Final Project: The Socioeconomic and Food Reatil Landscape of San Francisco's Mission District

By: Fischer Heimburger (@fheimburger97) & Caroline Mahdavi (@cmahdavi)

Introduction

The Mission District (known as "This Mission") in San Francisco known for it's widening economic inequity, relatively high ethnic diversity–including being a local hub for Latinx culture and heritage, and vibrant food scene. This research seeks to first cultivate an understanding of the socio-economic landscape of the Mission District, and then initiate research on how this landscape is reflected in the distribution of food retailers.

Economic inequity is a major challenge facing the Mission District today. In 2020, it was found that "the highest and lowest income brackets compose almost 40% of all Mission households" (Mission Action Plan 2020. This demonstrates the increasing income gap experienced in the Mission District. In addition, the economic landscape of the Mission District is highly influenced by gentrification–the process by which the migration of high-income residents to a place displaces low-income, often marginalized, residents (Chapple and Thomas 2021). San Francisco was recently found to be the most intensely gentrified city in America (Richardson, Mitchell, and Edlebi 2020) and the Mission District, one of the neighborhoods with the most advanced levels of gentrification (Chapple and Thomas 2021). We hope to understand this unique and stark economic landscape by mapping out median household incomes by census tract within the Mission District. Our research on this is influenced by data and figures from the Urban Displacement Project. The Urban Displacement Project traces gentrification throughout the San Francisco Bay Area. As a part of their studies, they gather data on median income levels within the Mission District and map out their findings. We hope, in part, to recreate aspects of their figures with updated census data.

While much of the Bay Area is growing more racially segregated (Menendian, Gambhir and Gailes 2021), The Mission is unique for its relatively high ethnic diversity (Hom 2021). While overall in 2019 the Mission District was 38.7% Latinx, 36.4% White, and 13.7% Asian (City Data 2021); the demographic makeup of race varies depending on locale (Hom 2021). Additionally, recent years have shown dynamic shifts in demographics as the 2020 Census revealed a 14.4% decline in Latinx residents within the Mission District and a 34.8% increase in Asian residents (Horowitz and Jarret 2021). The Othering & Belonging Institute researches levels of segregation in the San Francisco Bay Area. As part of their research, they have created maps of the Mission District that display levels of segregation and racial diversity. We seek to create similar maps that display racial majorities by census tract to further understand the demographic landscape of The Mission. We then hope to understand how these socio-economic landscapes interact the landscape of food retailers in the Mission District.

Food systems are complex, interconnected bio-physical and socio-economic webs that are influenced by environmental, social, political, and economic systems, institutions, and actors (Ericksen 2008). For many living in cities, like those within the Mission District, relationships with food systems revolve around food retailers—including supermarkets, grocery stores, corner markets, and convenience stores—that help connect food producers with food consumers (Trivette 2019). Using data from the City of San Francisco, we hope to map out the current distribution of food retailers in the Mission District. Then we seek to understand how this distribution may be interconnected with class and race.

Our research is organized as follows. First, we map out income levels by census tract in the Mission District. Second, we map out racial majorities by census tract in The Mission. We then create a demographic table that organizes the socio-economic landscape of the district. In the fourth section we map out the distribution of food retailers within the Mission District. Lastly, we overlay the socio-economic maps with the food retailer map.

Data note: We used the following libraries to analyze our data. Additionally, please run `install.packages("tidycensus")` in the console window.

I Tract Income Typology

Table 1.1: Downloading San Francisco's Geopackage First, we downloaded the geopackage for San Francisco. In the following chunk, we load the geopackage using `read_sf` and `filter` for census tracts in the Mission District. We `select` to show "GEOID" and "geom" which will help create our map.

```
gfile <- read_sf("sanfrancisco.gpkg") %>%
  select(GEOID, geom) %>%
  filter(GEOID == "6075020200" | GEOID == "6075020100" | GEOID == "6075020800" | GEOID == "6075020700"
```

Table 1.2: Downloading San Francisco's Demographics We then use `read.csv` to load typology data from the Urban Displacement Project github. The definitions for variable names can be found here.

```
demographics <- read.csv("https://raw.githubusercontent.com/urban-displacement/displacement-typologies/r
```

