IMDB & Suicide dataset cleaning & visualization

Luke Geel

11/16/2020

Question 1 IMDb data We will work on an IMDb data. Please download the dataset imdb_titles.csv from Moodle and import it in R. (see https://www.imdb.com/interfaces/ for the meaning of the variables)

```
loadRData <- function(fileName){
  load(fileName)
  get(ls()[ls() != "fileName"])
}
imdb_titles <- read.csv("/Users/lukegeel/Downloads/imdb_titles.csv")
```

A. This dataset has already been cleaned. First, we need to have a brief summary of the dataset: 1. How many observations and variables in the dataset? 2. Are there any missing data, in which variable? Can we ignore them?

```
library(tidyverse)

names(imdb_titles)
```

```
## [1] "primaryTitle"   "titleType"      "startYear"      "endYear"
## [5] "runtimeMinutes" "genres"         "averageRating"  "numVotes"
```

```
dim(imdb_titles)
```

```
## [1] 8007    8
```

```
nrow(imdb_titles%>%filter(primaryTitle == "NA"))
```

```
## [1] 0
```

```
nrow(imdb_titles%>%filter(titleType == "NA"))
```

```
## [1] 0
```

```
nrow(imdb_titles%>%filter(startYear == "NA"))
```

```
## [1] 0
```

```
nrow(imdb_titles%>%filter(endYear == "NA"))
```

```
## [1] 0
```

```
nrow(imdb_titles%>%filter(runtimeMinutes == "NA"))
```

```
## [1] 0
```

```
nrow(imdb_titles%>%filter(genres == "NA"))
```

```
## [1] 0
```

```r
nrow(imdb_titles%>%filter(averageRating == "NA"))
```

```
## [1] 0
```

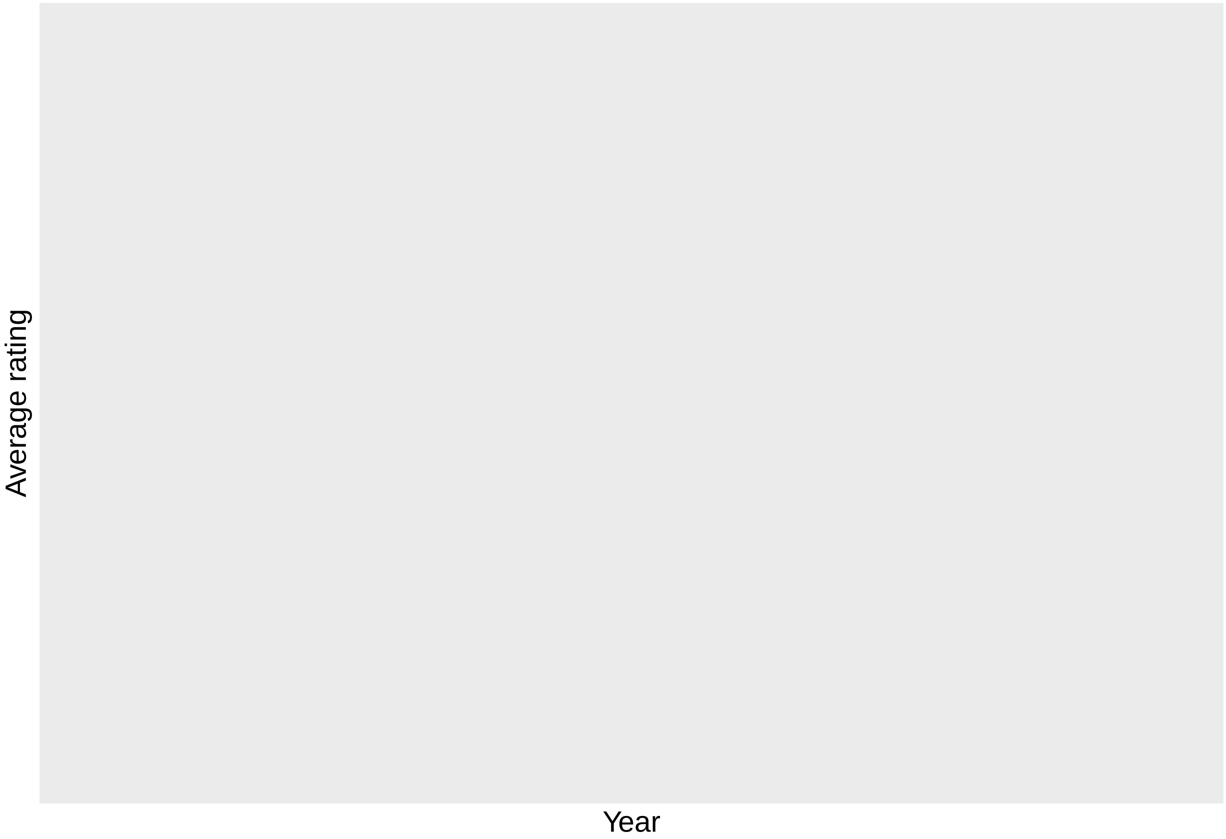
```r
nrow(imdb_titles%>%filter(numVotes == "NA"))
```

```
## [1] 0
```

