beers maps and scatterplots

Ida Johnsson

1/22/2017

```
knitr::opts_chunk$set(echo = TRUE)

rm(list=ls())
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.2.4
```

```
library(ggmap)
library(maps)
```

```
## Warning: package 'maps' was built under R version 3.2.5
```

```
library(mapdata)
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
```

```
library(sp)
```

```
## Warning: package 'sp' was built under R version 3.2.5
```

```
library(maptools)
```

```
## Checking rgeos availability: TRUE
```

```
# the following function is thanks to
# https://favorableoutcomes.wordpress.com/2012/10/19/create-an-r-function-to-convert-state-codes-to-ful
stateFromLower <-function(x) {
  #read 52 state codes into local variable [includes DC (Washington D.C. and PR (Puerto Rico)]
  st.codes<-data.frame(
    state=as.factor(c("AK", "AL", "AR", "AZ", "CA", "CO", "CT", "DC", "DE", "FL", "GA",
                      "HI", "IA", "ID", "IL", "IN", "KS", "KY", "LA", "MA", "MD", "ME",
                      "MI", "MN", "MO", "MS",  "MT", "NC", "ND", "NE", "NH", "NJ", "NM",
```

```
                      "NV", "NY", "OH", "OK", "OR", "PA", "PR", "RI", "SC", "SD", "TN",
                      "TX", "UT", "VA", "VT", "WA", "WI", "WV", "WY")),
    full=as.factor(c("alaska","alabama","arkansas","arizona","california","colorado",
                     "connecticut","district of columbia","delaware","florida","georgia",
                     "hawaii","iowa","idaho","illinois","indiana","kansas","kentucky",
                     "louisiana","massachusetts","maryland","maine","michigan","minnesota",