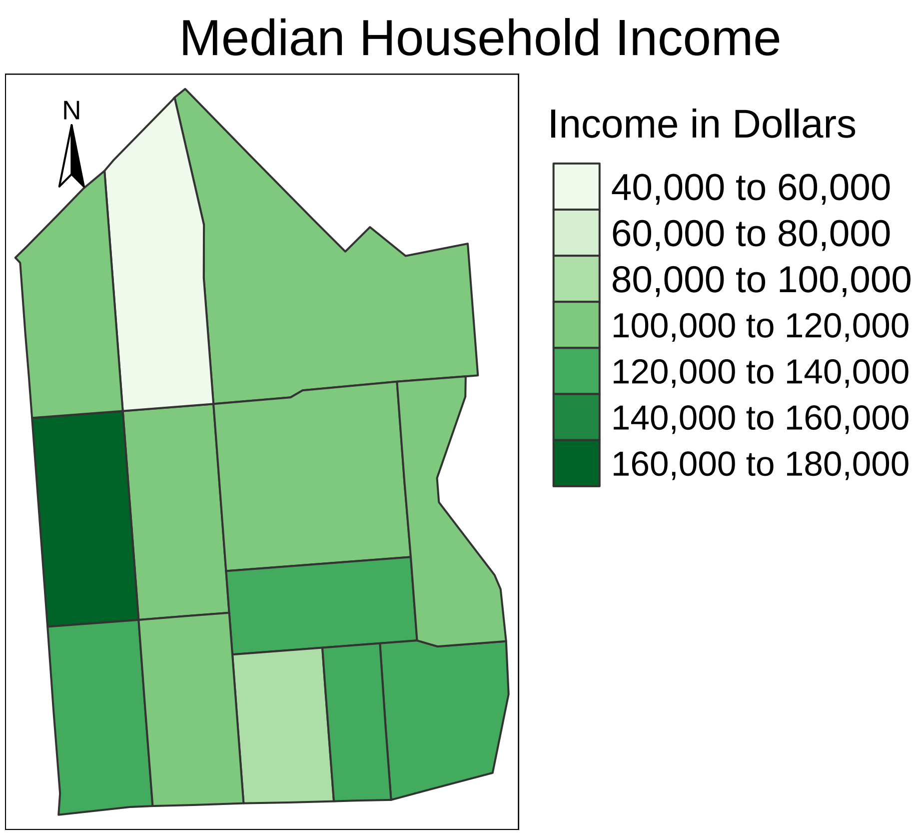
Table 1.3: Median Household Income Table First, we create the **medianincome** table which shows "GEOID" and "hinc_18"; "hinc_18" is the median household income in the past 12 months (in 2018 inflation-adjusted dollars) in 2018. Second, with `left_join`, we join the newly created **medianincome** table and the initial **gfile** table to create our **incomejoin** table.

```
medianincome <- demographics %>%
  select(GEOID, hinc_18) %>%
  rename("Income in Dollars" = hinc_18)
incomejoin <-left_join(gfile, medianincome, by = "GEOID")
incomejoin
```

```
## Simple feature collection with 13 features and 2 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -122.4269 ymin: 37.74782 xmax: -122.403 ymax: 37.77564
## Geodetic CRS:  WGS 84
## # A tibble: 13 x 3
##         GEOID                                            geom `Income in Dol~`
##         <dbl>                                <MULTIPOLYGON [°]>           <dbl>
##  1 6075020800 (((-122.4217 37.7633, -122.4173 37.76357, -122.4~           103134
##  2 6075017700 (((-122.4187 37.77564, -122.4156 37.7731, -122.4~           114722
##  3 6075020700 (((-122.4261 37.76304, -122.4217 37.7633, -122.4~           167422
##  4 6075020100 (((-122.4226 37.7725, -122.4222 37.77291, -122.4~            46750
##  5 6075022801 (((-122.4173 37.76357, -122.4136 37.76382, -122.~           117368
##  6 6075022803 (((-122.4167 37.75717, -122.4078 37.75771, -122.~           123000
##  7 6075022903 (((-122.4093 37.7544, -122.4074 37.75451, -122.4~           128750
##  8 6075021000 (((-122.4254 37.75503, -122.421 37.75529, -122.4~           138523
##  9 6075022901 (((-122.4164 37.75397, -122.412 37.75423, -122.4~            91464
## 10 6075020200 (((-122.4269 37.76917, -122.4264 37.7696, -122.4~           100099
## 11 6075022802 (((-122.4084 37.76443, -122.4051 37.76463, -122.~           102750
## 12 6075020900 (((-122.421 37.75529, -122.4187 37.75544, -122.4~           106875
## 13 6075022902 (((-122.412 37.75423, -122.4093 37.7544, -122.40~           133239
```

Figure 1.1: Mapping Median Household Income in the Mission District Using **tmap** and our **incomejoin** table, we then create a map showing median household income across census tracts in the Mission District.

```
incomemap <- tm_shape(incomejoin) +
             tm_style("watercolor") +
             tm_polygons("Income in Dollars") +
             tm_layout(main.title="Median Household Income",
                       main.title.position = "centre",
                       main.title.size = 1.6) +
             tm_legend(position = c("right", "top"),
             legend.outside = TRUE,
             legend.outside.size = .35,
             legend.title.size = 1.5,
             legend.text.size = 1.2) +
  tm_compass(position = c("left", "top"))
incomemap
```



II Tract Racial Typology

To work with census data, we install **tidycensus** from CRAN with the following command: `install.packages("tidycensus")`. After loading the **tidycensus** and **tidyverse** libraries, obtain a Census API key. Note: 2020 decennial Census data use differential privacy, a technique that introduces errors into data to preserve respondent confidentiality, small counts should be interpreted with caution.

```
census_api_key("77bb8e57772d9321db5adcd03b9bf2c3bce563c3", install = TRUE, overwrite = TRUE)
```

```
## [1] "77bb8e57772d9321db5adcd03b9bf2c3bce563c3"
```

Table 2.1 Population Numbers by Race Using **tidycensus**, we will get the values for four variables: total population, White population, Asian population, and Hispanic or Latino population. The argument to the `geography` parameter is "tract", and by by specifying `state`, `county`, and `year` we get the desired population numbers for each census tract in San Francisco from 2020. Note that `geometry` is set to true in the first table to help create maps later.

```
#White Population
white <- get_decennial(geography = "tract",
                       state = "CA",
                       county = "San Francisco",
                       year = 2020,
                       variables = c(white = "P1_003N"))
#Asian Population
asian <- get_decennial(geography = "tract",
                       state = "CA",
                       county = "San Francisco",
                       year = 2020,
                       variables = c(asian = "P1_006N"))
#Hispanic or Latino Population
latinx <- get_decennial(geography = "tract",
                       state = "CA",
                       county = "San Francisco",
                       year = 2020,
                       variables = c(latinx = "P2_002N"))
#Total Population
totalpop <- get_decennial(geography = "tract",
                       state = "CA",
                       county = "San Francisco",
                       year = 2020,
                       variables = c(totalpop = "P1_002N"),
                       geometry = TRUE)
```

```
## Downloading: 14 kB     Downloading: 14 kB     Downloading: 16 kB     Downloading: 16 kB     Download
```

