ChessGamesEDA

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
##
## 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(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.0.5
```

```
chess_games = read.csv('../Data/chess_games.csv')
head(chess_games)
```

```
##         id rated  created_at last_move_at turns victory_status winner
## 1 TZJHLljE FALSE 1.50421e+12  1.50421e+12    13      outoftime  white
## 2 l1NXvwaE  TRUE 1.50413e+12  1.50413e+12    16         resign  black
## 3 mIICvQHh  TRUE 1.50413e+12  1.50413e+12    61           mate  white
## 4 kWKvrqYL  TRUE 1.50411e+12  1.50411e+12    61           mate  white
## 5 9tXo1AUZ  TRUE 1.50403e+12  1.50403e+12    95           mate  white
## 6 MsoDV9wj FALSE 1.50424e+12  1.50424e+12     5           draw   draw
##   increment_code      white_id white_rating      black_id black_rating
## 1           15+2      bourgris         1500          a-00         1191
## 2           5+10          a-00         1322     skinnerua         1261
## 3           5+10        ischia         1496          a-00         1500
## 4           20+0 daniamurashov         1439  adivanov2009         1454
## 5           30+3     nik221107         1523  adivanov2009         1469
## 6           10+0     trelynn17         1250 franklin14532         1002
##
## 1
## 2
## 3
## 4
## 5 e4 e5 Nf3 d6 d4 Nc6 d5 Nb4 a3 Na6 Nc3 Be7 b4 Nf6 Bg5 O-O b5 Nc5 Bxf6 Bxf6 Bd3 Qd7 O-O Nxd3 Qxd3 c6
## 6
```

```
##   opening_eco                            opening_name opening_ply
## 1         D10       Slav Defense: Exchange Variation           5
## 2         B00 Nimzowitsch Defense: Kennedy Variation           4
## 3         C20  King's Pawn Game: Leonardis Variation           3
## 4         D02 Queen's Pawn Game: Zukertort Variation           3
## 5         C41                      Philidor Defense           5
## 6         B27   Sicilian Defense: Mongoose Variation           4
```

Typical Chess Rating Categories

Rating:Category

2500+ : Grandmaster 2200-2499 : Master 2000-2199 : Expert 1800-1999 : Class A 1600:1799 : Class B 1400-1599 : Class C 1200:1399 : Class D 900:1200 : Novice

Lichess ratings are about 300 rating points above (starting rating is 1200 for FIDE and Chess.com, 1500 on LiChess)

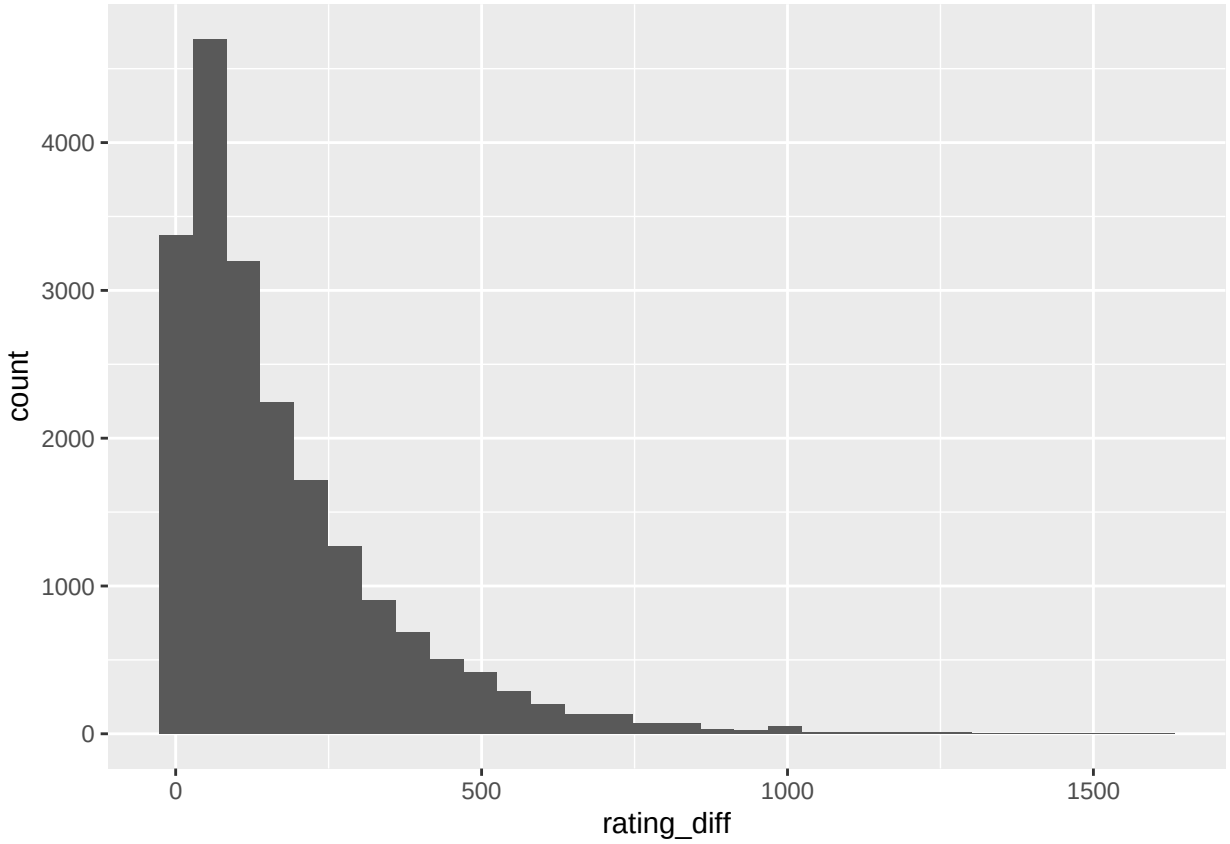
```
chess_games = chess_games %>%
  mutate(white_rating_adj = white_rating - 300) %>%
  mutate(black_rating_adj = black_rating - 300) %>%
  mutate(rating_diff = abs(white_rating - black_rating)) %>%
  mutate(avg_rating = (white_rating_adj + black_rating_adj) / 2) %>%
  mutate(rating_category = ifelse(avg_rating >= 2500, 'Grandmaster',
                           ifelse(avg_rating >= 2200 & avg_rating < 2500, 'Master',
                             ifelse(avg_rating >= 2000 & avg_rating < 2200, 'Expert',
                               ifelse(avg_rating >= 1800 & avg_rating < 2000, 'ClassA',
                                 ifelse(avg_rating >= 1600 & avg_rating < 1800, 'ClassB',
                                   ifelse(avg_rating >= 1400 & avg_rating < 1600, 'ClassC',
                                     ifelse(avg_rating >= 1200 & avg_rating < 1400, 'ClassD',
                                       ifelse(avg_rating >= 900 & avg_rating < 1200, 'Novice',
                                              'Rookie')
  ))))))))
```

