City name challenge

The challenge

Are there more cities with UK names on the east coast of the US or on the west coast of the US?
Dataset: http://download.maxmind.com/download/worldcities/worldcitiespop.txt.gz

Importing data

First we import the data from the file 'worldcitiespop.txt' into a dataframe:

```r
complete <- read.csv("worldcitiespop.txt", stringsAsFactors = FALSE)
```

Then we extract and save the relevant subsets, US cities and UK cities:

```r
complete_us=complete[complete$Country=="us",]
save(complete_us,file="data_us.Rda")
complete_uk=complete[complete$Country=="gb",]
save(complete_uk,file="data_uk.Rda")
```

After saving, the data can be quickly loaded:

```r
load("data_us.Rda")
load("data_uk.Rda")
```

Cleaning data

Before cleaning, we create a backup copy of the data that will be modified:

```r
complete_us_orig=complete_us
```

Some city names contain "bad" non-UTF-8 format. Check how many:

```
## Records with non UTF-8 city name format:
## in US
##         Country           City     AccentCity Region Population  Latitude
## 2912712      us      it\xfc`au      It\xfc`au     AS         NA -14.34778
## 2921665      us pi\xf1on hills Pi\xf1on Hills     CA         NA  34.43333
##         Longitude
## 2912712 -170.7664
## 2921665 -117.6458
## in UK
## [1] Country    City       AccentCity Region     Population Latitude
## [7] Longitude
## <0 rows> (or 0-length row.names)
```

There are few values; we just convert to NA:

```
complete_us$City=iconv(complete_us$City,from="",to="UTF-8")
```

Potentially we may want to filter data by population, but only $< 3\%$ of initial data would be left:

```
nrow(complete_us[!is.na(complete_us$Population) & complete_us$Population>0,])/nrow(complete_us_orig)
```

```
## [1] 0.02940369
```

For the moment we avoid population filtering:

```
cat("Percentage of initial data used: ", 100*nrow(complete_us)/nrow(complete_us_orig),"\n")
```

```
## Percentage of initial data used:  100
```

Elaborating data

First thing is to establish an UK name identity; there are many possible approaches.
An easy approach is just to get the list of UK city names from the dataset,

```
uk_names=unique(complete_uk$City)
```

and identify US city names exactly matching elements in this list:

```
us_cities_uk=complete_us[tolower(complete_us$City) %in% tolower(uk_names),]
```

A second method relies on using pattern matching, but it is much slower with the full dataset (~ 40 min on my laptop).

Results for east and west coasts

US states on the coasts according to Wikpedia definition (east coast set includes some states without shoreline, west coast set excludes Alaska):

```
east_us=c("FL","GA","SC","NC","VA","MD","DE","NJ","NY","CT","RI","MA","NH","ME","PA","DC","VT","WV");
west_us=c("CA","OR","WA");
```

Select cities in coast states:

```
west_us_cities=complete_us[tolower(complete_us$Region) %in% tolower(west_us),];
east_us_cities=complete_us[tolower(complete_us$Region) %in% tolower(east_us),];
```

Select cities with UK names in coast states:

```
west_us_cities_uk=us_cities_uk[tolower(us_cities_uk$Region) %in% tolower(west_us),];
east_us_cities_uk=us_cities_uk[tolower(us_cities_uk$Region) %in% tolower(east_us),];
```

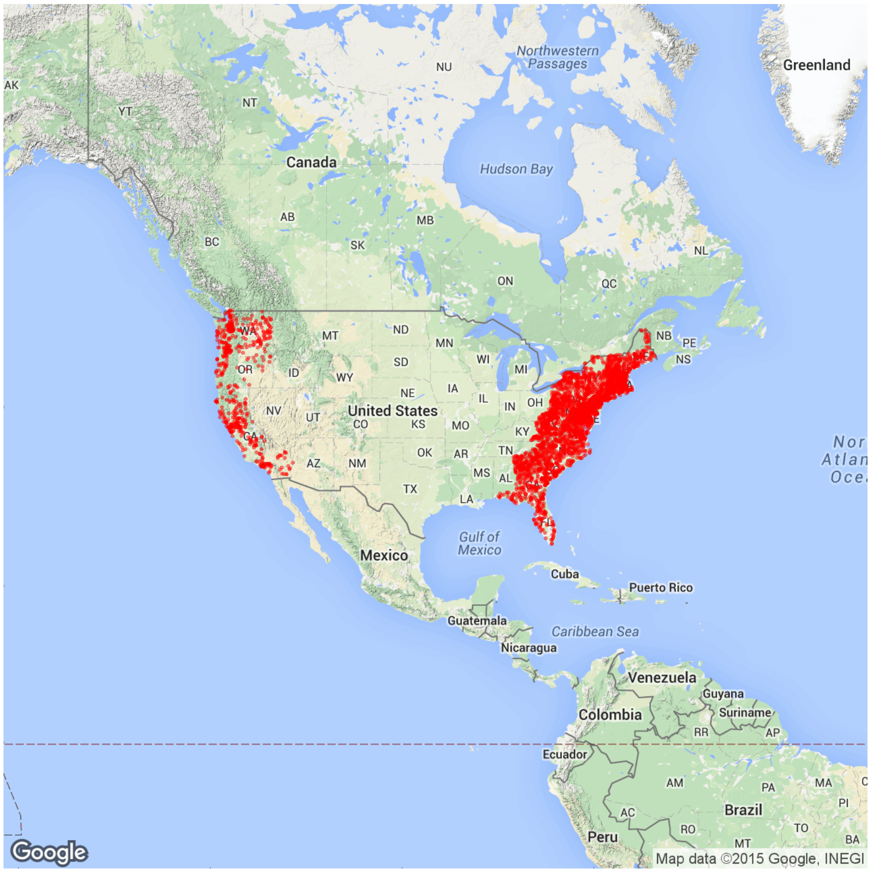
Print results:

```
##
##  Total number of cities in US:  141989
## Total number of cities with UK name in US:  9535
## Percentage of UK names in US:  0.06715309
##
##  Total number of cities in west coast:  9115
## Total number of cities with UK name in west coast:  553
## Percentage of UK names in west coast:  0.06066923
##
##  Total number of cities in east coast:  59887
## Total number of cities with UK name in east coast:  3610
## Percentage of UK names in east coast:  0.06028019
```