Table 2.2: Race Percents Next, with a series of `left_join`, we join the four previous tables by "GEOID". We `rename` and `select` several important variables for neatness and clarity. We use `mutate` to add three columns displaying the percentage of the total population that each race is.

```
totalandwhite <- left_join(totalpop, white, by = "GEOID") %>%
    rename(tract = NAME.x,
           totalvar = variable.x,
           totalval = value.x,
           whitevar = variable.y,
           whiteval = value.y) %>%
  select(GEOID, tract, totalvar, totalval, whitevar, whiteval)
asianandlatinx <- left_join(asian, latinx, by = "GEOID") %>%
    rename(tract = NAME.x,
           asianvar = variable.x,
           asianval = value.x,
           latinxvar = variable.y,
           latinxval = value.y) %>%
  select(GEOID, asianvar, asianval, latinxvar, latinxval)
racetable <- left_join(totalandwhite, asianandlatinx, by = "GEOID") %>%
```

```
  mutate(percwhite = (whiteval/totalval)*100) %>%
  mutate(percasian = (asianval/totalval)*100) %>%
  mutate(perclatinx = (latinxval/totalval)*100)
racetable
```

```
## Simple feature collection with 244 features and 13 fields (with 2 geometries empty)
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -123.0139 ymin: 37.69274 xmax: -122.328 ymax: 37.86334
## Geodetic CRS:  NAD83
## # A tibble: 244 x 14
##    GEOID     tract totalvar totalval whitevar whiteval                  geometry
##  * <chr>     <chr> <chr>       <dbl> <chr>       <dbl>        <MULTIPOLYGON [°]>
##  1 06075035~ Cens~ totalpop     3888 white        1957 (((-122.5099 37.76409, --
##  2 06075040~ Cens~ totalpop     3936 white        2354 (((-122.4648 37.78856, --
##  3 06075047~ Cens~ totalpop     3527 white        1582 (((-122.4887 37.77611, --
##  4 06075026~ Cens~ totalpop     3852 white         260 (((-122.4113 37.71061, --
##  5 06075012~ Cens~ totalpop     3088 white        1566 (((-122.4154 37.78932, --
##  6 06075020~ Cens~ totalpop     3247 white        2238 (((-122.4352 37.76273, --
##  7 06075021~ Cens~ totalpop     2620 white        2106 (((-122.4428 37.75238, --
##  8 06075026~ Cens~ totalpop     2902 white         818 (((-122.4318 37.72825, --
##  9 06075025~ Cens~ totalpop     4182 white         551 (((-122.4105 37.72849, --
## 10 06075032~ Cens~ totalpop     3979 white        1553 (((-122.4848 37.76516, --
## # ... with 234 more rows, and 7 more variables: asianvar <chr>, asianval <dbl>,
## #   latinxvar <chr>, latinxval <dbl>, percwhite <dbl>, percasian <dbl>,
## #   perclatinx <dbl>
```

Table 2.3: Race Percents and Majorities Next, we `mutate` to add a "majority" column. In this column, we use `case_when` to show which of the three races has the highest percent of the population, and thus, is the majority of its tract.

```
percents <- racetable  %>%
  mutate(Majority = case_when(
    perclatinx > percasian & perclatinx > percwhite ~ "Hispanic or Latinx",
    percwhite > percasian & percwhite > perclatinx ~ "White",
    percasian > percwhite & percasian > perclatinx ~ "Asian")) %>%
  select(GEOID, tract, percwhite, percasian, perclatinx, Majority, geometry)
```

Table 2.4: Specifying the Boundaries To narrow our scope, we then `filter` to only include census tracts in the Mission District of San Francisco.

```
mission_geo_zero <- percents %>% pull(GEOID)
mission_percents_clean <- percents %>%
  mutate(GEOID = substring(mission_geo_zero, 2)) %>%
  filter(GEOID == "6075020201" | GEOID == "6075020202" | GEOID == "6075020101" | GEOID == "6075020102"
mission_percents_clean
```

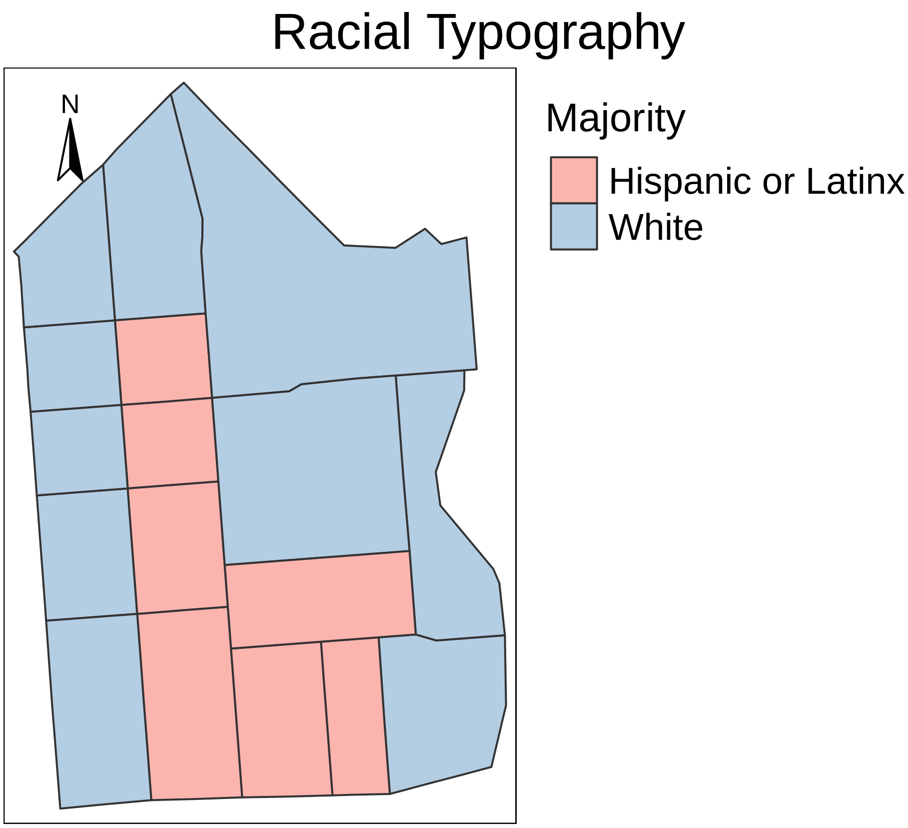
```
## Simple feature collection with 17 features and 6 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -122.4269 ymin: 37.74784 xmax: -122.4031 ymax: 37.77564
## Geodetic CRS:  NAD83
## # A tibble: 17 x 7
##    GEOID tract percwhite percasian perclatinx Majority                  geometry
##  * <chr> <chr>     <dbl>     <dbl>      <dbl> <chr>           <MULTIPOLYGON [°]>
```

```
## 1 6075~ Cens~      41.7      25.2       35.5 White    (((-122.4226 37.7725, -1~
## 2 6075~ Cens~      35.6      20.9       45.2 Hispani~ (((-122.422 37.76654, -1~
## 3 6075~ Cens~      72.8      14.3       18.0 White    (((-122.4258 37.75984, -~
## 4 6075~ Cens~      44.4      17.8       49.4 Hispani~ (((-122.421 37.75529, -1~
## 5 6075~ Cens~      37.7      11.6       61.6 Hispani~ (((-122.4164 37.75397, -~
## 6 6075~ Cens~      38.0      18.8       54.5 Hispani~ (((-122.4217 37.7633, -1~
## 7 6075~ Cens~      51.9      24.1       29.9 White    (((-122.4265 37.76627, -~
## 8 6075~ Cens~      48.8      17.1       47.8 White    (((-122.4093 37.7544, -1~
## 9 6075~ Cens~      47.6      26.0       32.8 White    (((-122.4084 37.76443, -~
## 10 6075~ Cens~     66.5      20.9       20.1 White    (((-122.4261 37.76304, -~
## 11 6075~ Cens~     68.2      17.5       21.8 White    (((-122.4254 37.75503, -~
## 12 6075~ Cens~     45.0      16.2       53.4 Hispani~ (((-122.412 37.75423, -1~
## 13 6075~ Cens~     44.9      19.6       46.9 Hispani~ (((-122.4214 37.7601, -1~
## 14 6075~ Cens~     46.2      23.2       30.2 White    (((-122.4187 37.77564, -~
## 15 6075~ Cens~     48.3      17.3       41.6 White    (((-122.4173 37.76357, -~
## 16 6075~ Cens~     52.7      26.1       25.8 White    (((-122.4269 37.76917, -~
## 17 6075~ Cens~     46.5      15.4       51.5 Hispani~ (((-122.4167 37.75717, -~
```

