**Using Google API to geocode a list of address – (gmap)**

#load up the ggmap library

install.packages("ggplot2")

install.packages("ggmap")

library(ggmap)

# get the input data

infile **<-** "ListAddress"

data **<-** read.csv(file="c:/\_data/nameoffile.csv", header=TRUE, sep=",")

# get the address list, and append "USA" to the end to increase accuracy

# (change or remove this if your address already include a country etc.)

addresses = data$Address

addresses = paste0(addresses, ", USA")

#define a function that will process googles server responses for us.

getGeoDetails **<-** **function**(address){

#use the gecode function to query google servers

geo\_reply = geocode(address, output='all', messaging=TRUE, override\_limit=TRUE)

#now extract the bits that we need from the returned list

answer **<-** data.frame(lat=NA, long=NA, accuracy=NA, formatted\_address=NA, address\_type=NA, status=NA)

answer$status **<-** geo\_reply$status

#if we are over the query limit - want to pause for an hour

**while**(geo\_reply$status == "OVER\_QUERY\_LIMIT"){

print("OVER QUERY LIMIT - Pausing for 1 hour at:")

time **<-** Sys.time()

print(as.character(time))

Sys.sleep(60\*60)

geo\_reply = geocode(address, output='all', messaging=TRUE, override\_limit=TRUE)

answer$status **<-** geo\_reply$status

}

#return Na's if we didn't get a match:

**if** (geo\_reply$status != "OK"){

return(answer)

}

#else, extract what we need from the Google server reply into a dataframe:

answer$lat **<-** geo\_reply$results[[1]]$geometry$location$lat

answer$long **<-** geo\_reply$results[[1]]$geometry$location$lng

**if** (length(geo\_reply$results[[1]]$types) > 0){

answer$accuracy **<-** geo\_reply$results[[1]]$types[[1]]

}

answer$address\_type **<-** paste(geo\_reply$results[[1]]$types, collapse=',')

answer$formatted\_address **<-** geo\_reply$results[[1]]$formatted\_address

return(answer)

}

#initialise a dataframe to hold the results

geocoded **<-** data.frame()

# find out where to start in the address list (if the script was interrupted before):

startindex **<-** 1

#if a temp file exists - load it up and count the rows!

tempfilename **<-** paste0(infile, '\_temp\_geocoded.rds')

**if** (file.exists(tempfilename)){

print("Found temp file - resuming from index:")

geocoded **<-** readRDS(tempfilename)

startindex **<-** nrow(geocoded)

print(startindex)

}

# Start the geocoding process - address by address. geocode() function takes care of query speed limit.

**for** (ii **in** seq(startindex, length(addresses))){

print(paste("Working on index", ii, "of", length(addresses)))

#query the google geocoder - this will pause here if we are over the limit.

result = getGeoDetails(addresses[ii])

print(result$status)

result$index **<-** ii

#append the answer to the results file.

geocoded **<-** rbind(geocoded, result)

#save temporary results as we are going along

saveRDS(geocoded, tempfilename)

}

#now we add the latitude and longitude to the main data

data$lat **<-** geocoded$lat

data$long **<-** geocoded$long

data$accuracy **<-** geocoded$accuracy

#write it all to the output files

saveRDS(data, paste0("c:/\_data/patch/”, infile ,"\_geocoded.rds"))

write.table(data, file=paste0("c:/\_data/patch/", infile ,"\_geocoded.csv"), sep=",", row.names=FALSE)