This dataset has 8007 observations with 8 variables. The only variable that has missing data is endYear which we can ignore because the NA values in this variable signify that it's a movie and not a tv show.

B. Choose one categorical variable and one continuous variable to visualize their distributions and describe your findings (e.g., range, typical values, interesting patterns).

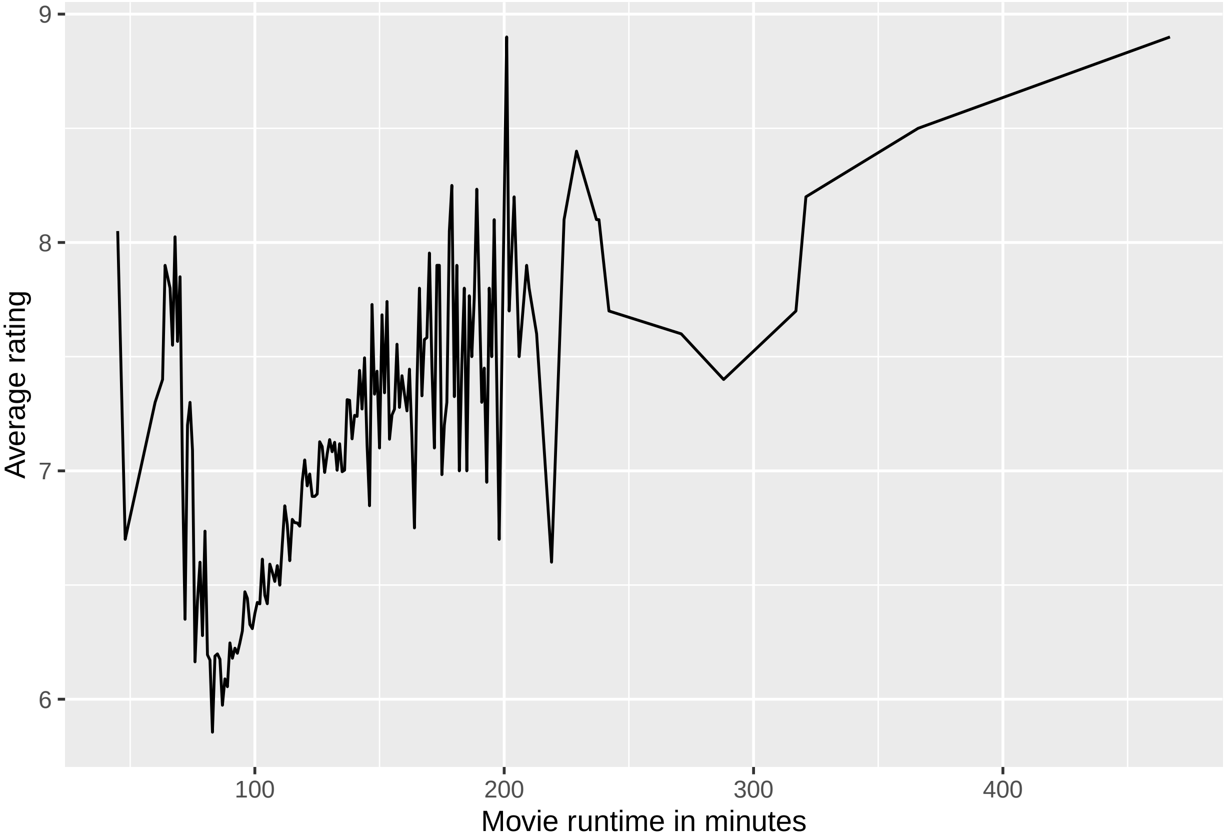
```r
imdb_titles %>%
  filter(endYear == "NA") %>%
  group_by(startYear)%>%
  summarize(avg_rating = mean(averageRating)) %>%
  ggplot()+
  geom_line(mapping=aes(x=startYear,y=avg_rating))+
  labs(x= "Year", y="Average rating")
```