Figure 2.1: Mapping Racial Majorities in the Mission District

-

```
racemap <-
  tm_shape(mission_percents_clean) +
  tm_polygons("Majority") +
  tm_style("watercolor") +
  tm_layout(main.title = "Racial Typography",
            main.title.position = "centre",
            main.title.size = 1.6) +
  tm_legend(position = c("right", "top"),
            legend.outside = TRUE,
            legend.outside.size = .35,
            legend.title.size = 1.5,
            legend.text.size = 1.2) +
  tm_compass(position = c("left", "top"))
racemap
```



III Demographics Table

```
incomejoin_chr <- incomejoin %>%
  mutate_at("GEOID", as.character)
incomejoin_chr
```

Table 3.1

```
## Simple feature collection with 13 features and 2 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -122.4269 ymin: 37.74782 xmax: -122.403 ymax: 37.77564
## Geodetic CRS:  WGS 84
## # A tibble: 13 x 3
##    GEOID                                                   geom `Income in Dol~`
##  * <chr>                                     <MULTIPOLYGON [°]>            <dbl>
##  1 6075020800 (((-122.4217 37.7633, -122.4173 37.76357, -122.4~           103134
##  2 6075017700 (((-122.4187 37.77564, -122.4156 37.7731, -122.4~           114722
##  3 6075020700 (((-122.4261 37.76304, -122.4217 37.7633, -122.4~           167422
##  4 6075020100 (((-122.4226 37.7725, -122.4222 37.77291, -122.4~            46750
##  5 6075022801 (((-122.4173 37.76357, -122.4136 37.76382, -122.~           117368
##  6 6075022803 (((-122.4167 37.75717, -122.4078 37.75771, -122.~           123000
##  7 6075022903 (((-122.4093 37.7544, -122.4074 37.75451, -122.4~           128750
##  8 6075021000 (((-122.4254 37.75503, -122.421 37.75529, -122.4~           138523
##  9 6075022901 (((-122.4164 37.75397, -122.412 37.75423, -122.4~            91464
## 10 6075020200 (((-122.4269 37.76917, -122.4264 37.7696, -122.4~           100099
```

```
## 11 6075022802 (((-122.4084 37.76443, -122.4051 37.76463, -122.~           102750
## 12 6075020900 (((-122.421 37.75529, -122.4187 37.75544, -122.4~           106875
## 13 6075022902 (((-122.412 37.75423, -122.4093 37.7544, -122.40~           133239
```

```
finaljoin <- right_join(st_drop_geometry(incomejoin_chr), st_drop_geometry(mission_percents_clean), by =
  select(GEOID, tract, "Income in Dollars", Majority, percwhite, perclatinx, percasian) %>%
  rename("Median Income ($)" = "Income in Dollars", Tract = tract, "White Percentage" = percwhite, "His
finaljoin
```

Table 3.2: Demographics Table

```
## # A tibble: 17 x 7
##    GEOID      Tract            `Median Income~` `Racial Majori~` `White Percent~`
##    <chr>      <chr>                       <dbl> <chr>                       <dbl>
##  1 6075017700 Census Tract 1~            114722 White                        46.2
##  2 6075022801 Census Tract 2~            117368 White                        48.3
##  3 6075022803 Census Tract 2~            123000 Hispanic or Lat~             46.5
##  4 6075022903 Census Tract 2~            128750 White                        48.8
##  5 6075021000 Census Tract 2~            138523 White                        68.2
##  6 6075022901 Census Tract 2~             91464 Hispanic or Lat~             37.7
##  7 6075022802 Census Tract 2~            102750 White                        47.6
##  8 6075020900 Census Tract 2~            106875 Hispanic or Lat~             44.4
##  9 6075022902 Census Tract 2~            133239 Hispanic or Lat~             45.0
## 10 6075020101 Census Tract 2~                NA White                        41.7
## 11 6075020102 Census Tract 2~                NA Hispanic or Lat~             35.6
## 12 6075020701 Census Tract 2~                NA White                        72.8
## 13 6075020802 Census Tract 2~                NA Hispanic or Lat~             38.0
## 14 6075020201 Census Tract 2~                NA White                        51.9
## 15 6075020702 Census Tract 2~                NA White                        66.5
## 16 6075020801 Census Tract 2~                NA Hispanic or Lat~             44.9
## 17 6075020202 Census Tract 2~                NA White                        52.7
## # ... with 2 more variables: `Hispanic/Latinx Percentage` <dbl>,
## #   `Asian Percentage` <dbl>
```

IV Food Retailers

```
SF_businesses <- read_csv("https://data.sfgov.org/api/views/g8m3-pdis/rows.csv?accessType=DOWNLOAD")
```

