Homework 5

Cody

October 19, 2017

```
suppressPackageStartupMessages(library(tidyverse))
suppressPackageStartupMessages(library(gapminder))
suppressPackageStartupMessages(library(knitr))
suppressPackageStartupMessages(library(dplyr))
suppressPackageStartupMessages(library(forcats))
```

```
## Warning: package 'forcats' was built under R version 3.4.2
```

```
suppressPackageStartupMessages(library(readxl))
```

```
## Warning: package 'readxl' was built under R version 3.4.2
```

```
suppressPackageStartupMessages(library(RColorBrewer))
```

Factor Management

Step goals:

- Define factor variables;
- Drop factor/levels;
- Reorder levels based on knowldge from data.

Gapminder version:

Drop Oceania Filter the Gapminder data to remove observations associated with the continent of Oceania. Additionally, remove unused factor levels. Provide concrete information on the data before and after removing these rows and Oceania; address the number of rows and the levels of the affected factors.

Dropping Oceania

This will drop any values in the continent column that take "Oceania".

```
gapminder_no_oceania <- gapminder %>%
  filter(continent != "Oceania")

summary(gapminder)
```

```
##         country        continent        year         lifeExp
##  Afghanistan:  12   Africa  :624   Min.   :1952   Min.   :23.60
##  Albania    :  12   Americas:300   1st Qu.:1966   1st Qu.:48.20
##  Algeria    :  12   Asia    :396   Median :1980   Median :60.71
##  Angola     :  12   Europe  :360   Mean   :1980   Mean   :59.47
##  Argentina  :  12   Oceania : 24   3rd Qu.:1993   3rd Qu.:70.85
##  Australia  :  12                  Max.   :2007   Max.   :82.60
##  (Other)    :1632
##       pop               gdpPercap
##  Min.   :6.001e+04   Min.   :   241.2
##  1st Qu.:2.794e+06   1st Qu.:  1202.1
##  Median :7.024e+06   Median :  3531.8
##  Mean   :2.960e+07   Mean   :  7215.3
##  3rd Qu.:1.959e+07   3rd Qu.:  9325.5
##  Max.   :1.319e+09   Max.   :113523.1
##
```

```
summary(gapminder_no_oceania)
```

```
##         country        continent        year         lifeExp
##  Afghanistan:  12   Africa  :624   Min.   :1952   Min.   :23.60
##  Albania    :  12   Americas:300   1st Qu.:1966   1st Qu.:48.08
##  Algeria    :  12   Asia    :396   Median :1980   Median :60.34
##  Angola     :  12   Europe  :360   Mean   :1980   Mean   :59.26
##  Argentina  :  12   Oceania :  0   3rd Qu.:1993   3rd Qu.:70.75
##  Austria    :  12                  Max.   :2007   Max.   :82.60
##  (Other)    :1608
##       pop              gdpPercap
##  Min.   :6.001e+04   Min.   :   241.2
##  1st Qu.:2.780e+06   1st Qu.:  1189.1
##  Median :7.024e+06   Median :  3449.5
##  Mean   :2.990e+07   Mean   :  7052.4
##  3rd Qu.:1.987e+07   3rd Qu.:  8943.2
##  Max.   :1.319e+09   Max.   :113523.1
##
```

In the summaries we see that the Oceania factor under continent previously had 24 entries, and now has 0.