This is a plot that compares the year that each movie came out with the average rating of each movie from that year. As you can see, the result is rather messy so I choose to split it into 3 categories: Before 1940, 1940-1975, and after 1975. For movies that came out before 1940, they had the highest average rating out of the 3 groups at around 8. Then for movies that came out from 1940-1975 the relationship between the year and the rating is completely linear, with movies closer to 1975 having a slightly lower rating. Then for movies that came out after 1975, the relationship isn't linear but it's clear that the average rating is far lower, roughly 6.5. At 1975, the average rating was 7.0, then it decreased to 6.5 by 2000, and since then it was increased slightly.

C. Choose two variables and visualize the relationship between them and describe your findings.

```
imdb_titles %>%
  filter(titleType == "movie") %>%
  group_by(runtimeMinutes)%>%
  summarize(avg_rating = mean(averageRating)) %>%
  ggplot()+
  geom_line(mapping=aes(x=runtimeMinutes,y=avg_rating))+
  labs(x= "Movie runtime in minutes", y="Average rating")
```

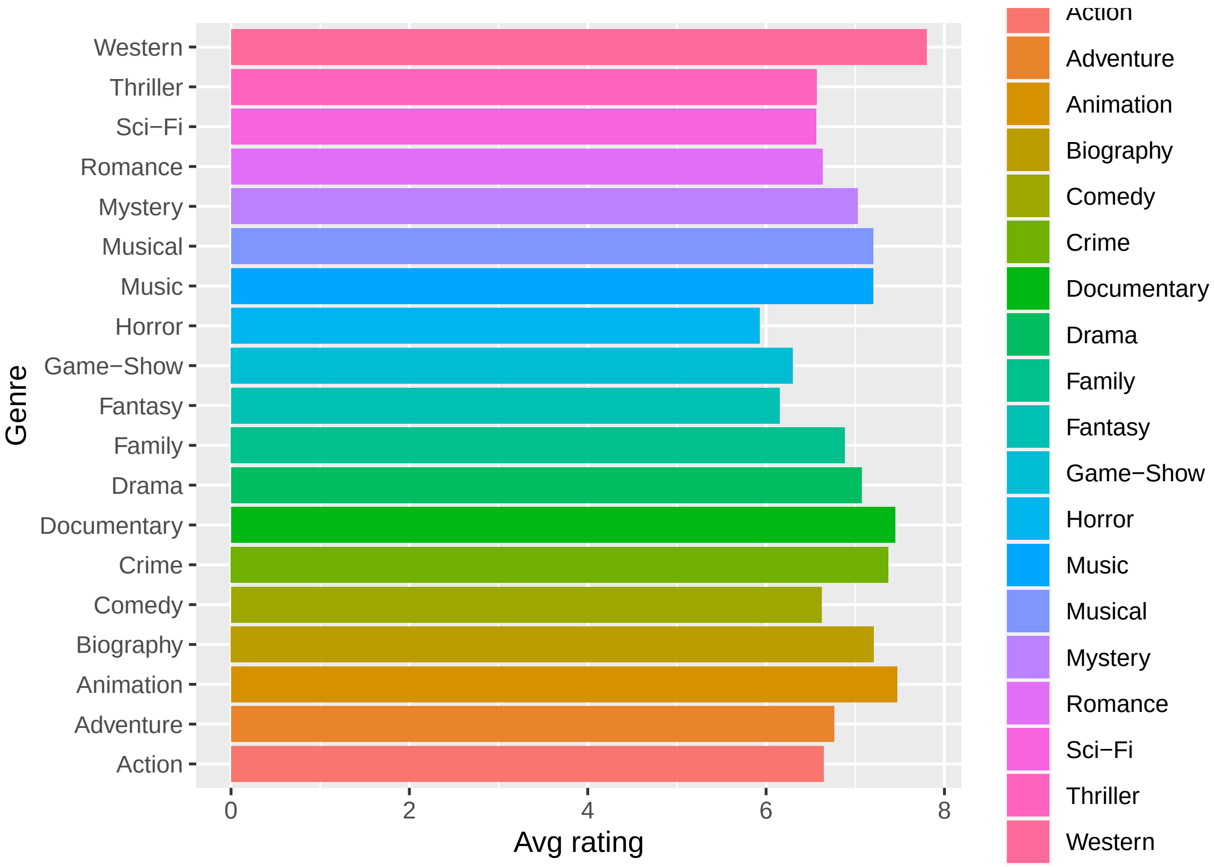


This plot compares the runtime of movies with the average rating for each runtime. The relationship betwene the two isn't exactly linear but it roughly follows a pattern where the longer the runtime the higher the average rating.

D. (optional) The genres variable is complicated because it can contain 1, 2, or 3 different tags. Can you figure out a way to deal with this complication: 1. How many different genres are there in the data? 2. Explore the relationship between genres and the ratings.

```
imdb_titles %>% separate(genres, into = c("genre","genre2","genre3"), sep=",") %>%
  group_by(genre)%>%
  summarize(avg_rating=mean(averageRating)) %>%
  ggplot()+
  geom_col(aes(y=genre,x=avg_rating,fill=genre))+