Plotting spatial data: cities with UK name

```
map <- qmap('US',zoom=3)
#plot the city points on top
map <- map + geom_point(data = east_us_cities_uk,
                        aes(x = Longitude, y = Latitude),
                        color="red", size=0.7, alpha=0.5) +
  geom_point(data = west_us_cities_uk,
             aes(x = Longitude, y = Latitude),
             color="red", size=0.7, alpha=0.5)

show(map)
```



Results for each state in US

We now perform an analysis at level of single state in US. First we get a list of the “continental” states in US:

```
us_states=unique(complete_us$Region)
us_states_extra=c("AS","GU","MP","PR","VI","UM","FM","MH","PW","HI")
us_states_50=us_states[!(us_states %in% us_states_extra)] # excluding Hawaii
```

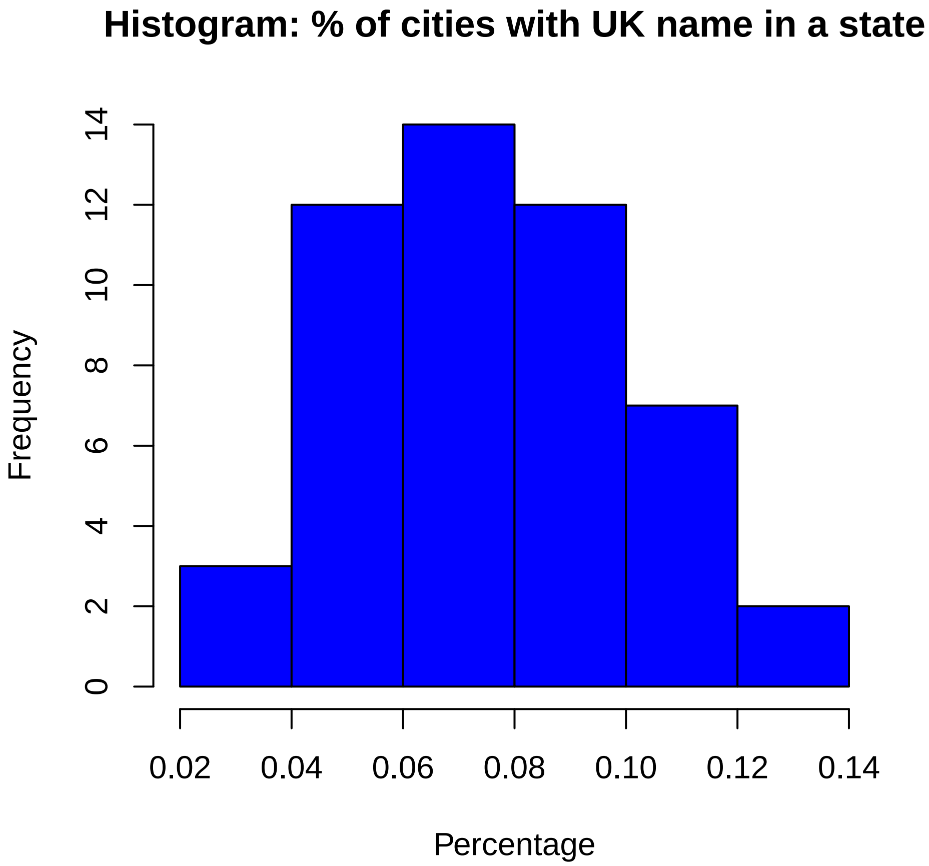
Then we generate a dataframe containing number of cities, number of cities with UK name, and percentage for each state:

```
dfStates <- all_states(complete_us,us_cities_uk,us_states_50)
head(dfStates)
```

```
##   code_state nrCities nrCitiesUK fractionUK
## 1         AL     4258        270 0.06341005
## 2         AK      677         23 0.03397341
## 3         AZ     1926         60 0.03115265
## 4         AR     3158        238 0.07536415
## 5         CA     5436        243 0.04470199
## 6         CO     1494        115 0.07697456
```