Next we will drop the factor altogether,

```
gapminder_no_oceania <- gapminder_no_oceania %>%
  droplevels()
summary(gapminder)
```

```
##         country        continent        year         lifeExp
##  Afghanistan:  12   Africa  :624   Min.   :1952   Min.   :23.60
##  Albania    :  12   Americas:300   1st Qu.:1966   1st Qu.:48.20
##  Algeria    :  12   Asia    :396   Median :1980   Median :60.71
##  Angola     :  12   Europe  :360   Mean   :1980   Mean   :59.47
##  Argentina  :  12   Oceania : 24   3rd Qu.:1993   3rd Qu.:70.85
##  Australia  :  12                  Max.   :2007   Max.   :82.60
##  (Other)    :1632
##       pop              gdpPercap
##  Min.   :6.001e+04   Min.   :   241.2
##  1st Qu.:2.794e+06   1st Qu.:  1202.1
##  Median :7.024e+06   Median :  3531.8
##  Mean   :2.960e+07   Mean   :  7215.3
##  3rd Qu.:1.959e+07   3rd Qu.:  9325.5
##  Max.   :1.319e+09   Max.   :113523.1
##
```

```
summary(gapminder_no_oceania)
```

```
##         country        continent        year         lifeExp
##  Afghanistan:  12   Africa  :624   Min.   :1952   Min.   :23.60
##  Albania    :  12   Americas:300   1st Qu.:1966   1st Qu.:48.08
##  Algeria    :  12   Asia    :396   Median :1980   Median :60.34
##  Angola     :  12   Europe  :360   Mean   :1980   Mean   :59.26
##  Argentina  :  12                  3rd Qu.:1993   3rd Qu.:70.75
##  Austria    :  12                  Max.   :2007   Max.   :82.60
##  (Other)    :1608
##       pop              gdpPercap
##  Min.   :6.001e+04   Min.   :   241.2
##  1st Qu.:2.780e+06   1st Qu.:  1189.1
##  Median :7.024e+06   Median :  3449.5
##  Mean   :2.990e+07   Mean   :  7052.4
##  3rd Qu.:1.987e+07   3rd Qu.:  8943.2
##  Max.   :1.319e+09   Max.   :113523.1
##
```

Here we no longer have any trace of the Oceania continent and we can see that here.

```
levels(gapminder$continent)
```

```
## [1] "Africa"   "Americas" "Asia"     "Europe"   "Oceania"
```

```
levels(gapminder_no_oceania$continent)
```

```
## [1] "Africa"   "Americas" "Asia"     "Europe"
```

Reorder the levels of country or continent Use the forcats package to change the order of the factor levels, based on a principled summary of one of the quantitative variables. Consider experimenting with a summary statistic beyond the most basic choice of the median.

Here I use 'fct_reorder' to reorder by population. I also choose to use the "max" as a summary statistic, as I want to see the maximum over each year and order by this.

```
fct_reorder(gapminder_no_oceania$country,gapminder_no_oceania$pop,fun = max, .desc = TRUE) %>% levels()
  head()
```

```
## [1] "China"         "India"         "United States" "Indonesia"
## [5] "Brazil"        "Pakistan"
```

We can also order by other variables, like gdpPercap, where here it may make more sense to average our entires with mean.

```
fct_reorder(gapminder$country,gapminder$gdpPercap,fun = mean, .desc= TRUE) %>% levels() %>%
  head()
```

```
## [1] "Kuwait"        "Switzerland"   "Norway"        "United States"
## [5] "Canada"        "Netherlands"
```

Where we see that Kuwait is the country with the highest average gdpPercap!

For a little less of an interesting ordering, here are the continents ordered by average gdpPercap.