  labs(x="Avg rating",y="Genre")
```

```
## Warning: Expected 3 pieces. Missing pieces filled with `NA` in 2667 rows [1, 4,
## 5, 6, 8, 11, 16, 20, 28, 30, 31, 34, 35, 42, 53, 56, 58, 63, 67, 71, ...].
```



Question 2 Suicide data We will work on an suicide data from Kaggle. Please download the dataset master.csv from Moodle and import it in R. It is an overview of suicide rates from 1985 to 2016. (You can google to get the meaning of HDI and generation)

A. First, we need to do some data cleaning. 1. How many observations and variables in the dataset? 2. Are there any missing data, in which variable? Can we ignore them? 3. Is there any deterministic relationship between the variables? For example, year and country determines country-year

```r
master <- read.csv("/Users/lukegeel/Downloads/master.csv")
dim(master)
```

```
## [1] 27820    12
```

```r
names(master)
```

```
##  [1] "country"            "year"               "sex"
##  [4] "age"                "suicides_no"        "population"
##  [7] "suicides.100k.pop"  "country.year"       "HDI.for.year"
## [10] "gdp_for_year...."   "gdp_per_capita...." "generation"
```

```r
nrow(master%>%filter(is.na(`country`)))
```

```
## [1] 0
```

```r
nrow(master%>%filter(is.na(`year`)))
```

```
## [1] 0
```

```r
nrow(master%>%filter(is.na(`sex`)))
```

```
## [1] 0
```

```
nrow(master%>%filter(is.na(`age`)))
```

```
## [1] 0
```

```
nrow(master%>%filter(is.na(`suicides_no`)))
```

```
## [1] 0
```

```
nrow(master%>%filter(is.na(`population`)))
```

```
## [1] 0
```

```
nrow(master%>%filter(is.na(`suicides.100k.pop`)))
```

```
## [1] 0
```

```
nrow(master%>%filter(is.na(`country.year`)))
```

```
## [1] 0
```

```
nrow(master%>%filter(is.na(`HDI.for.year`)))
```

```
## [1] 19456
```

```
nrow(master%>%filter(is.na(`gdp_for_year....`)))
```

```
## [1] 0
```

```
nrow(master%>%filter(is.na(`gdp_per_capita....`)))
```

```
## [1] 0
```

