HW1 Advanced R

Calvin Makelky, A01872013

October 4, 2016

1

a

```
load(url("http://www.math.usu.edu/adele/IntroR/SFHousing.rda"))
sf = housing
names(sf)
```

```
##  [1] "county"  "city"    "zip"     "street"  "price"   "br"      "lsqft"
##  [8] "bsqft"   "year"    "date"    "long"    "lat"     "quality" "match"
## [15] "wk"
```

```
sapply(sf, function(x) sum(is.na(x)))
```

```
##  county    city     zip  street   price      br   lsqft   bsqft    year
##       0       0       5       0       0       0   21687     426    9202
##    date    long     lat quality   match      wk
##       0   23316   23316   23316   23316       0
```

The number of missing values for each variable is listed above.

b

```
tapply(sf$price, sf$county, median,  na.rm=TRUE)
```

```
##       Alameda County  Contra Costa County         Marin County
##               510000               466000               739000
##          Napa County San Francisco County     San Mateo County
##               505000               702000               700000
##   Santa Clara County        Solano County        Sonoma County
##               582000               380000               476500
```

The median housing price of each county is listed above.

c

```
citymean = tapply(sf$price, sf$city, mean,  na.rm=TRUE)
citysort = sort(citymean, decreasing=TRUE)
head(citysort, 10)
```

```
##   Los Altos Hills          Atherton      Hillsborough Belvedere/Tiburon
##           2393311           2379174           2354199           2217681
##         Belvedere              Ross            Diablo Belvedere/tiburon
##           2170088           2135883           1973025           1776572
##      Monte Sereno     Stinson Beach
##           1656639           1640469
```

The top ten most expensive cities based on mean housing price is listed above.

d

```
zips = as.numeric(as.character(sf$zip))
sanfran = function(x){
    if ( abs(94118-x)>=17 | is.na(x)==TRUE )
        return(FALSE)

    else
        return(TRUE)
}
SFZip = sapply(zips, sanfran)
length(SFZip[SFZip==TRUE])
```

```
## [1] 8134
```

There are 8134 observations that fall into the SF area.

e

```
tapply(sf$br, SFZip, mean)
```

```
##    FALSE     TRUE
## 3.043077 2.369560
```

On average there is about .7 less bedrooms in houses in San Fransisco than in other cities outside of San Fransisco.

2

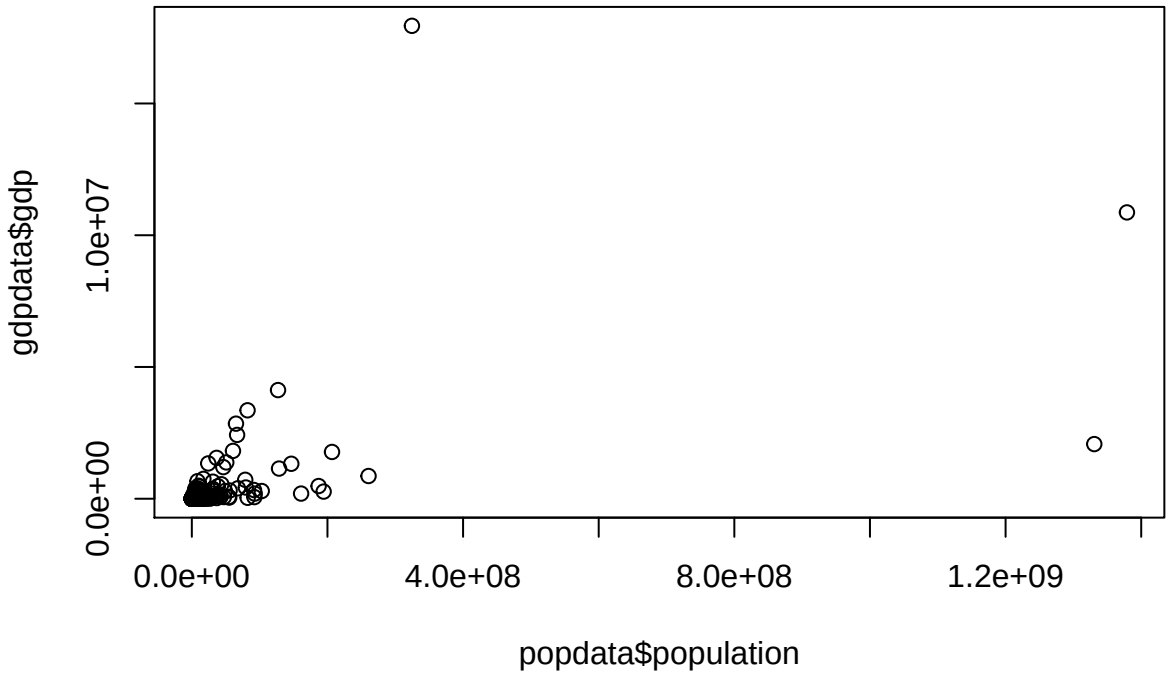
a

```
gdpdata= read.table("gdp.txt")
athdata= read.table("athletes2016.txt")
popdata= read.table("population.txt")
```

