Bio-494 Week 1

Noah

HEADER 1

HEADER 2

BOLD *ITALICS*

this is a list

- item 1
- item 2
- item 3

Any text you write outside of code "chunks" is just text. It is how you annotate the text.

lines 20-22 are a chunk of R code, bookended by the three back-ticks

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Here I am reading in a file.

```
movies_imdb=read_delim("movies/movies_imdb.txt",delim=",")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   color = col_character(),
##   director_name = col_character(),
##   actor_2_name = col_character(),
##   genres = col_character(),
##   actor_1_name = col_character(),
##   movie_title = col_character(),
##   actor_3_name = col_character(),
##   plot_keywords = col_character(),
##   movie_imdb_link = col_character(),
##   language = col_character(),
##   country = col_character(),
##   content_rating = col_character()
```

```
## )
## i Use `spec()` for the full column specifications.
```

```
movies_rottentom=read_delim("movies/movies_rottentom.txt",delim=",")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   title = col_character(),
##   metacritic = col_double(),
##   rotten_tomatoes = col_double()
## )
```

Including Plots

You can also embed plots, for example:

```
movies_imdb %>% select(movie_title,title_year,duration,imdb_score)
```

```
## # A tibble: 4,704 x 4
##    movie_title                                title_year duration imdb_score
##    <chr>                                           <dbl>    <dbl>      <dbl>
##  1 Avatar                                           2009      178        7.9
##  2 Pirates of the Caribbean: At World's End         2007      169        7.1
##  3 Spectre                                          2015      148        6.8
##  4 The Dark Knight Rises                            2012      164        8.5
##  5 John Carter                                      2012      132        6.6
##  6 Spider-Man 3                                     2007      156        6.2
##  7 Tangled                                          2010      100        7.8
##  8 Avengers: Age of Ultron                          2015      141        7.5
##  9 Harry Potter and the Half-Blood Prince           2009      153        7.5
## 10 Batman v Superman: Dawn of Justice               2016      183        6.9
## # ... with 4,694 more rows
```

```
imdb_minimal=movies_imdb %>% select(movie_title,title_year,duration,imdb_score)
```

```
imdb_minimal=movies_imdb %>% select(movie_title,title_year,duration,imdb_score)

joined=full_join(imdb_minimal,movies_rottentom,by=c("movie_title"="title"))
```

ggplot data

```
plot_imdb = ggplot(movies_imdb)
summary(plot_imdb)
```

```
## data: color, director_name, num_critic_for_reviews, duration,
##   director_facebook_likes, actor_3_facebook_likes, actor_2_name,
##   actor_1_facebook_likes, gross, genres, actor_1_name, movie_title,
##   num_voted_users, cast_total_facebook_likes, actor_3_name,
##   facenumber_in_poster, plot_keywords, movie_imdb_link,
##   num_user_for_reviews, language, country, content_rating, budget,
##   title_year, actor_2_facebook_likes, imdb_score, aspect_ratio,
##   movie_facebook_likes [4704x28]
## faceting: <ggproto object: Class FacetNull, Facet, gg>
##     compute_layout: function
##     draw_back: function
```

```
##     draw_front: function
##     draw_labels: function
##     draw_panels: function
##     finish_data: function
##     init_scales: function
##     map_data: function
##     params: list
##     setup_data: function
##     setup_params: function
##     shrink: TRUE
##     train_scales: function
##     vars: function
##     super:  <ggproto object: Class FacetNull, Facet, gg>
```