                     "missouri","mississippi","montana","north carolina","north dakota",
                     "nebraska","new hampshire","new jersey","new mexico","nevada",
                     "new york","ohio","oklahoma","oregon","pennsylvania","puerto rico",
                     "rhode island","south carolina","south dakota","tennessee","texas",
                     "utah","virginia","vermont","washington","wisconsin",
                     "west virginia","wyoming"))
  )
  #create an nx1 data.frame of state codes from source column
  st.x<-data.frame(state=x)
  #match source codes with codes from 'st.codes' local variable and use to return the full state name
  refac.x<-st.codes$full[match(st.x$state,st.codes$state)]
  #return the full state names in the same order in which they appeared in the original source
  return(refac.x)

}


beers<-read.csv("~/Dropbox/ML/data/craft-cans/beers.csv")
brws<-read.csv("~/Dropbox/ML/data/craft-cans/breweries.csv")

d<-merge(beers, brws, by.x = "brewery_id", by.y = "row.names")
names(d)[names(d)=="name.x"]<-"beer"
names(d)[names(d)=="name.y"]<-"brewery"

# Summary Statistics
# plot average abv and ibu by style
states<-map_data("state")
head(states)
```

```
##        long      lat group order  region subregion
## 1 -87.46201 30.38968     1     1 alabama      <NA>
## 2 -87.48493 30.37249     1     2 alabama      <NA>
## 3 -87.52503 30.37249     1     3 alabama      <NA>
## 4 -87.53076 30.33239     1     4 alabama      <NA>
## 5 -87.57087 30.32665     1     5 alabama      <NA>
## 6 -87.58806 30.32665     1     6 alabama      <NA>
```