Table 4.1: Uploading business data from data.sfgov.org.

```
## Rows: 287022 Columns: 32
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (24): Location Id, Business Account Number, Ownership Name, DBA Name, St...
## dbl  (6): Supervisor District, SF Find Neighborhoods, Current Police Distric...
## lgl  (2): Parking Tax, Transient Occupancy Tax
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
SF_businesses
```

```
## # A tibble: 287,022 x 32
##    `Location Id`  `Business Accou~` `Ownership Name` `DBA Name` `Street Address`
```

```
##    <chr>           <chr>             <chr>            <chr>      <chr>
##  1 0008450-25-002 0008450           Sutter Bay Hosp~ Californi~ 855 Geary St
##  2 1014347-12-141 0077021           Abm Industries ~ Ampco Sys~ 600 Montgomery ~
##  3 1014379-12-141 0077021           Abm Industries ~ Ampco Sys~ 501 Post St
##  4 0030032-46-001 0030032           Walgreen Co      Walgreens~ 845 Market St
##  5 0088595-01-001 0088595           Spano Robert & ~ 282-290 C~ 282 Clipper St
##  6 0028703-02-001 0028703           Vericlaim Inc    Vericlaim~ 500 Sansome St ~
##  7 0008450-01-013 0008450           Sutter Bay Hosp~ Californi~ 2323 Sacramento~
##  8 0009829-01-001 0009829           Forderer Cornic~ Forderer ~ 269 Potrero Ave
##  9 1012834-11-141 0091116           Urban Land Serv~ Urban Lan~ 1170 Sacramento~
## 10 0348331-01-001 0348331           Tran Sandy Dung  Elizabeth~ 672 Geary St
## # ... with 287,012 more rows, and 27 more variables: City <chr>, State <chr>,
## #   `Source Zipcode` <chr>, `Business Start Date` <chr>,
## #   `Business End Date` <chr>, `Location Start Date` <chr>,
## #   `Location End Date` <chr>, `Mail Address` <chr>, `Mail City` <chr>,
## #   `Mail Zipcode` <chr>, `Mail State` <chr>, `NAICS Code` <chr>,
## #   `NAICS Code Description` <chr>, `Parking Tax` <lgl>,
## #   `Transient Occupancy Tax` <lgl>, `LIC Code` <chr>, ...
```

```
Mission_H_Codes <- SF_businesses %>%
   filter(str_detect(`LIC Code`, "H"), `Neighborhoods - Analysis Boundaries` == "Mission") %>%
  select(`Ownership Name`, `DBA Name`, `Street Address`, `Business Start Date`, `Business End Date`, `Ma
Mission_H_Codes %>% select(`Ownership Name`, `DBA Name`, `LIC Code`, `LIC Code Description`)
```

Table 4.2: Business in the Mission with LIC Code H in it. LIC codes with H's are retail businesses, this was a crucial filtering step to finding food retailers.

```
## # A tibble: 984 x 4
##    `Ownership Name`              `DBA Name`          `LIC Code`  `LIC Code Desc~`
##    <chr>                         <chr>               <chr>       <chr>
##  1 Leigh Wendy A                 Listening Hands Ma~ H68         General Massage~
##  2 Walgreen Co                   Walgreens #03711    H05 Pos01~  Multiple
##  3 Walgreen Co                   Walgreens #09886    H04 Pos01~  Multiple
##  4 Walgreen Co                   Walgreen Co         H83         Supermarkets W/~
##  5 Rtrn Investment Llc           Travelodge Central  Hhh         <NA>
##  6 Dai Shujuan/altamirano Carlos Sanguchon           H25         Restaurant 1,00~
##  7 Chu Edwin W Y & Priscilla P C E P Laundromat      H46         Auto Laundry Me~
##  8 Naran Mangu                   Frances Hotel       Hhh         <NA>
##  9 Pan O Rama Baking Inc         Pan-O-Rama          H30         Catering Facili~
## 10 Eastern Pegasus Inc           Wild Pepper         H25         Restaurant 1,00~
## # ... with 974 more rows
```

```
H_strings <- c("H01", "H02", "H03", "H04", "H05", "H06", "H07", "H08", "H09", "H10", "H11", "H12", "H13
```

List 4.1: The following is a list of LIC codes that are food retailers. This list includes grocery stores, corner markets, convienence stores, supermarkets, bakeries, and drug stores. This list does not include restaurants.

```
Final_Mission_Food_Retailers <- SF_businesses %>%
  filter(`Neighborhoods - Analysis Boundaries` == "Mission") %>%
  filter(str_detect(`LIC Code`, paste(H_strings, collapse = "|")))
Final_Mission_Food_Retailers
```