ggplot aesthetics

```
plot_imdb = ggplot(movies_imdb) + aes(x=title_year,y=imdb_score)
summary(plot_imdb)
```

```
## data: color, director_name, num_critic_for_reviews, duration,
##   director_facebook_likes, actor_3_facebook_likes, actor_2_name,
##   actor_1_facebook_likes, gross, genres, actor_1_name, movie_title,
##   num_voted_users, cast_total_facebook_likes, actor_3_name,
##   facenumber_in_poster, plot_keywords, movie_imdb_link,
##   num_user_for_reviews, language, country, content_rating, budget,
##   title_year, actor_2_facebook_likes, imdb_score, aspect_ratio,
##   movie_facebook_likes [4704x28]
## mapping:  x = ~title_year, y = ~imdb_score
## faceting: <ggproto object: Class FacetNull, Facet, gg>
##     compute_layout: function
##     draw_back: function
##     draw_front: function
##     draw_labels: function
##     draw_panels: function
##     finish_data: function
##     init_scales: function
##     map_data: function
##     params: list
##     setup_data: function
##     setup_params: function
##     shrink: TRUE
##     train_scales: function
##     vars: function
##     super:  <ggproto object: Class FacetNull, Facet, gg>
```

Add layers to a ggplot object with +

```
plot_imdb = ggplot(movies_imdb)
plot_imdb = plot_imdb + aes(x=title_year,y=imdb_score)
summary(plot_imdb)
```

```
## data: color, director_name, num_critic_for_reviews, duration,
##   director_facebook_likes, actor_3_facebook_likes, actor_2_name,
##   actor_1_facebook_likes, gross, genres, actor_1_name, movie_title,
##   num_voted_users, cast_total_facebook_likes, actor_3_name,
##   facenumber_in_poster, plot_keywords, movie_imdb_link,
```

```
##   num_user_for_reviews, language, country, content_rating, budget,
##   title_year, actor_2_facebook_likes, imdb_score, aspect_ratio,
##   movie_facebook_likes [4704x28]
## mapping:  x = ~title_year, y = ~imdb_score
## faceting: <ggproto object: Class FacetNull, Facet, gg>
##     compute_layout: function
##     draw_back: function
##     draw_front: function
##     draw_labels: function
##     draw_panels: function
##     finish_data: function
##     init_scales: function
##     map_data: function
##     params: list
##     setup_data: function
##     setup_params: function
##     shrink: TRUE
##     train_scales: function
##     vars: function
##     super:  <ggproto object: Class FacetNull, Facet, gg>
```