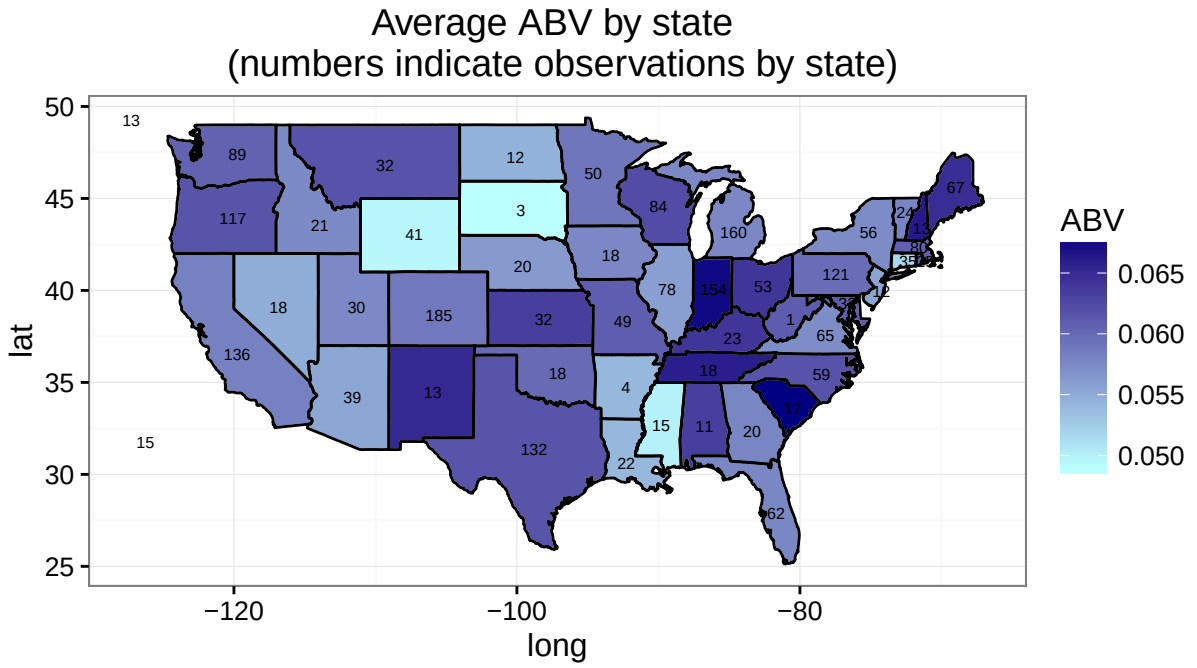
```
d$region<-stateFromLower(d$state)
# we see that this doesn't work because there is an extra space in the state abbrevations
levels(d$state)
```

```
##  [1] " AK" " AL" " AR" " AZ" " CA" " CO" " CT" " DC" " DE" " FL" " GA"
## [12] " HI" " IA" " ID" " IL" " IN" " KS" " KY" " LA" " MA" " MD" " ME"
## [23] " MI" " MN" " MO" " MS" " MT" " NC" " ND" " NE" " NH" " NJ" " NM"
## [34] " NV" " NY" " OH" " OK" " OR" " PA" " RI" " SC" " SD" " TN" " TX"
## [45] " UT" " VA" " VT" " WA" " WI" " WV" " WY"
```

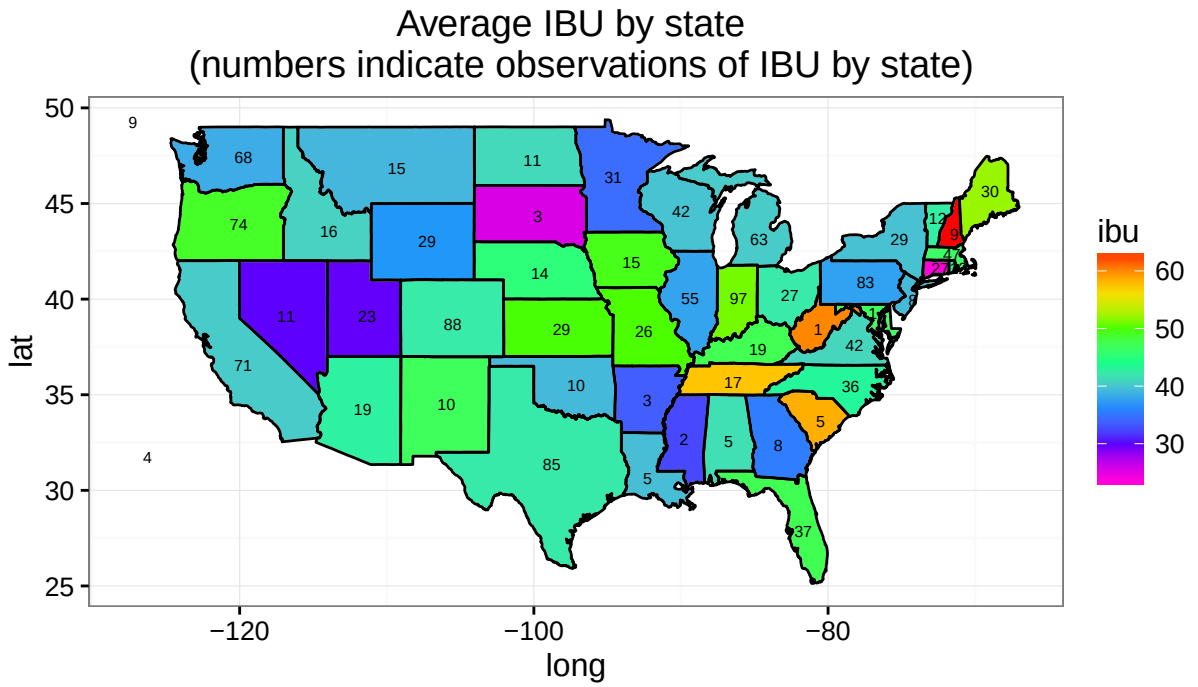
```
d$state<-gsub("[[:space:]]", "", d$state)
d$region<-stateFromLower(d$state)


agg<-aggregate(cbind(ibu,abv)~region,FUN = mean,d)

# text data for maps
counts<-as.data.frame(table(d$state)) # no. of observations per state
d.ibu<-d[!is.na(d$ibu),]
counts.ibu<-as.data.frame(table(d.ibu$state))
colnames(counts.ibu)<-c("state.abb","count.ibu")
colnames(counts)<-c("state.abb","count")
txt <- data.frame(state.center, state.abb)
d1<-txt
d2<-counts
d3<-counts.ibu
lab<-merge(d1,d2, by = "state.abb", all=FALSE)
lab<-merge(lab,d3,by="state.abb")
rm(counts,txt,d1,d2,d3)

plot.data <- inner_join(states, agg, by = "region")
```

```
## Warning in inner_join_impl(x, y, by$x, by$y): joining character vector and
## factor, coercing into character vector
```

```
# ABV
ggplot(data = plot.data, mapping = aes(x = long, y = lat, group = group)) +
  coord_fixed(1.3) +   geom_polygon(data = plot.data, aes(fill = abv), color = "white") +