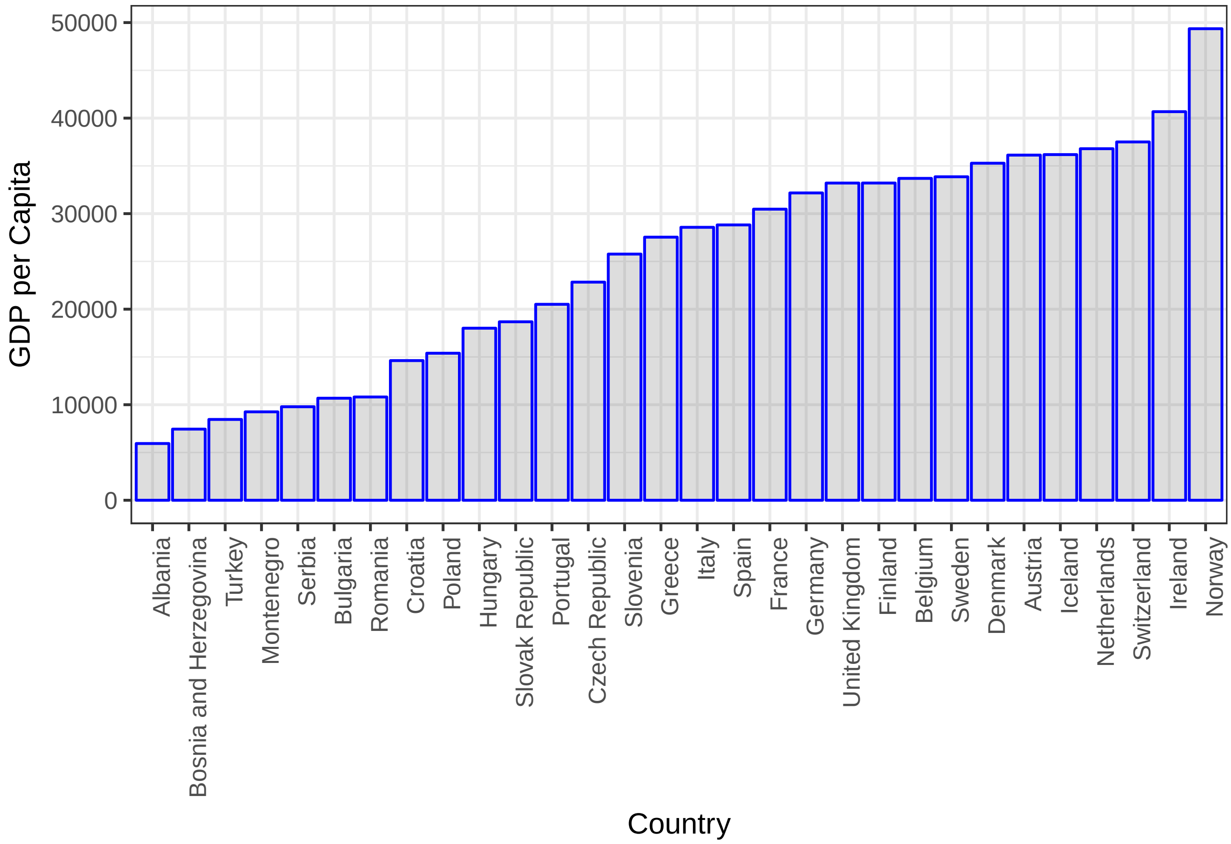
```
fct_reorder(gapminder_no_oceania$continent,gapminder_no_oceania$gdpPercap,fun = mean, .desc = TRUE) %>%
  head()
```

```
## [1] "Europe"   "Asia"     "Americas" "Africa"
```

With Europe leading the way in average gdpPercap.