Table 4.3: The follow is a table that includes data on all food retailers within the Mission District.

```
## # A tibble: 147 x 32
##    `Location Id`  `Business Accou~` `Ownership Name` `DBA Name` `Street Address`
##    <chr>          <chr>             <chr>            <chr>      <chr>
##  1 0030032-06-001 0030032           Walgreen Co      Walgreens~ 1189 Potrero Ave
##  2 0030032-40-001 0030032           Walgreen Co      Walgreens~ 3400 Cesar Chav~
##  3 0030032-01-015 0030032           Walgreen Co      Walgreen ~ 1979 Mission St
##  4 1021716-03-151 1010484           Yangtze Market ~ Yangtze M~ 2026 Mission St
##  5 0069288-01-001 0069288           Samiramis Impor~ Samiramis~ 2990 Mission St
##  6 0301049-01-001 0301049           Officemax Inc    Officemax~ 1750 Harrison St
##  7 0303375-02-001 0303375           Karajah Kamel F  Smoke Time 2733 Mission St
##  8 0090813-01-001 0090813           Rainbow Grocery~ Rainbow G~ 1745 Folsom St
##  9 1201886-10-181 1093331           Karla Garcia     Bris's Cr~ 2782 24th St
## 10 0108305-01-001 0108305           Totah B/totah M~ Norms Mar~ 2201 Bryant St
## # ... with 137 more rows, and 27 more variables: City <chr>, State <chr>,
## #   `Source Zipcode` <chr>, `Business Start Date` <chr>,
## #   `Business End Date` <chr>, `Location Start Date` <chr>,
## #   `Location End Date` <chr>, `Mail Address` <chr>, `Mail City` <chr>,
## #   `Mail Zipcode` <chr>, `Mail State` <chr>, `NAICS Code` <chr>,
## #   `NAICS Code Description` <chr>, `Parking Tax` <lgl>,
## #   `Transient Occupancy Tax` <lgl>, `LIC Code` <chr>, ...
```

```r
Clean_Final_Retail <- Final_Mission_Food_Retailers %>%
  select("Ownership Name", "DBA Name", "Street Address", "LIC Code Description")
Clean_Final_Retail
```

```
## # A tibble: 147 x 4
##    `Ownership Name`        `DBA Name`          `Street Address` `LIC Code Desc~`
##    <chr>                   <chr>               <chr>            <chr>
##  1 Walgreen Co             Walgreens #03711    1189 Potrero Ave Multiple
##  2 Walgreen Co             Walgreens #09886    3400 Cesar Chav~ Multiple
##  3 Walgreen Co             Walgreen Co         1979 Mission St  Supermarkets W/~
##  4 Yangtze Market Inc      Yangtze Market      2026 Mission St  Multiple
##  5 Samiramis Imports Inc   Samiramis Imports~  2990 Mission St  Multiple
##  6 Officemax Inc           Officemax No Amer~  1750 Harrison St Multiple
##  7 Karajah Kamel F         Smoke Time          2733 Mission St  Multiple
##  8 Rainbow Grocery Inc     Rainbow Grocery C~  1745 Folsom St   Multiple
##  9 Karla Garcia            Bris's Creations    2782 24th St     Retail Bakeries~
## 10 Totah B/totah M/ Totah N Norms Market       2201 Bryant St   Multiple
## # ... with 137 more rows
```

```r
Clean_Order_r <- Clean_Final_Retail[order(Clean_Final_Retail$"DBA Name"),]
Clean_Order_r
```

```
## # A tibble: 147 x 4
##    `Ownership Name`                 `DBA Name` `Street Address` `LIC Code Desc~`
##    <chr>                            <chr>      <chr>            <chr>
##  1 Binaya Pokharel And Mandira Shr~ 23rd & Gu~ 3558 23rd St     Retail Food Mar~
##  2 Samra Bros Inc                   26th & Gu~ 1400 Guerrero St Multiple
##  3 Shehadeh Nizar A                 Abc Market 2801 Bryant St   Multiple
##  4 Mosleh Hamood                    All Seaso~ 401 Capp St      Multiple
##  5 All Season Market                All Seaso~ 401 Capp St      Multiple
##  6 Cervantes Marisa                 All Star ~ 3350 18th St     Retail Mkts W/o~
##  7 Anthony's Cookies Inc            Anthony's~ 1417 Valencia St Retail Bakeries~
```

```
##   8 Elias Carmen                       Bakery La~ 3329 24th St     Retail Bakeries~
##   9 Lai Hung Dat                       Basa Seaf~ 3064 24th St     Multiple
## 10 Best Buy Stores, L.p.              Best Buy ~ 1717 Harrison St Retail Mkts W/o~
## # ... with 137 more rows
```

```
write.csv(Clean_Order_r, "C:\\Users\\fheim\\Desktop\\Test\\People.csv", row.names = FALSE)
```

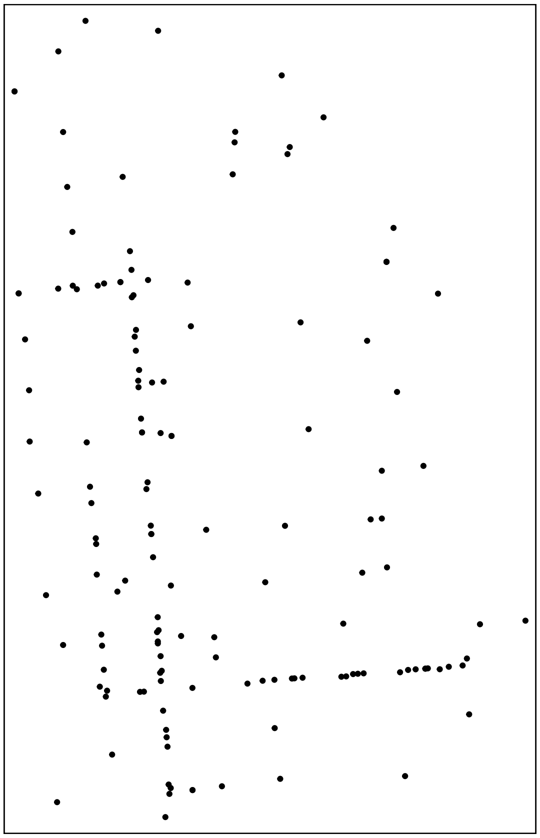
```
coordinates <- Final_Mission_Food_Retailers %>% pull("Business Location")
```

List 4.2: The following are the coordinates for each food retail store in the Mission District.

Figure 4.1: The follow is a map of all food retailers within the Mission District. Each dot represents a food retailer. While this map does not include streets or the outline of the Mission Distict, maps in the following section will include an outline of the Mission District along with demographic information.

-

```
geo_final <- Final_Mission_Food_Retailers %>%
  rename(geometry = `Business Location`) %>%
  mutate(geometry = as.list(geometry)) %>%
  st_as_sf(crs = "WGS84")
tm_shape(geo_final) + tm_dots()
```



V Final Maps

```
dots <- tm_shape(geo_final) + tm_dots(col="black") + tm_add_legend(shape = "tm_dots", labels = ('Food Re
incomemap_with_food_retailers <- tm_shape(incomejoin) +
             tm_style("watercolor") +
             tm_polygons("Income in Dollars") +
             tm_layout(main.title="Median Household Income & Food Retailers",
                       main.title.position = "centre",
                       main.title.size = 1.6) +
             tm_legend(position = c("right", "top"),
             legend.outside = TRUE,
             legend.outside.size = .35,
             legend.title.size = 1.5,
             legend.text.size = 1.2) +
  tm_compass(position = c("left", "top"))
incomemap_with_food_retailers + dots
```

Figure 5.1: Food Retailers and Median Household Income in the Mission District

```
## Some legend labels were too wide. These labels have been resized to 1.12, 1.05, 1.05, 1.05, 1.05. In
```