  geom_polygon(color = "black", fill = NA) +theme_bw() +labs( title="Average ABV by state \n (numbers i
  scale_fill_gradientn("ABV",colors=c("#BBFFFF","#000080" ))+
  theme(axis.text = element_blank(),
        axis.line = element_blank(),
        axis.ticks = element_blank(),
        panel.border = element_blank(),
        panel.grid = element_blank(),
        axis.title = element_blank())+
   geom_text(data = lab, aes(x = x, y = y, label = count, group = NULL), size = 2)+theme_bw()
```



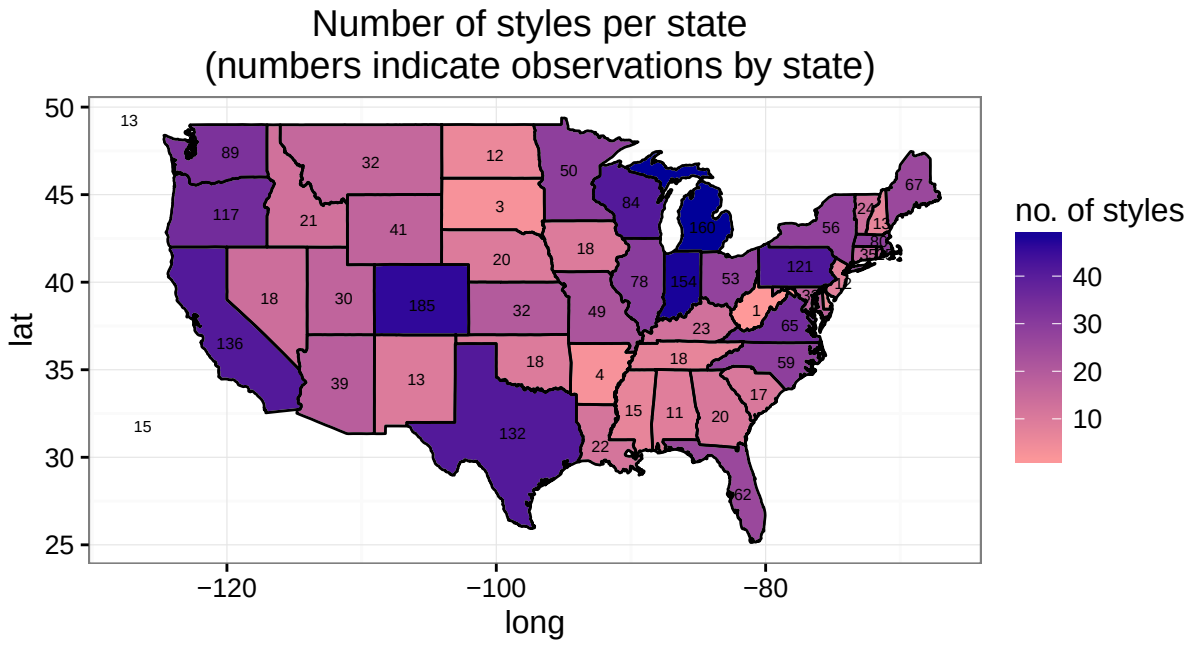

```
# IBU
ggplot(data = plot.data, mapping = aes(x = long, y = lat, group = group)) +
  coord_fixed(1.3) +   geom_polygon(data = plot.data, aes(fill = ibu), color = "white") +
  geom_polygon(color = "black", fill = NA) +theme_bw() +labs( title="Average IBU by state \n (numbers i
  scale_fill_gradientn(colours = rev(rainbow(7)))+
                       # breaks = c(2, 4, 10, 30, 50, 60),
                       # trans = "log10")+
  theme(axis.text = element_blank(),
        axis.line = element_blank(),
        axis.ticks = element_blank(),
        panel.border = element_blank(),
        panel.grid = element_blank(),
        axis.title = element_blank())+
  geom_text(data = lab, aes(x = x, y = y, label = count.ibu, group = NULL), size = 2)+theme_bw()
```




```
# number of styles per state
d$style<-as.character(d$style)
d.new<-within(d,{no.styles<-ave(style,region,FUN=function(x) length(unique(x)))})
agg<-subset(d.new,select=c("region","no.styles"))
agg<-unique(agg)