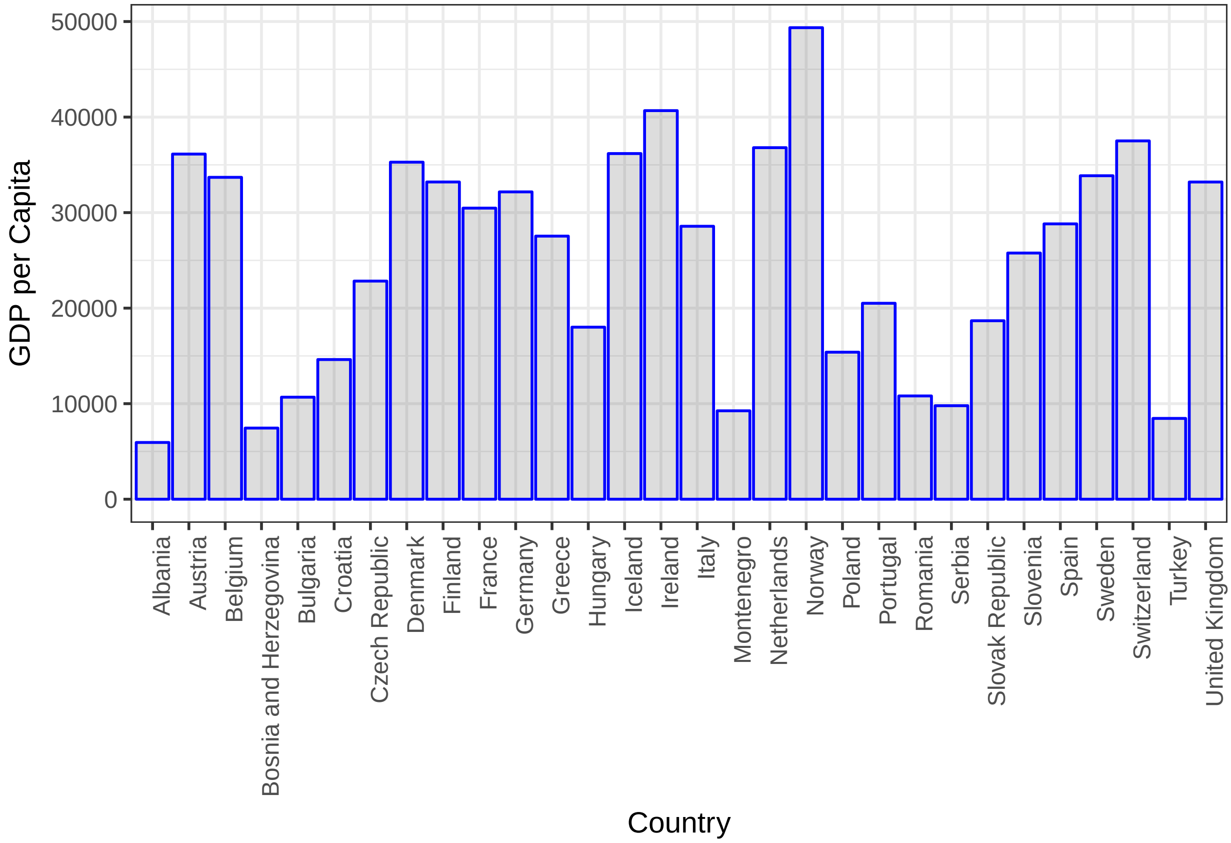
Inside of a ggplot, lets consider the most recent gdpPercap for Europe.

```
only_europe <- gapminder %>%
  filter(continent == "Europe",year == 2007) %>%
  droplevels()
goodplot <- only_europe %>%
  ggplot(aes(x=fct_reorder(country,gdpPercap),y=gdpPercap))+
  geom_col(color = "blue",alpha=.2)+
  theme_bw()+
  theme(axis.text.x=element_text(angle=90,hjust=1))+
  labs(x = "Country", y = "GDP per Capita")
goodplot
```



Now comparing this to an arrangement instead.

```
only_europe <- gapminder %>%
  filter(continent == "Europe",year == 2007) %>%
  droplevels()
badplot <- only_europe %>%
  arrange(gdpPercap) %>%
  ggplot(aes(x=country,y=gdpPercap))+
  geom_col(color = "blue",alpha=.2)+
  theme_bw()+
  theme(axis.text.x=element_text(angle=90,hjust=1))+
  labs(x = "Country", y = "GDP per Capita")
badplot
```



We can see this did not do what we intended for at all, where we asked the data to be arranged by gdpPercap and not alphebetically.

Just to check that things are indeed working like they're supposed to,

```
arranged_europe <- arrange(only_europe,gdpPercap)
 arranged_europe %>%
  glimpse() %>%
  knitr::kable()
```

```
## Observations: 30
## Variables: 6
## $ country   <fctr> Albania, Bosnia and Herzegovina, Turkey, Montenegro...
## $ continent <fctr> Europe, Europe, Europe, Europe, Europe, Europe, Eur...
## $ year      <int> 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007...
## $ lifeExp   <dbl> 76.423, 74.852, 71.777, 74.543, 74.002, 73.005, 72.4...
## $ pop       <int> 3600523, 4552198, 71158647, 684736, 10150265, 732285...
## $ gdpPercap <dbl> 5937.030, 7446.299, 8458.276, 9253.896, 9786.535, 10...
```