We can make a histogram:

```
hist(dfStates$fractionUK,
     main="Histogram: % of cities with UK name in a state",
     xlab="Percentage",
     col="blue")
```



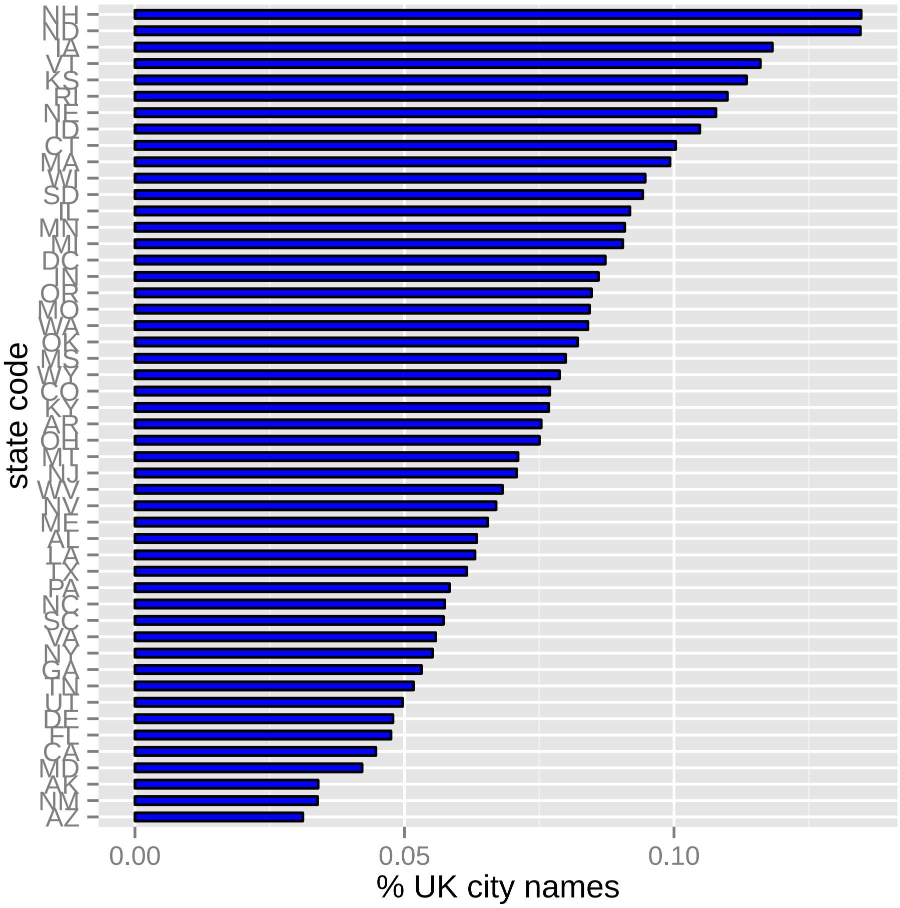
and give an estimate of the statistics for the states on the coasts:

```
mean_east <- mean(dfStates$fractionUK[dfStates$code_state %in% east_us])
sd_east <- sd(dfStates$fractionUK[dfStates$code_state %in% east_us])
mean_west <- mean(dfStates$fractionUK[dfStates$code_state %in% west_us])
```

```
##
##  Percentage of cities with UK name, state by state:
## East coast mean and sd:  0.07369821 0.02731128
## West coast mean:  0.07113871
```

Bar plot of states reordered according percentage of UK city names:

```
ggplot(dfStates, aes(x = reorder(code_state, fractionUK), y = fractionUK))+
  geom_bar(colour="black", fill="blue", width=0.5, position = position_dodge(width = 3), stat = "identi
  ylab("% UK city names") + xlab("state code") +
  coord_flip() +
  guides(fill=FALSE)
```



Many states in the southern parth of US have the lowest percentage; this could reflect the importance of Spaniard colonization in those states.

Appendix

Function "all_states", to generate the dataframe for each US state:

```r
all_states <- function(total_us,total_us_uk,us_states){
  #compute nr of cities, uk cities and percentage for each state
  nr_cities=vector('numeric')
  nr_cities_uk=vector('numeric')
  percent=vector('numeric')
  for (i in (1:length(us_states))){
    state=us_states[i]
    nr_cities_uk[i]=nrow(total_us_uk[tolower(total_us_uk$Region)==tolower(state),])
    nr_cities[i]=nrow(total_us[tolower(total_us$Region)==tolower(state),])
    percent[i]=nr_cities_uk[i]/nr_cities[i]
  }
  dfStates=data.frame(code_state=us_states,
                      nrCities=nr_cities,
                      nrCitiesUK=nr_cities_uk,
                      fractionUK=percent)
  return(dfStates)
}
```