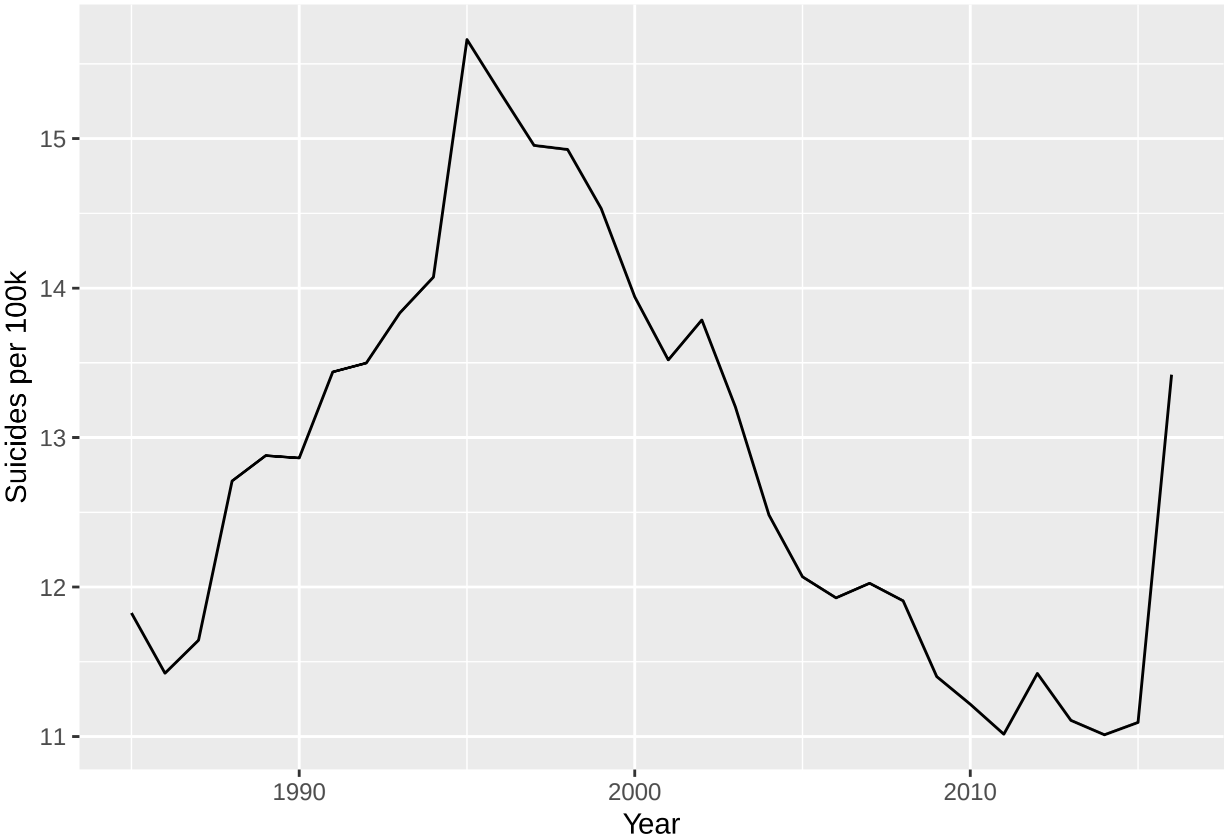
```
nrow(master%>%filter(is.na(`generation`)))
```

```
## [1] 0
```

This dataset has 27820 observations and 12 variables. The only variable that has NA values is HDI for year. These NA values indicate that the country doesn't have one of the following: life expectancy, education, and GNI index. I believe that we should ignore these observations as they won't provide as much data. Some variables have a deterministic relationship. Suicides/100k pop is just the number of suicides per 100k people. The country-year variable is just the country and year.