agg$no.styles<-as.numeric(paste(agg$no.styles))
plot.data <- inner_join(states, agg, by = "region")
```

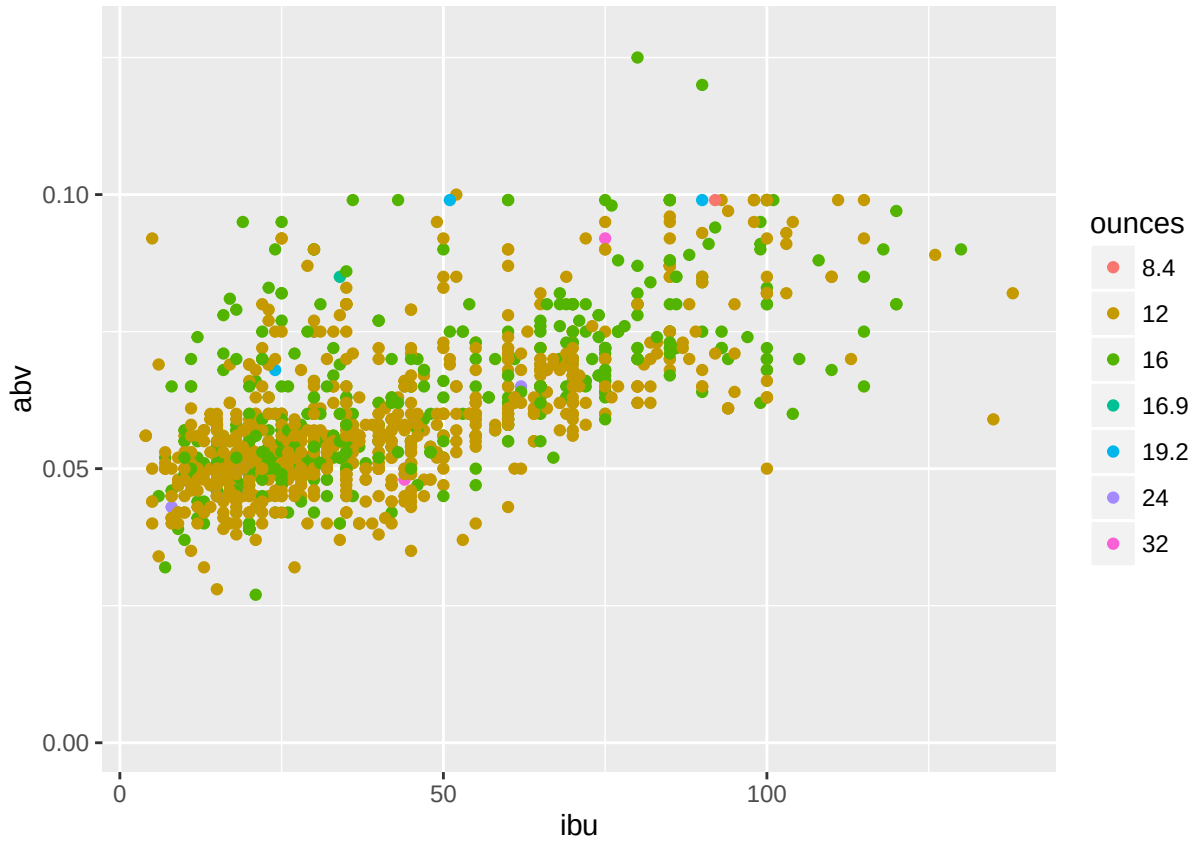
```
## Warning in inner_join_impl(x, y, by$x, by$y): joining character vector and
## factor, coercing into character vector
```

```
ggplot(data = plot.data, mapping = aes(x = long, y = lat, group = group)) +
  coord_fixed(1.3) +   geom_polygon(data = plot.data, aes(fill = no.styles), color = "white") +
  geom_polygon(color = "black", fill = NA) +theme_bw() +labs( title="Number of styles per state \n (numb
  scale_fill_gradientn("no. of styles",colors=c("#FF9999","#000099" ))+
  theme(axis.text = element_blank(),
        axis.line = element_blank(),
        axis.ticks = element_blank(),
        panel.border = element_blank(),
        panel.grid = element_blank(),
        axis.title = element_blank())+
  geom_text(data = lab, aes(x = x, y = y, label = count, group = NULL), size = 2)+theme_bw()
```




```
#### Relationship between ABV and IBU
d$ounces<-as.factor(d$ounces)
qplot(ibu,abv,data=d,color=ounces)
```

```
## Warning: Removed 1005 rows containing missing values (geom_point).
```



```r
#let's relabel 16.9 ounces as 16 ounces
d$ounces.new<-d$ounces
d$ounces.new[d$ounces==16.9]<-16
ggplot(data=d,aes(x=ibu,y=abv,color=ounces.new))+geom_point(size=1.5,alpha=.8)+
  scale_color_manual(breaks = c("8.4","12","16","19.2","24","32"),
                     values=c("red", "blue", "green","yellow","black","pink"))+theme_bw()
```

```
## Warning: Removed 1005 rows containing missing values (geom_point).
```