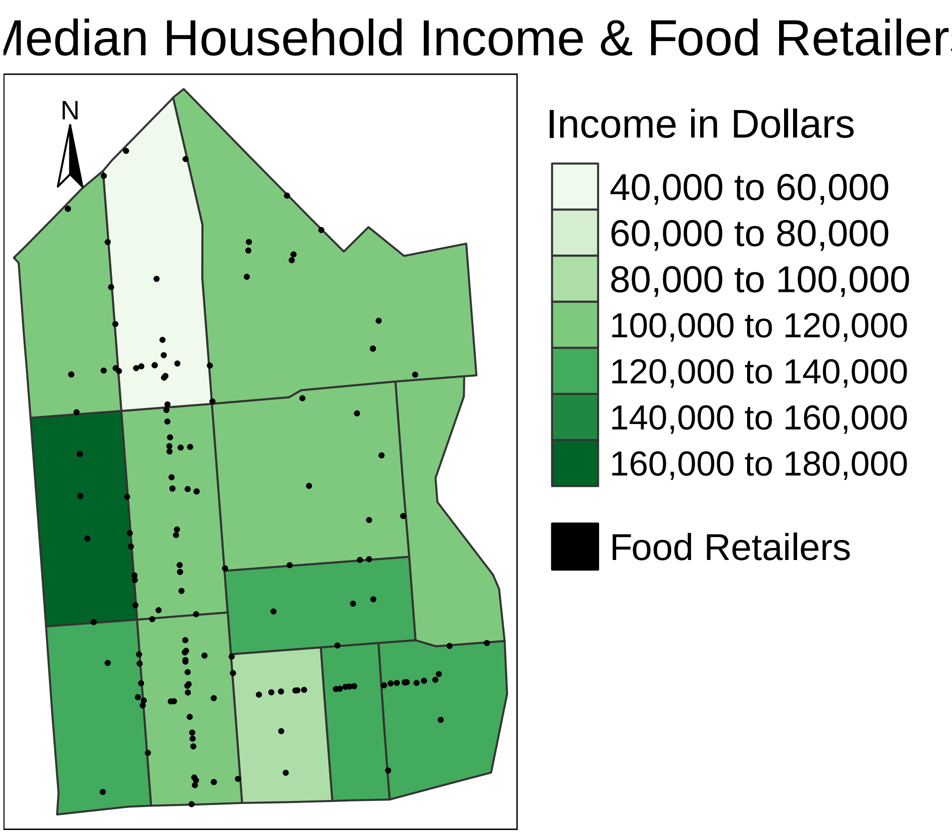
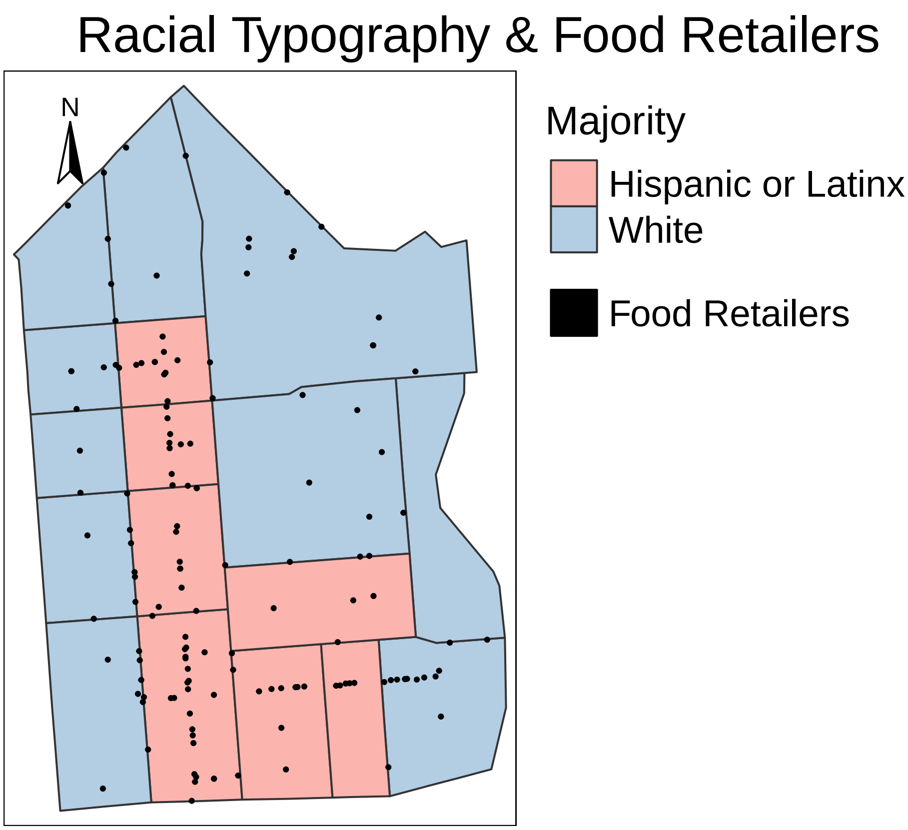


Figure 5.2: Food Retailers and Racial Majorities in the Mission District

-

```
dots <- tm_shape(geo_final) + tm_dots(col="black") + tm_add_legend(shape = "tm_dots", labels = ('Food Re
racemap_with_food_retailers <-
  tm_shape(mission_percents_clean) +
```

```r
  tm_polygons("Majority") +
  tm_style("watercolor") +
  tm_layout(main.title = "Racial Typography & Food Retailers",
            main.title.position = "centre",
            main.title.size = 1.6) +
  tm_legend(position = c("right", "top"),
            legend.outside = TRUE,
            legend.outside.size = .35,
            legend.title.size = 1.5,
            legend.text.size = 1.2) +
  tm_compass(position = c("left", "top"))
racemap_with_food_retailers + dots
```



Conclusion

We were able to create maps of the Mission District based off median household income, racial majorities, and the distribution of food retailers. First, our map displaying median household income demonstrates the income gap experienced in the Mission today. Median income in tracts ranged from 40,000-60,000 USD on the lower end, to 160,000-180,000 USD on the upper end, with five other ranges in between. The lowest and highest median income tracts share a corner in the Northwestern quadrant of the Mission, demonstrating the proximity of the income gap in the Mission District.

Our racial typography map revealed that Hispanic or Latinx and White are the majority races in the Mission District. There are ten tracts that were predominantly white and seven tracts that are predominantly Hispanic or Latinx. These findings demonstrate that a degree of racial diversity exists in the Mission District, albeit largely between two races. If you compare the Median Household Income map with the Racial Typography map it is apparent that there is some degree of correlation between lower-income census tracts and tracts that are predominately Hispanic/Latinx. This may be an important trend when researching levels of gentrification within the Mission District. However, while our findings display that there is a degree of racial integration within the Mission District, they do not reveal the degree of segregation occurring within each tact. Further research would be needed to understand the landscape of segregation within each tract.