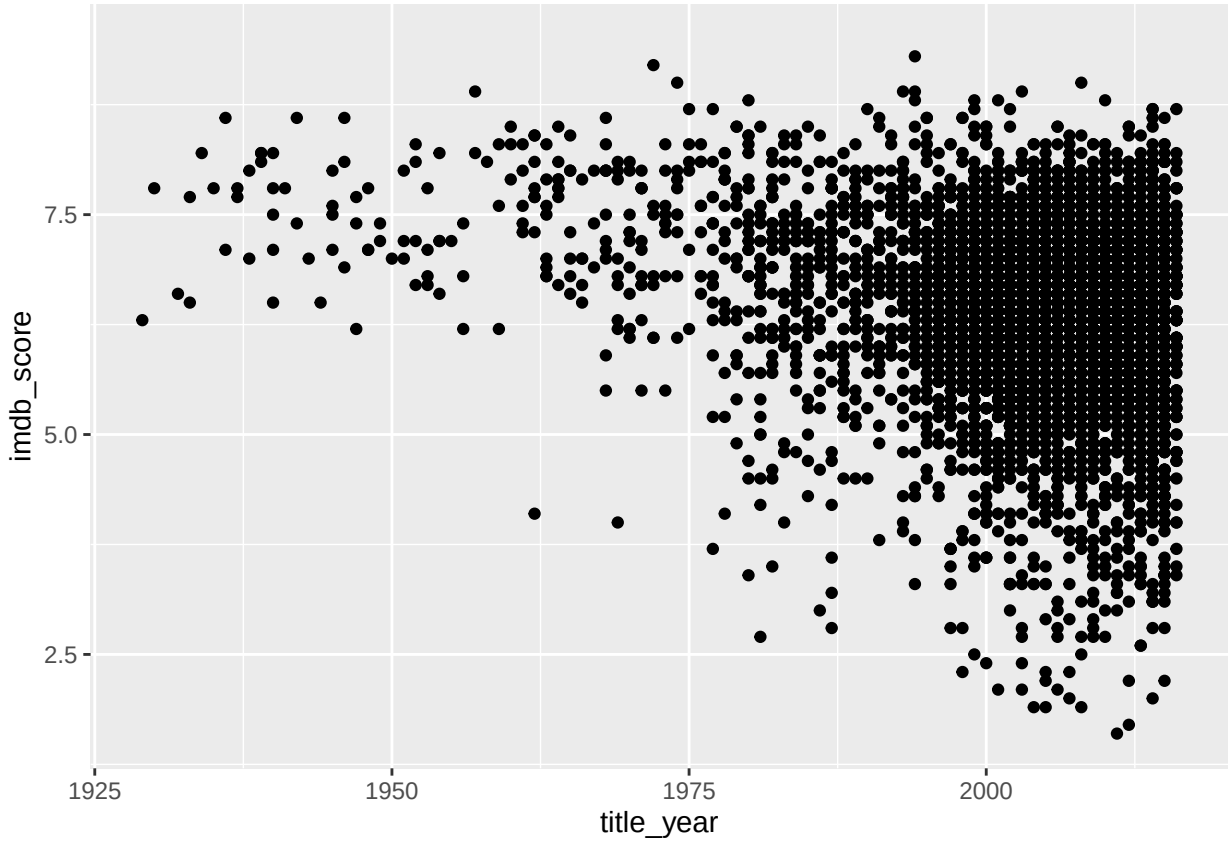
ggplot geoms

```
plot_imdb = plot_imdb + geom_point()
summary(plot_imdb)
```

```
## data: color, director_name, num_critic_for_reviews, duration,
##   director_facebook_likes, actor_3_facebook_likes, actor_2_name,
##   actor_1_facebook_likes, gross, genres, actor_1_name, movie_title,
##   num_voted_users, cast_total_facebook_likes, actor_3_name,
##   facenumber_in_poster, plot_keywords, movie_imdb_link,
##   num_user_for_reviews, language, country, content_rating, budget,
##   title_year, actor_2_facebook_likes, imdb_score, aspect_ratio,
##   movie_facebook_likes [4704x28]
## mapping:  x = ~title_year, y = ~imdb_score
## faceting: <ggproto object: Class FacetNull, Facet, gg>
##     compute_layout: function
##     draw_back: function
##     draw_front: function
##     draw_labels: function
##     draw_panels: function
##     finish_data: function
##     init_scales: function
##     map_data: function
##     params: list
##     setup_data: function
##     setup_params: function
##     shrink: TRUE
##     train_scales: function
##     vars: function
##     super:  <ggproto object: Class FacetNull, Facet, gg>
## -----------------------------------
## geom_point: na.rm = FALSE
## stat_identity: na.rm = FALSE
## position_identity
```

```
plot_imdb
```

```
## Warning: Removed 97 rows containing missing values (geom_point).
```