B. Visualize the time trend of global suicide. What do you find? (Think about which variable is better to describe the trend, suicide_no or suicides/100k pop)

```
master%>%group_by(year)%>%summarize(suicides_100k_avg = mean(`suicides.100k.pop`) )%>%
  ggplot()+ geom_line(aes(x=year, y=suicides_100k_avg))+
  labs(x="Year", y="Suicides per 100k")
```

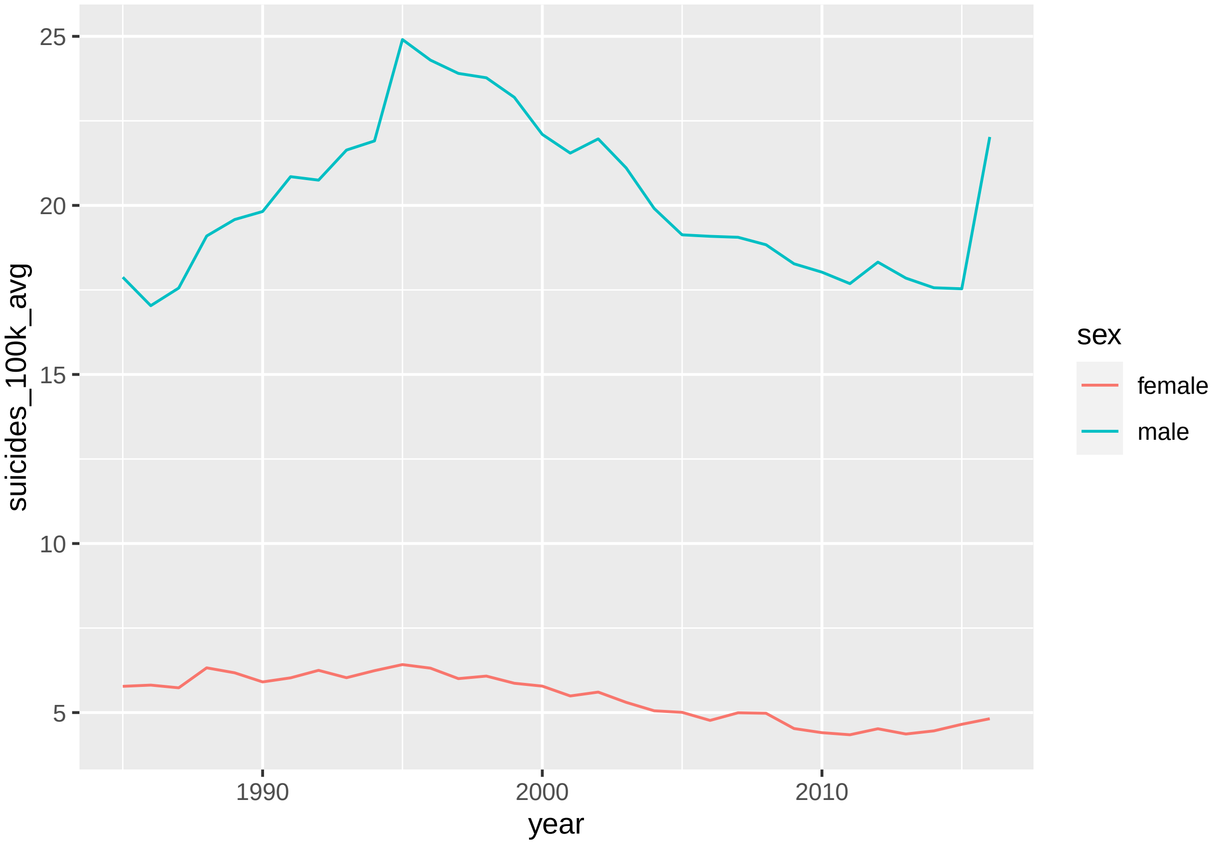


This plot visualizes the trend of suicides/100k population over time. As you can tell, from 1985 to 1995 the suicides per 100k population increased rather significantly. Since then, however, that number is decreasing and by 2010 it was lower than what it was in the year 1985.