country	continent	year	lifeExp	pop	gdpPercap
Albania	Europe	2007	76.423	3600523	5937.030
Bosnia and Herzegovina	Europe	2007	74.852	4552198	7446.299
Turkey	Europe	2007	71.777	71158647	8458.276
Montenegro	Europe	2007	74.543	684736	9253.896
Serbia	Europe	2007	74.002	10150265	9786.535
Bulgaria	Europe	2007	73.005	7322858	10680.793
Romania	Europe	2007	72.476	22276056	10808.476
Croatia	Europe	2007	75.748	4493312	14619.223

country	continent	year	lifeExp	pop	gdpPercap
Poland	Europe	2007	75.563	38518241	15389.925
Hungary	Europe	2007	73.338	9956108	18008.944
Slovak Republic	Europe	2007	74.663	5447502	18678.314
Portugal	Europe	2007	78.098	10642836	20509.648
Czech Republic	Europe	2007	76.486	10228744	22833.309
Slovenia	Europe	2007	77.926	2009245	25768.258
Greece	Europe	2007	79.483	10706290	27538.412
Italy	Europe	2007	80.546	58147733	28569.720
Spain	Europe	2007	80.941	40448191	28821.064
France	Europe	2007	80.657	61083916	30470.017
Germany	Europe	2007	79.406	82400996	32170.374
United Kingdom	Europe	2007	79.425	60776238	33203.261
Finland	Europe	2007	79.313	5238460	33207.084
Belgium	Europe	2007	79.441	10392226	33692.605
Sweden	Europe	2007	80.884	9031088	33859.748
Denmark	Europe	2007	78.332	5468120	35278.419
Austria	Europe	2007	79.829	8199783	36126.493
Iceland	Europe	2007	81.757	301931	36180.789
Netherlands	Europe	2007	79.762	16570613	36797.933
Switzerland	Europe	2007	81.701	7554661	37506.419
Ireland	Europe	2007	78.885	4109086	40675.996
Norway	Europe	2007	80.196	4627926	49357.190

Which is proper, in a ggplot, it seems that the factor reordering is taking priority over the arrange function.

File I/O

Here I will save the ordered data Europe's 2007 entries.

```
write.csv(arranged_europe,"arranged_europe.csv")
```

Clearing the entire environment and loading this csv gives,

```
read_file <- read.csv("arranged_europe.csv")
```

Comparing the two,

```
head(arranged_europe)
```

```
## # A tibble: 6 x 6
##                  country continent  year lifeExp      pop gdpPercap
##                   <fctr>    <fctr> <int>   <dbl>    <int>     <dbl>
## 1                Albania    Europe  2007  76.423  3600523  5937.030
## 2 Bosnia and Herzegovina    Europe  2007  74.852  4552198  7446.299
## 3                 Turkey    Europe  2007  71.777 71158647  8458.276
## 4             Montenegro    Europe  2007  74.543   684736  9253.896
## 5                 Serbia    Europe  2007  74.002 10150265  9786.535
## 6               Bulgaria    Europe  2007  73.005  7322858 10680.793
```

```
head(read_file)
```

```
##   X                country continent year lifeExp      pop gdpPercap
## 1 1                Albania    Europe 2007  76.423  3600523  5937.030
## 2 2 Bosnia and Herzegovina    Europe 2007  74.852  4552198  7446.299
## 3 3                 Turkey    Europe 2007  71.777 71158647  8458.276
## 4 4             Montenegro    Europe 2007  74.543   684736  9253.896
## 5 5                 Serbia    Europe 2007  74.002 10150265  9786.535
## 6 6               Bulgaria    Europe 2007  73.005  7322858 10680.793
```

Here we note that an extra column was added that is a duplicate of the row numbers. Outside of this the two are identical.

Writing figures to file

Use `ggsave()` to explicitly write a figure to file. Then use `![Alt text](/path/to/img.png)` to embed it in your report. Things to play around with:

- Arguments of `ggsave()`, such as width, height, resolution or text scaling.
- Various graphics devices, e.g. a vector vs. raster format.
- Explicit provision of the plot object `p` via `ggsave(..., plot = p)`. Show a situation in which this actually matters.

```
ggsave("Euro_gdp_inc.pdf",goodplot,width=20,height=20,units="cm")
```

I now have a csv and a pdf on my local computer! Here is an embedding of the image My Plot