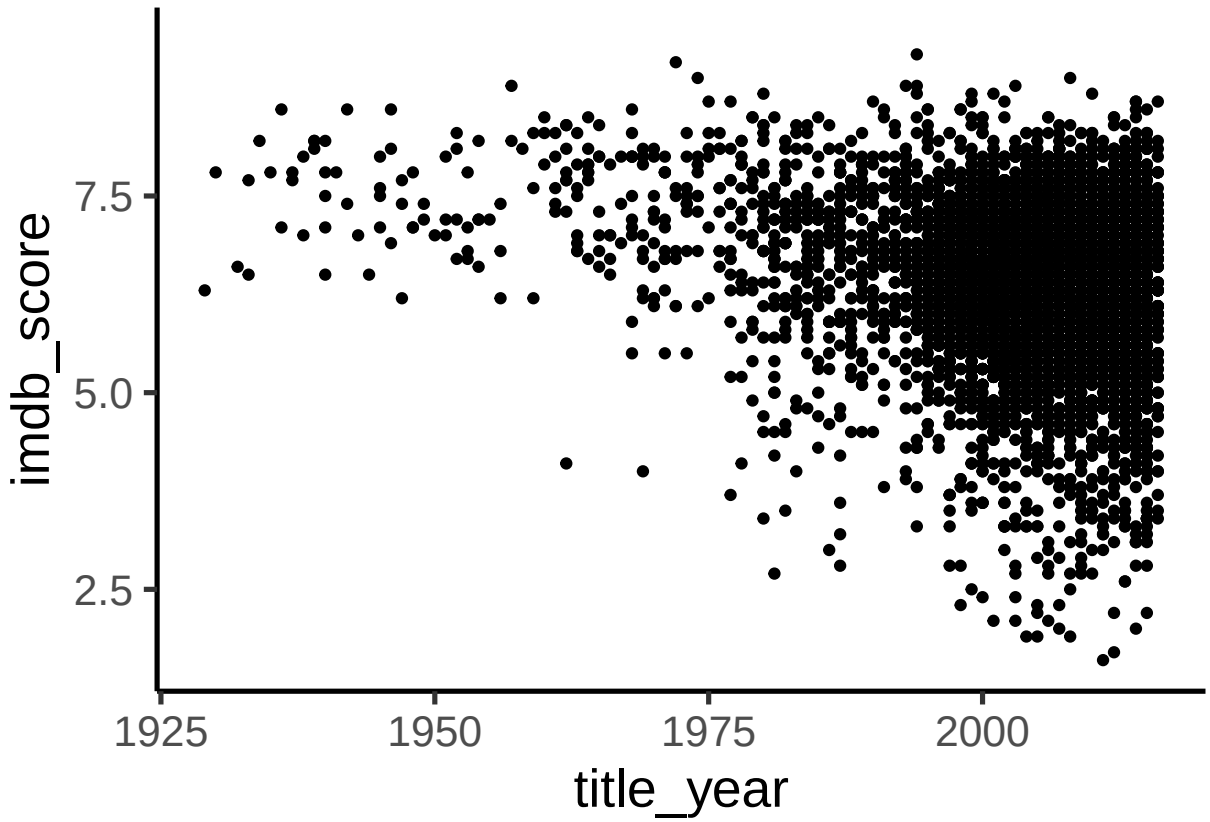


back to the slides for a second.

some nice default themes

```
plot_imdb = plot_imdb + theme_classic(base_size = 20)
plot_imdb
```

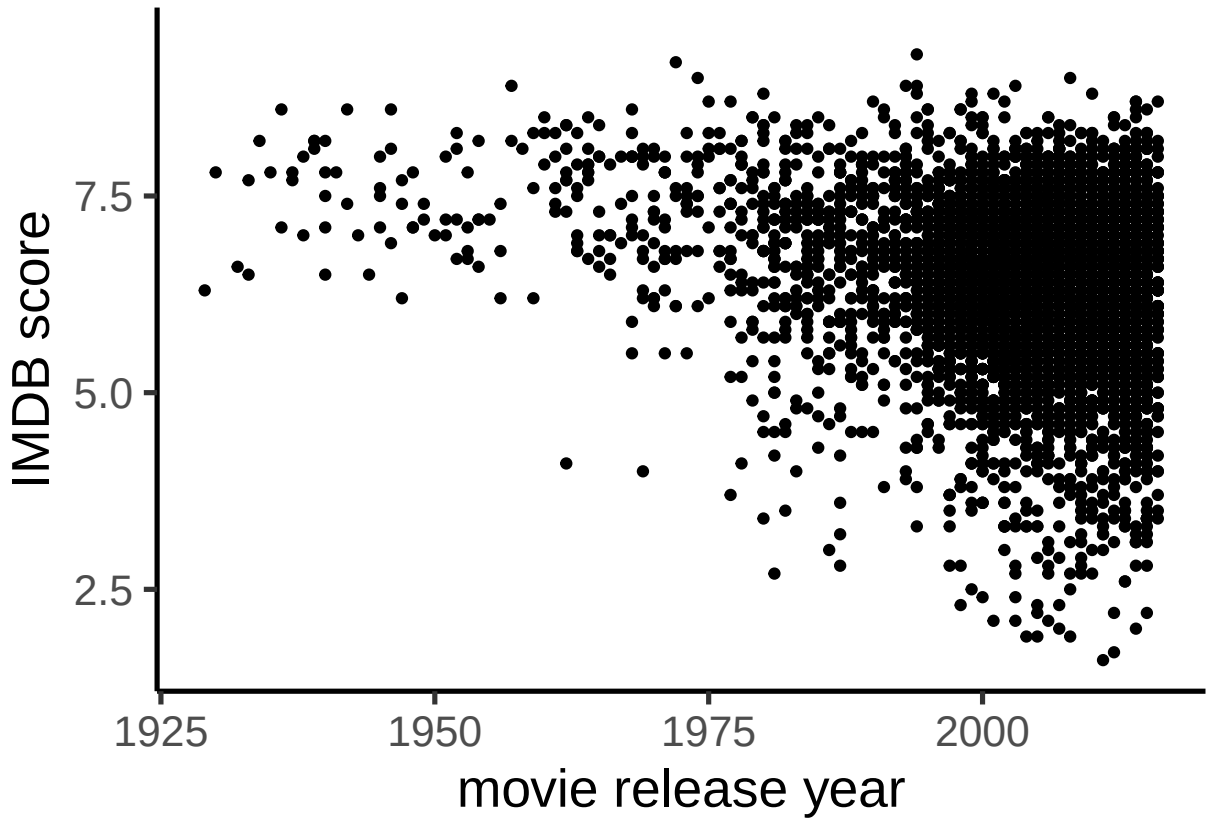
```
## Warning: Removed 97 rows containing missing values (geom_point).
```