Gdp is a discrete quantitative variable. Population is a discrete quantitative variable. Total is a discrete quantitative variable.

b

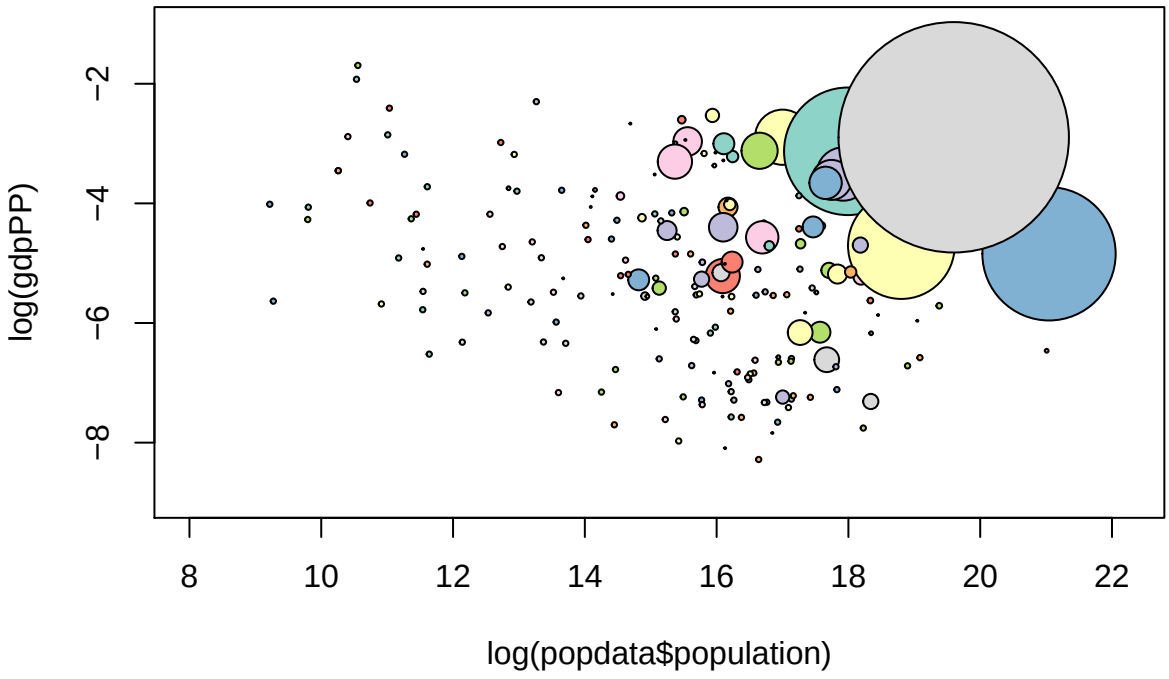
```
colnames(gdpdata) = c("country", "gdp")
athdata = athdata[,-(2:5)]
colnames(athdata) = c("country", "total")
colnames(popdata) = c("country", "population")
plot(popdata$population, gdpdata$gdp)
```



