HW2

Melaina Wright

May 9, 2017

```
library(tidyverse)

library(tibble)

library(readr)
```

Import

```
#Read the gazetteer data as-is (all columns; no type conversion) into a gaz_raw tibble.
gaz_raw <-read_csv("CA_Features_20170401.zip")
```

```
## Parsed with column specification:
## cols(
##   `FEATURE_ID|FEATURE_NAME|FEATURE_CLASS|STATE_ALPHA|STATE_NUMERIC|COUNTY_NAME|COUNTY_NUMERIC|PRIMAR
## )
```

```
## Warning: 11 parsing failures.
##   row col  expected    actual                       file
## 60566  -- 1 columns 2 columns 'CA_Features_20170401.zip'
## 63884  -- 1 columns 2 columns 'CA_Features_20170401.zip'
## 70687  -- 1 columns 2 columns 'CA_Features_20170401.zip'
## 70688  -- 1 columns 2 columns 'CA_Features_20170401.zip'
## 73865  -- 1 columns 2 columns 'CA_Features_20170401.zip'
## ..... ... ......... ......... ......................
## See problems(...) for more details.
```

```
#unzipped it but is not deliminated correctly
gaz_raw <- read_delim("CA_Features_20170401.zip", delim="|")
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   FEATURE_ID = col_integer(),
##   PRIM_LAT_DEC = col_double(),
##   PRIM_LONG_DEC = col_double(),
##   SOURCE_LAT_DEC = col_double(),
##   SOURCE_LONG_DEC = col_double(),
##   ELEV_IN_M = col_integer(),
##   ELEV_IN_FT = col_integer()
## )
```

```
## See spec(...) for full column specifications.
```

Tidy

```
# Copy only the following columns into a gaz tibble (you can rename them if you like):
# feature ID
# feature name
# feature class
# state alpha
# county name
# primary latitude (decimal)
# primary longitude (decimal)
# source latitude (decimal)
# source longitude (decimal)
# elevation in meters
# map name
# date created
# date edited

gaz <- select(gaz_raw, FEATURE_ID, FEATURE_NAME, FEATURE_CLASS, STATE_ALPHA, COUNTY_NAME, PRIM_LAT_DEC,



# Convert the gaz columns to the appropriate type.
# Parsed with column specification:
# cols(
#   .default = col_character(), -> thinks date created and date edited are characters, should be dates
#   FEATURE_ID = col_integer(),
#   PRIM_LAT_DEC = col_double(),
#   PRIM_LONG_DEC = col_double(),
#   SOURCE_LAT_DEC = col_double(),
#   SOURCE_LONG_DEC = col_double(),
#   ELEV_IN_M = col_integer(),
#   ELEV_IN_FT = col_integer()
# )

gaz$DATE_CREATED = as.Date(gaz$DATE_CREATED, "%m/%d/%Y")

gaz$DATE_EDITED = as.Date(gaz$DATE_EDITED, "%m/%d/%Y")



#Convert any placeholders for unknown data to NA
gaz$MAP_NAME <- parse_character(gaz$MAP_NAME, na="Unknown")

gaz$PRIM_LAT_DEC <- parse_character(gaz$PRIM_LAT_DEC, na="0")

gaz$PRIM_LONG_DEC <- parse_character(gaz$PRIM_LONG_DEC, na="0")

#revert back to numeric after using parse_character
gaz$PRIM_LAT_DEC <- as.numeric(gaz$PRIM_LAT_DEC)

gaz$PRIM_LONG_DEC <- as.numeric(gaz$PRIM_LONG_DEC)
```

```r
# Delete from gaz rows where:
# the primary latitude or longitude are unknown -> delete NA
gaz <- filter(gaz, !PRIM_LAT_DEC == "NA")
gaz <- filter(gaz, !PRIM_LONG_DEC == "NA")



#Delete from gaz rows where:
# the feature is not in California -> only select CA
gaz <- filter(gaz, STATE_ALPHA == "CA")



# Write the gaz tibble to a CSV files (using "|" as a delimiter)
write.table(gaz, "gaz.csv", sep="|")
```