and axis labels

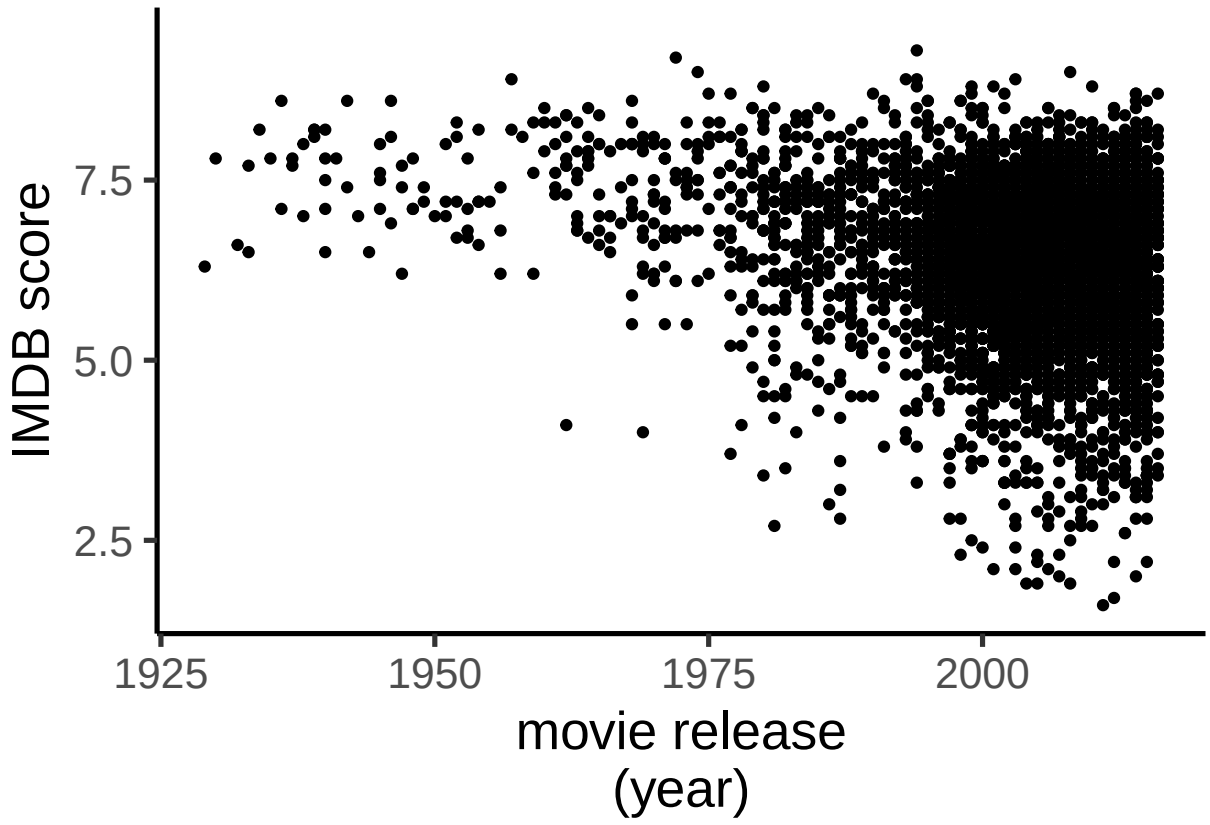
```
plot_imdb + xlab("movie release year") + ylab("IMDB score")
```

```
## Warning: Removed 97 rows containing missing values (geom_point).
```




```
plot_imdb + xlab("movie release\n(year)") + ylab("IMDB score")
```

```
## Warning: Removed 97 rows containing missing values (geom_point).
```