C. Visualize the time trend of global suicide by sex and age groups. What do you find?

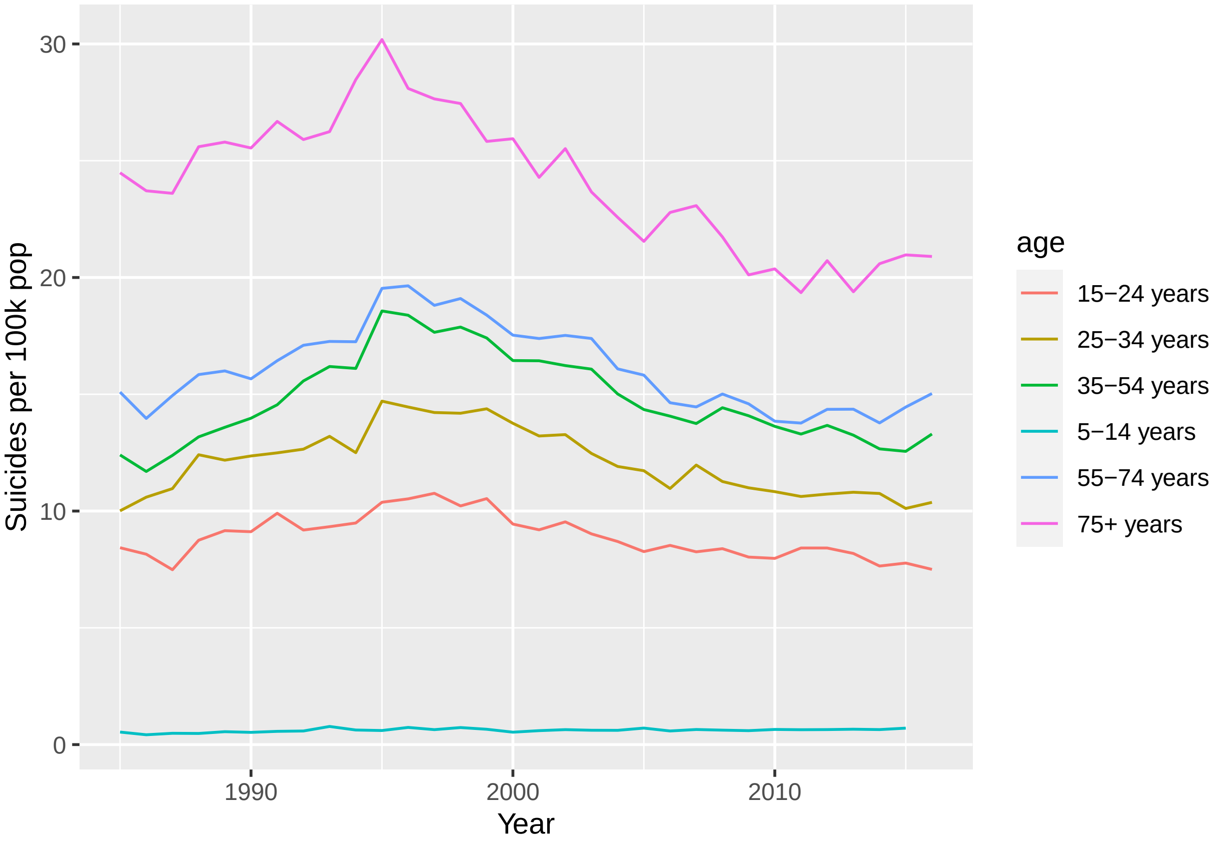
```
master%>%group_by(year, sex)%>%summarize(suicides_100k_avg = mean(`suicides.100k.pop`) )%>%
  ggplot()+ geom_line(aes(x=year, y=suicides_100k_avg, color = sex))
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```



```
master%>%group_by(year, age)%>%summarize(suicides_100k_avg = mean(`suicides.100k.pop`) )%>%
  ggplot()+ geom_line(aes(x=year, y=suicides_100k_avg, color = age))+
  labs(x="Year", y="Suicides per 100k pop")
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```



Looking at these 2 plots, there are some very interesting findings. Firstly, it's clear that males are far more likely, roughly 3x, to commit suicide than their female counterparts. Secondly, the proportion of suicides to population increases with age. So the older someone is the more likely they are to commit suicide.