Analyze

Most-Frequently-Occuring Feature Name

```r
# Create R code snippets that answer the following questions about California:
# What is the most-frequently-occuring feature name?

Mode <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]
}

Mode(gaz$FEATURE_NAME)
```

```
## [1] "Church of Christ"
```

Least-Frequently-Occuring Feature Class

```r
# What is the least-frequently-occuring feature class?

UnMode <- function(x) {
  ux <- unique(x)
  ux[which.min(tabulate(match(x, ux)))]
}

UnMode(gaz$FEATURE_CLASS)
```

```
## [1] "Sea"
```

Approximate Center Point of Each County

```r
# What is the approximate center point of each county?
# Hint: Calculate the center of the bounding box of the county's point features.

gaz <- filter(gaz, !COUNTY_NAME == "NA")

ucounties <- unique(gaz$COUNTY_NAME)

CountyCenter <- function(county, dataframe) {
  points = subset(gaz, COUNTY_NAME == county)
  latmin = min(points$PRIM_LAT_DEC)
  latmax = max(points$PRIM_LAT_DEC)
  longmin = min(points$PRIM_LONG_DEC)
  longmax = max(points$PRIM_LONG_DEC)
  centerpoint = c((latmin+latmax)/2, (longmin+longmax)/2)
  return(centerpoint)
}


for(county in ucounties) {
  cp = CountyCenter(county,gaz)
  #print(cp)
  cat(county,"\t",cp, "\n")
}
```

```
## Imperial      33.05796 -115.2855
## Sacramento    39.09157 -121.6143
## Monterey      36.3326 -121.1135
## Merced    37.18383 -120.6907
## Alameda   37.68525 -121.9243
## Contra Costa      37.90659 -121.9944
## Solano    36.13968 -120.5618
## Santa Clara   38.88102 -121.8937
## Tuolumne      39.00205 -121.5801
## Humboldt      40.65793 -122.0243
## Calaveras     36.46287 -119.8929
## El Dorado     37.97298 -121.4447
## Marin     36.83411 -121.9622
## Santa Cruz    35.63555 -120.4298
## Fresno    36.74745 -119.6338
## San Francisco     36.00691 -120.8974
## Mendocino     39.38642 -123.4288
## Sutter    39.03162 -121.6965
## Lake      39.13503 -122.7503
## Siskiyou      41.5011 -122.581
## San Mateo     37.39077 -122.3197
## Lassen    40.46185 -120.8094
## Stanislaus    37.61389 -120.9406
## Tehama    40.05988 -122.1986
## Yolo      38.63463 -121.9447
## San Joaquin   37.89854 -121.253
## Nevada    39.26712 -120.6413
## Placer    39.03074 -120.7767
## Alpine    37.61799 -118.229
## Sonoma    38.46991 -122.5055
```

```
## Napa      38.49838 -122.3625
## Tulare    36.29215 -118.78
## Madera    36.04061 -119.7934
## Inyo      36.60175 -117.2923
## Colusa    39.16739 -122.278
## Amador    38.35542 -121.0613
## Shasta    40.78738 -121.6522
## Sierra    37.37716 -120.5399
## Modoc     41.58628 -120.7315
## Glenn     39.62933 -122.4071
## Yuba      39.27781 -121.3127
## San Benito    36.59107 -121.1142
## Trinity   40.66529 -123.0399
## Mono      38.06252 -118.9393
## Mariposa      37.24062 -119.3346
## Plumas    37.34911 -119.4346
## Butte     39.72335 -121.5716
## Del Norte     41.69998 -123.955
## Riverside     36.14524 -118.4051
## Los Angeles   35.08041 -118.9973
## San Bernardino    35.87137 -117.8211
## San Diego     35.16475 -119.236
## San Luis Obispo   35.3558 -120.4077
## Ventura   34.04778 -119.1644
## Orange    33.66613 -117.7801
## Santa Barbara     33.96514 -119.5775
## Kern      35.34304 -119.4605
## Kings     36.13049 -119.887
```

```
#prints out the x and y coordinates of the center of the boundary box for each county
```