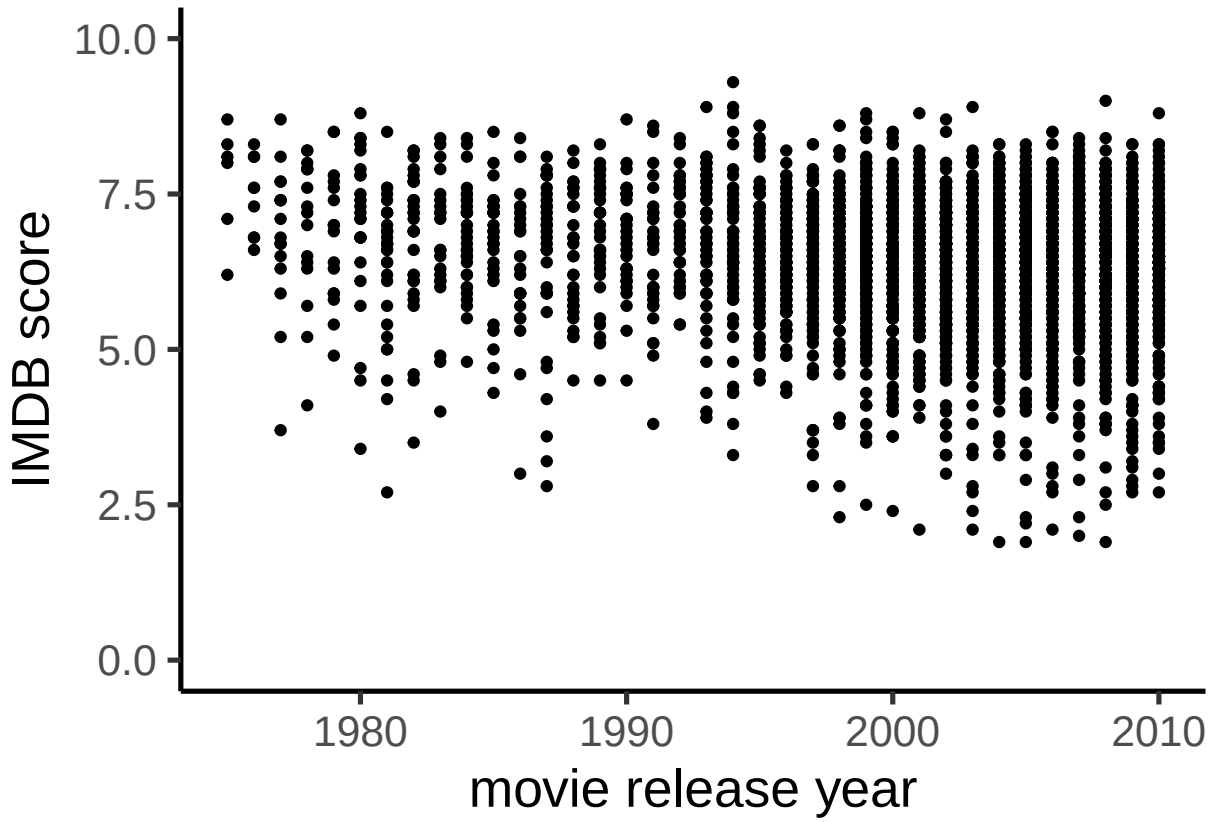



```
plot_imdb=plot_imdb + xlab("movie release year") + ylab("IMDB score")
```

