map our proposed Chipotle restaurants in Sioux Falls and Rapid City so we can quickly see how close they are to the nearest open location. Let's apply this concept in a new leaflet map that plots all of the available and proposed Chipotle locations in South Dakota and its bordering states. Then adding a second layer to draw a circle around each of the proposed locations to determine if there is an open store within 100 miles. When using a categorical variable to create a color palette, colors can be mapped directly to the levels of the factor (i.e., there is one color in the palette for each level of the element), or the colorFactor function can interpolate the palette to create the necessary number of colors.
```{r}
# Create a blue and red color palette to distinguish between open and proposed stores
pal <- colorFactor(palette = c("Blue", "Red"), domain = c("open", "proposed"))

# Map the open and proposed locations
sd_proposed_map <-
  chipotle_market_research %>%
  leaflet() %>%
  # Add the Stamen Toner provider tile
  addProviderTiles("Stamen.Toner") %>%
  # Apply the pal color palette
  addCircles(color = ~pal(status)) %>%
  # Draw a circle with a 100 mi radius around the proposed locations
  addCircles(data = chipotle_sd_locations, radius = 100 * 1609.34, color = ~pal(status), fill = FALSE)

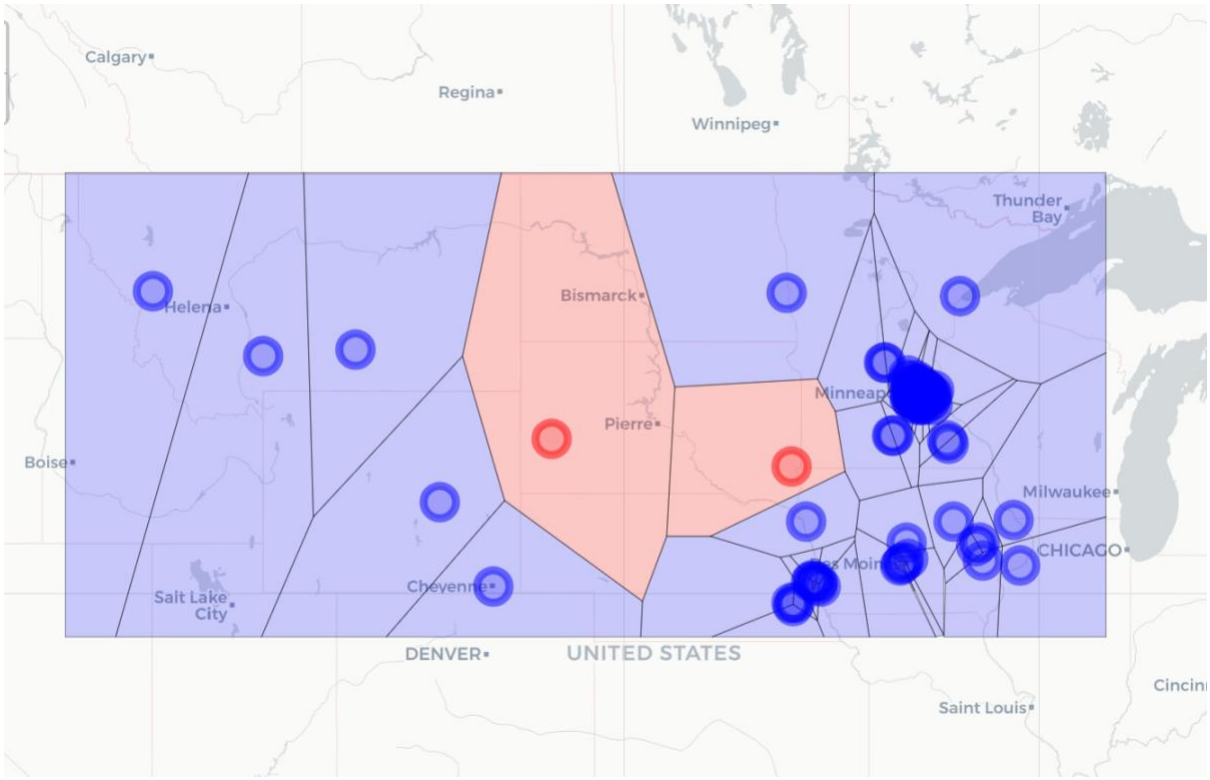
# Print the map of proposed locations
sd_proposed_map
```
```



It looks like there is a Chipotle within 100 miles of the proposed Sioux Falls location, but not Rapid City. This is helpful to know but perhaps even more helpful to understand all of the locations closer to a proposed Chipotle than to an open one. Voronoi polygons can be used to plot a polygon around each location. The bounds of each polygon will enclose all of the points on the map that is closest to a specific Chipotle. These polygons can then be used to visualize an approximation of the area covered by each Chipotle.

```{r}
voronoi_map <-
  polys %>%
  leaflet() %>%
  # Use the CartoDB provider tile
  addProviderTiles("CartoDB") %>%
  # Plot Voronoi polygons using addPolygons
  addPolygons(fillColor = ~pal(status), weight = 0.5, color = "black") %>%
  # Add proposed and open locations as another layer
  addCircleMarkers(data = chipotle_market_research, label = ~city, color = ~pal(status))

# Print the Voronoi map
voronoi_map
```



Conclusion

After we create a county-level choropleth map showing population, drawing circles with a 100-mile radius around each proposed location and mapping Voronoi polygons to estimate the area covered by each proposed Chipotle. We can conclude that if you want to open the Chipotle, we can open it in Rapid City, SD.

Source:

- https://en.wikipedia.org/wiki/List_of_states_and_territories_of_the_United_States
- https://www.thinknum.com/datasets/nyse:cmg/store
- https://statehood.dc.gov/page/faq
- https://id.wikipedia.org/wiki/Chipotle_Mexican_Grill

Notes:

- I use "dplyr::select()" instead of "select()" because there's new update from the R studio 4.0.0 , for further information please check on https://stackoverflow.com/questions/61428168/error-in-function-classes-fdef-mtable-unable-to-find-an-inherited-method