Fraction of Total Number of Features in Each Country that are Natural and that are Man-Made

```
# What are the fractions of the total number of features in each county that are natural? man-made?
# Hint: Construct a tibble with two columns, one containing all possible feature classes (see "Feature

POSSIBLE_FEATURE_CLASSES <- c("Airport", "Arch", "Area", "Arroyo", "Bar", "Basin", "Bay", "Beach", "Ben


MANMADE_VS_NATURAL <- c("ManMade", "ManMade", "Natural", "Natural", "Natural", "Natural", "Natural", "Na


FeatureTibble <- tibble(
  FEATURE_CLASS = POSSIBLE_FEATURE_CLASSES,
  MANMADE_VS_NATURAL = MANMADE_VS_NATURAL
)


gaz2 <- FeatureTibble %>% left_join(gaz, by = "FEATURE_CLASS")


ucounties <- unique(gaz2$COUNTY_NAME)
```

```
Fraction <- function(county, dataframe) {
  points = subset(gaz2, COUNTY_NAME == county)
  manmade = filter(points, MANMADE_VS_NATURAL == "ManMade")
  natural = filter(points, MANMADE_VS_NATURAL == "Natural")
  manmadelength = length(manmade$MANMADE_VS_NATURAL)
  naturallength = length(natural$MANMADE_VS_NATURAL)
  FractionNatural = (naturallength)/((manmadelength) + (naturallength))
  FractionManMade = (manmadelength)/((manmadelength) + (naturallength))
  return(c(FractionNatural, FractionManMade))

}

cat("County\t\tfraction natural\tfraction manmade\n")
```

```
## County       fraction natural    fraction manmade
```

```
for(county in ucounties) {
  cn = Fraction(county,gaz2)
  cat(county,"\t",cn, "\n")
}
```

```
## Monterey      0.3849765 0.6150235
## Fresno    0.4340528 0.5659472
## Del Norte     0.5810147 0.4189853
## Mono      0.5905689 0.4094311
## San Bernardino    0.3099698 0.6900302
## Kern      0.332799 0.667201
## Inyo      0.5813268 0.4186732
## Plumas    0.6440785 0.3559215
## Glenn     0.5210084 0.4789916
## Modoc     0.5998415 0.4001585
## Lassen    0.6070912 0.3929088
## San Luis Obispo   0.4206081 0.5793919
## Riverside     0.2289318 0.7710682
## San Diego     0.1973956 0.8026044
## Los Angeles   0.1179533 0.8820467
## Orange    0.0884053 0.9115947
## Mendocino     0.6462901 0.3537099
## Sonoma    0.3415189 0.6584811
## Calaveras     0.366167 0.633833
## Napa      0.3797314 0.6202686
## Colusa    0.5636743 0.4363257
## Humboldt      0.6490551 0.3509449
## Merced    0.1451767 0.8548233
## Placer    0.4 0.6
## Kings     0.2656716 0.7343284
## Alpine    0.6395564 0.3604436
## Butte     0.3951947 0.6048053
## Imperial      0.1133036 0.8866964
## Contra Costa      0.1760933 0.8239067
## Yuba      0.3937622 0.6062378
## Yolo      0.3249581 0.6750419
## San Mateo     0.1917404 0.8082596
## Shasta    0.6181575 0.3818425
```

```
## Sierra    0.5140515 0.4859485
## Ventura   0.2572347 0.7427653
## El Dorado     0.4754286 0.5245714
## Santa Barbara     0.3639896 0.6360104
## Madera    0.4442524 0.5557476
## Lake      0.6338798 0.3661202
## Tuolumne      0.5754779 0.4245221
## Solano    0.2198758 0.7801242
## Tehama    0.6345291 0.3654709
## Santa Cruz    0.2906641 0.7093359
## Tulare    0.5071429 0.4928571
## Siskiyou      0.6944943 0.3055057
## Amador    0.3065693 0.6934307
## Sacramento    0.04281099 0.957189
## San Joaquin   0.1008534 0.8991466
## Trinity   0.7182576 0.2817424
## Alameda   0.08868185 0.9113182
## Santa Clara   0.1552774 0.8447226
## Nevada    0.4217877 0.5782123
## San Benito    0.5226917 0.4773083
## Sutter    0.2037037 0.7962963
## Mariposa      0.6079836 0.3920164
## Stanislaus    0.1613508 0.8386492
## Marin     0.3078024 0.6921976
## San Francisco     0.0864371 0.9135629
## NA    NaN NaN
```