This plot violates all three properties of a good graph. 1) The data does not stand out, as it is in black and white. 2) The plot does not facilitate comparison because of the lack of color and more importantly how 3 super large or rich countries obscure the bulk of the data 3) The plot is not information rich at all. It lacks a title and a clear x and y axis title. There's also no caption to decribe what is going on. This is the gdp and population of what countries?

c

```
gdpPP = gdpdata$gdp / popdata$population
library(RColorBrewer)
mycolors = brewer.pal(9, "Set3")
symbols(log(popdata$population), log(gdpPP), circles=athdata$total,
        col=mycolors, bg=mycolors, inches=.6 )
```

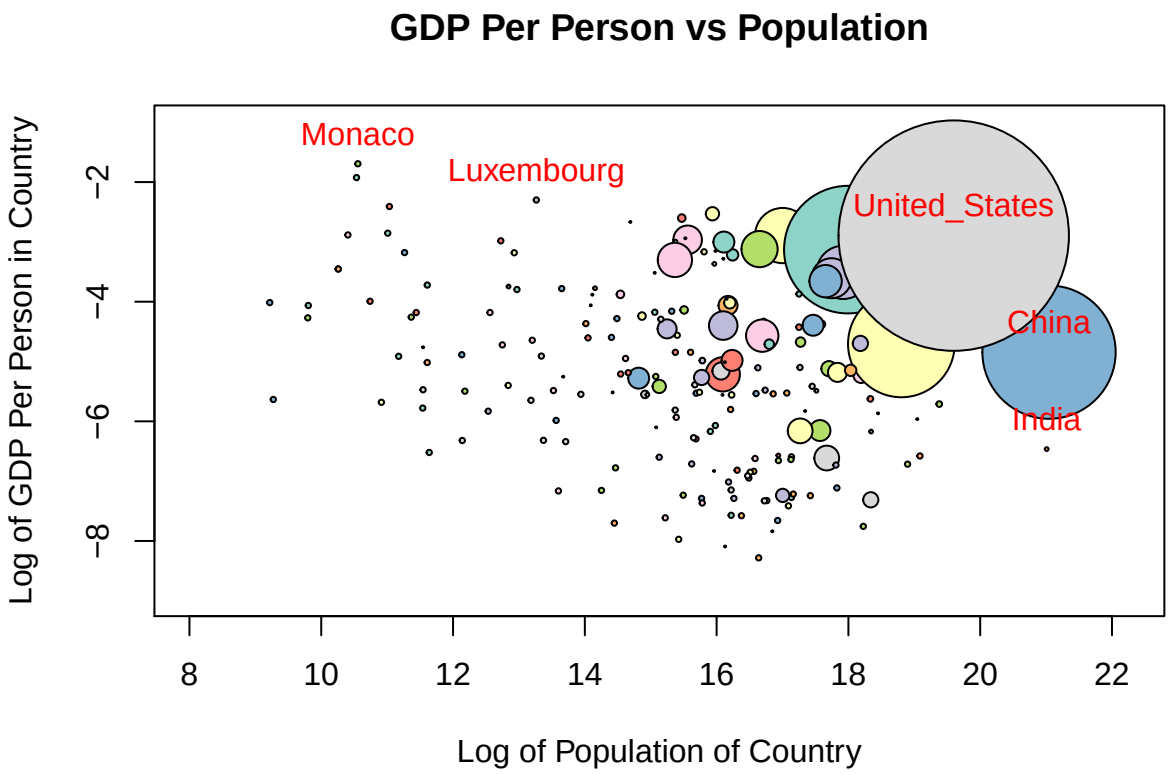


d

```
symbols(log(popdata$population), log(gdpPP), inches =.6,
circles=athdata$total, bg=mycolors, main="GDP Per Person vs Population",
xlab="Log of Population of Country", ylab="Log of GDP Per Person in Country")

countries5 = c(198, 41, 87, 111, 124)

text(log(popdata[countries5,2]), log(gdpPP[countries5]),
     popdata[countries5,1], cex=1, pos=3, col="red")
```



Seems to be a relationship between number of medals and population, while none between GDP per person and number of medals won.