ggplot scale

```
plot_imdb + scale_y_continuous(limits=c(0,10))+scale_x_continuous(limits=c(1975,2010))
```

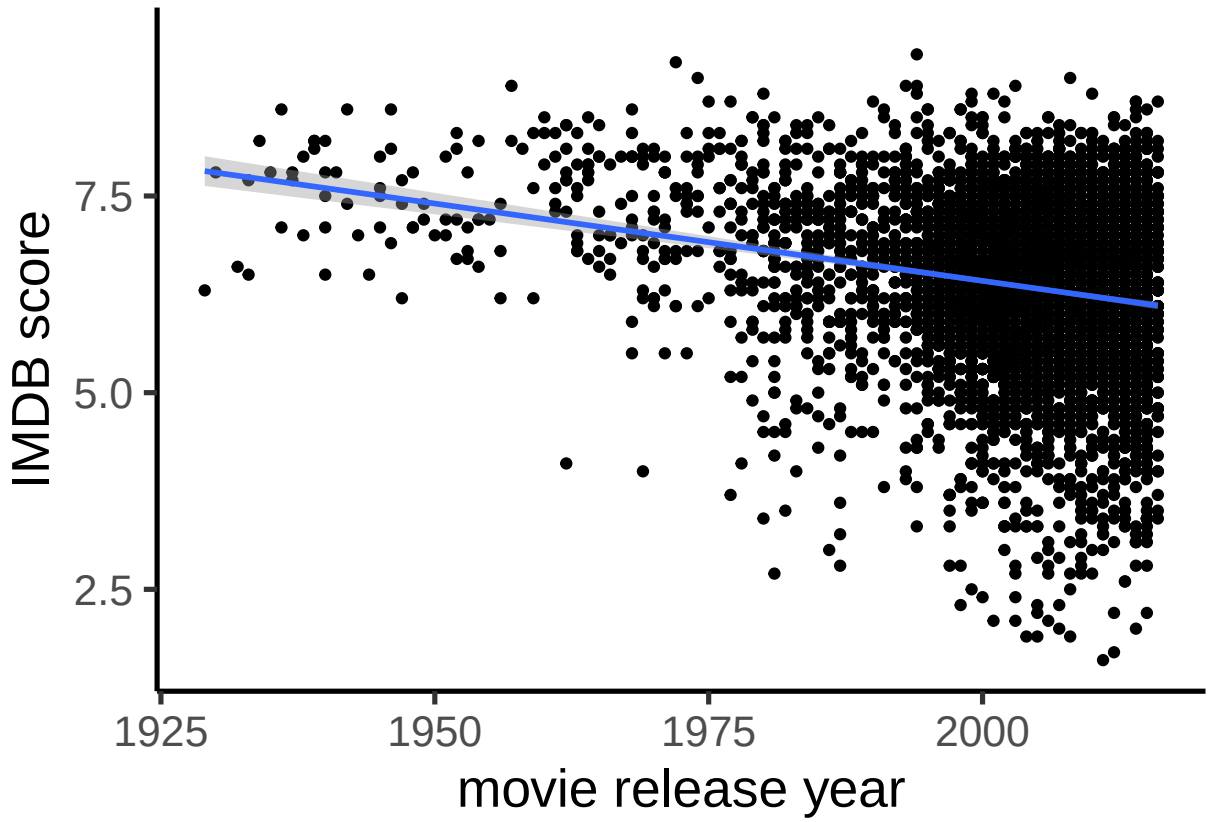
```
## Warning: Removed 1449 rows containing missing values (geom_point).
```



ggplot statistics

```
plot_imdb + stat_smooth(method="lm")
```

```
## `geom_smooth()` using formula 'y ~ x'
## Warning: Removed 97 rows containing non-finite values (stat_smooth).
## Warning: Removed 97 rows containing missing values (geom_point).
```

