HW 4

Joe Brew

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
#################
## Setwd
if ( Sys.info()["sysname"] == "Linux" ){
  setwd("/home/joebrew/")
} else {
  setwd("C:/Users/BrewJR/")
}
```

Read txt file (converted from dbf file) for Orlando

```
orlando <- read.csv("~/Documents/uf/phc6194/hw4/ORLANDO.txt")
```

Write a function geocode addresses using Google's API

```
#### This script uses RCurl and RJSONIO to download data from Google's API:
#### Latitude, longitude, location type (see explanation at the end), formatted address
#### Notice ther is a limit of 2,500 calls per day

library(RCurl)
library(RJSONIO)
library(plyr)

url <- function(address, return.call = "json", sensor = "false") {
  root <- "http://maps.google.com/maps/api/geocode/"
  u <- paste(root, return.call, "?address=", address, "&sensor=", sensor, sep = "")
  return(URLencode(u))
}

geoCode <- function(address,verbose=FALSE) {
  if(verbose) cat(address,"\n")
  u <- url(address)
  doc <- getURL(u)
  x <- fromJSON(doc,simplify = FALSE)
  if(x$status=="OK") {
    lat <- x$results[[1]]$geometry$location$lat
    lng <- x$results[[1]]$geometry$location$lng
    location_type <- x$results[[1]]$geometry$location_type
    formatted_address <- x$results[[1]]$formatted_address
    return(c(lat, lng, location_type, formatted_address))
  } else {
    return(c(NA,NA,NA, NA))
  }
}
```

Perform the geocode

```
# Use plyr to getgeocoding for a vector
address <- paste(orlando$FAC_ADDR,
                 orlando$FAC_CITY,
                 "Florida",
                 orlando$FAC_ZIP)
locations <- ldply(address, function(x) geoCode(x))
names(locations) <- c("lat","lon","location_type", "forAddress")

orlando <- cbind(orlando, locations)
```

Save the .rdata file of the geocoded addresses (so as to not have to repeat)

```
save.image("~/Documents/uf/phc6194/hw4/hw4.RData")
```

On subsequent runs, simply reload (rather than call the API again)

```
load("~/Documents/uf/phc6194/hw4/hw4.RData")
```

Clean up the dataframe a little bit

```
# save lat and lon as numeric objects
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
##
## The following objects are masked from 'package:stats':
##
##     filter, lag
##
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
orlando <-
  orlando %>%
  mutate(lat = as.numeric(lat),
         lon = as.numeric(lon))
```

Save a shapefile version or orlando

```
library(sp)
library(rgdal)
```

```
## rgdal: version: 0.9-1, (SVN revision 518)
## Geospatial Data Abstraction Library extensions to R successfully loaded
## Loaded GDAL runtime: GDAL 1.10.1, released 2013/08/26
## Path to GDAL shared files: /usr/share/gdal/1.10
## Loaded PROJ.4 runtime: Rel. 4.8.0, 6 March 2012, [PJ_VERSION: 480]
## Path to PROJ.4 shared files: (autodetected)
```

Keep only the non-NA's

```
orlando_sp <- orlando[which(is.finite(orlando$lat) &
                              is.finite(orlando$lon)),]
```

Convert to spatial points data frame projected in latitude and longitude

```
orlando_sp <- SpatialPointsDataFrame(orlando_sp[,c("lon", "lat")], orlando_sp,
                                     proj4string = CRS("+init=epsg:4326"))
```

Write the shapefile

```
writeOGR(orlando_sp,
         dsn = "~/Documents/uf/phc6194/hw4/orlando",
         layer = "orlando",
         driver = "ESRI Shapefile")
```

Read in the Orange county shapefiles from http://www.census.gov/cgi-bin/geo/shapefiles2010/main

```
# Read in orlando all lines (from http://www.census.gov/cgi-bin/geo/shapefiles2010/main)
#orlando_all <- readOGR("~/Documents/uf/phc6194/hw4", "tl_2010_12095_edges")

# Read in orlando roads only
#orlando_roads <- readOGR("~/Documents/uf/phc6194/hw4", "tl_2010_12095_roads")
```