When we examined the socio-economic landscape of the Mission District in relation to the distribution of food retailers, we observed that there seems to be a level of correlation between the number of food retailers and tracts that are predominantly Hispanic and Latinx. If the maps we generated are compared to street maps (we direct readers to use Google Maps or other mapping software), the majority of food retailers are located along Mission Street and 24th Street–both streets are hubs for Latinx populations and culture. Our Racial Typography & Food Retailers map exemplifies that there may be a correlation between the location and quantity of food retailers in the Mission District and racial majority; however, our Median Household Income & Food Retailers map doesn't demonstrate the same level of correlation. In the later case, the location of food retailers may have more to do with business corridors than median household income distribution.

In conclusion, we were able to replicate multiple maps used in researching the socio-economic landscapes of the Mission District. We were then able to start research, and open a discussion, about how this correlates with the distribution of food retailers within the district. Our work should be scene as a catalyst for future research on the food retail landscape of the Mission District. For example, it should be stated that in our maps all food retailers–regardless of size, stock, or purpose–are resembled by a single black dot. However, there is great diversity among food retailers. Some may be supermarkets, convenience stores, or corner markets. Additionally, some may cater to the general public, while others may sell specialty or ethnic foods. These kinds of differences would be important to know before drawing any conclusions regarding the distribution of food retailers in the Mission District in relation to socio-economic landscapes. Using our research as a launching pad, it is our hope that researches take our findings and code to further examine the role of food retailers in the socio-economic fabric of the Mission District.

###{r} `Test_businesses <- read_csv("TEST_Survey_Responses.csv") %>%   rename(addresses = Address) Test_businesses url_nominatim_search <- function(search_query_url, country_url, language_url, email_url) {   # load libraries   library(RCurl)   # nominatim search api url   url_nominatim_search_api <- "https://nominatim.openstreetmap.org/search/"   # convert input into a list   search_query_url <- sapply(search_query_url, as.list)   # percent-encode search request   search_query_url <- sapply(search_query_url, URLencode)   # parameters   if (!is.null(country_url)) {   country_url <- paste0("&countrycodes=", country_url)   }   parameters_url <- paste0("?format=json",   "&addressd country_url, "&accept-language=", language_url,   "&email=", email_url)   # construct search request for geocode   url_nominatim_search_call <- paste0(url_nominatim_search_api,   search_query_url, parameters_url)   return(url_nominatim_search_call) } # // # 2. EXTRACT DATA FROM JSON # // get_geodata_from_json_nom <- function(geodata_json) {   # load library   library(jsonlite)   # convert json output into r object   geodata <- lapply(geodata_json, fromJSON,simplifyVector = FALSE)   # extract coordinates, address and contacts   lat_lng_a_c <- Test_businesses(lat = NA, lng = NA, address = NA, pub_name = NA,   street_name = NA, house_number = NA, suburb = NA,   postcode = NA, state_district = NA, website_1 = NA,   website_2 = NA, website_3 = NA, phone_1 = NA,   phone_2 = NA, email_1 = NA, email_2 = NA)   for(i in 1:length(geodata)) {   if(length(geodata[[i]]) != 0) {   # get data   lat <- geodata[[i]][[1]]$lat   lng <- geodata[[i]][[1]]$lon address <- geodata[[i]][[1]]$display_name   pub_name <- geodata[[i]][[1]]$address$pub street_name <- geodata[[i]][[1]]$address$road   house_number <- geodata[[i]][[1]]$address$ suburb <- geodata[[i]][[1]]$address$suburb   postcode <- geodata[[i]][[1]]$address$postco state_district <- geodata[[i]][[1]]$address$state_district   website_1 <-`

```
geodata[[i]][[1]]$extratags$website                website_2 <- geodata[[i]][[1]]$extratags$url
website_3 <- geodata[[i]][[1]]$extratags$`contact:website`                phone_1 <-
geodata[[i]][[1]]$extratags$phone                phone_2 <- geodata[[i]][[1]]$extratags$`contact:phone
email_1 <- geodata[[i]][[1]]$extratags$email                email_2 <- geodata[[i]][[1]]$extratags$`co
# get rid of NULLs                info <- list(lat, lng, address, pub_name, street_name,
house_number, suburb, postcode, state_district,                              website_1,
website_2, website_3,                              phone_1, phone_2, email_1, email_2)
for (j in 1:length(info)) {                     if (is.null(info[[j]])) info[[j]] <- NA
}                # create output data frame                lat_lng_a_c[i, ] <- info
} else {                lat_lng_a_c[i, ] <- NA           }     }     return(lat_lng_a_c)
} # ////////////////////////////////////////////// # MAIN FUNCTION # //////////////////////////////////
geocode_nominatim <- function(search_query, country = NULL, language = "en",
= "coordinates", email) {     # LOAD LIBRARIES     library(RCurl)     # EXTRACT DATA
# construct url for geocoding         url_geocode <- url_nominatim_search(search_query,
country, language, email)         # get data from nominatim         # wait 3 seconds
between each call         geodata_json <- list()         for (i in 1:length(url_geocode))
{             geodata_json[i] <- getURL(url_geocode[i])             Sys.sleep(3)         }
# get data from json output         geodata_df <- as.data.frame(sapply(search_query,
as.character),                              stringsAsFactors = FALSE)         names(geodata_df)
<- "search query"         rownames(geodata_df) <- NULL         geodata_df[, 2:17] <-
get_geodata_from_json_nominatim(geodata_json)         geodata_df_query <- data.frame(search_query
= geodata_df[, 1],                              stringsAsFactors = FALSE)
geodata_df_coordinates <- geodata_df[, 2:3]         geodata_df_address <- geodata_df[,
4:10]         geodata_df_contacts <- geodata_df[, 11:17]         # return dataframe with
the geodata         geodata_result <- geodata_df_query         if("all" %in% fields)
{             geodata_result <- cbind(geodata_result, geodata_df[, 2:17])         }
if("coordinates" %in% fields) {             geodata_result <- cbind(geodata_result,
geodata_df_coordinates)         }         if("address" %in% fields) {             geodata_result
<- cbind(geodata_result, geodata_df_address)         }         if("contacts" %in% fields)
{             geodata_result <- cbind(geodata_result, geodata_df_contacts)         }
return(geodata_result) } ###
```