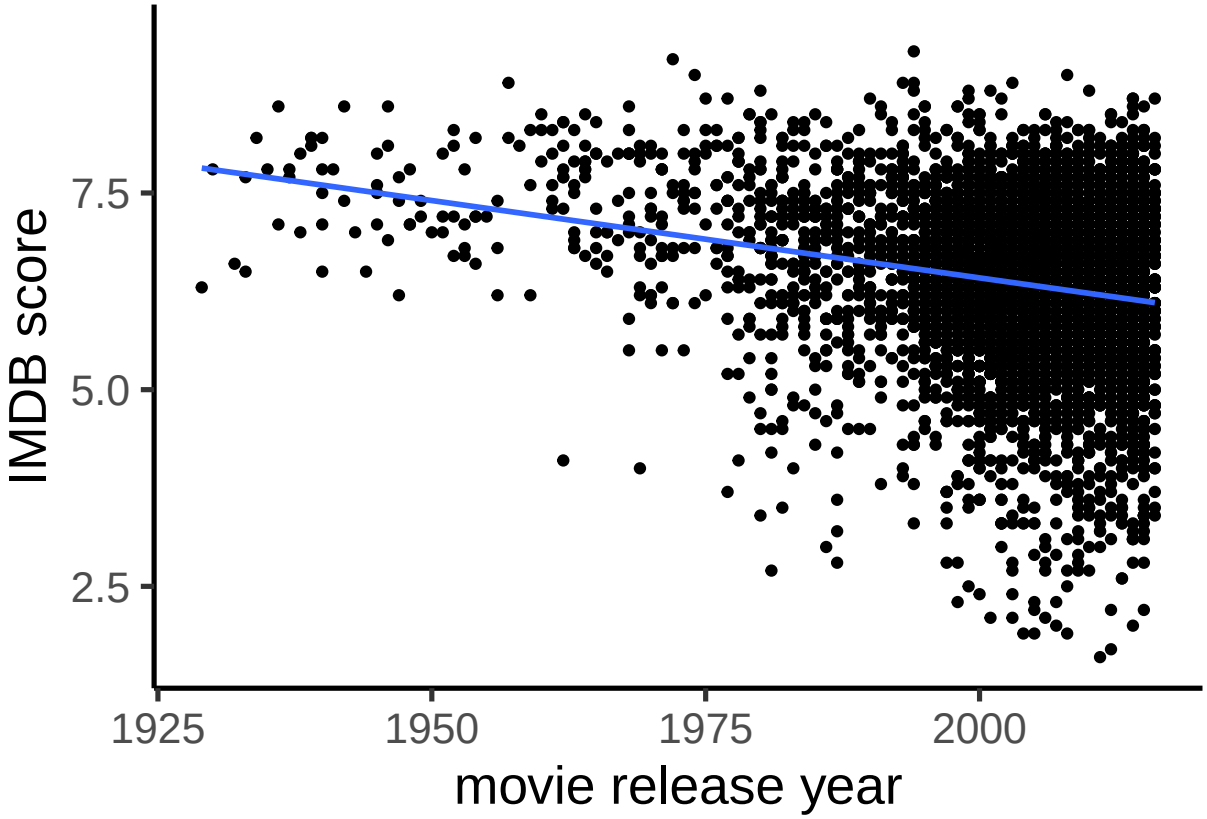



```
plot_imdb + stat_smooth(method="lm",se=F)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 97 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 97 rows containing missing values (geom_point).
```


IMDB score
7.5
5.0
2.5
1925
1950
1975
2000
movie release year