```
chess_games %>%
  group_by(rating_category) %>%
  summarise(count = n(),
            avg_turns = mean(turns)) %>%
  arrange(-avg_turns)
```

```
## # A tibble: 7 x 3
##   rating_category count avg_turns
##   <chr>           <int>     <dbl>
## 1 Expert             83      73.5
## 2 ClassA            634      71.3
## 3 ClassB           1979      66.6
## 4 ClassC           3915      65.5
## 5 ClassD           5721      61.8
## 6 Novice           6557      55.1
## 7 Rookie           1169      50.0
```

```
ggplot(chess_games, aes(rating_diff)) +
  geom_histogram()
```

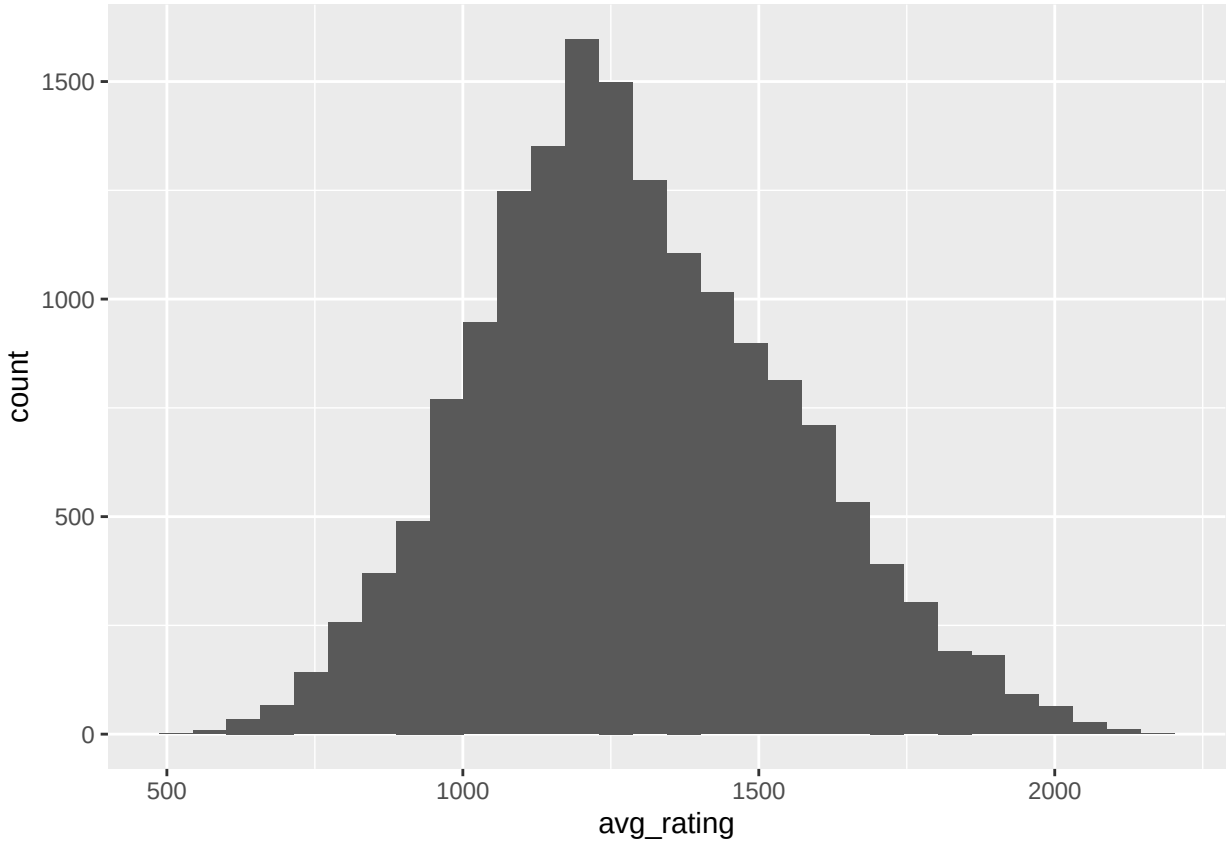
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
chess_games_filt = chess_games %>%
  filter(rating_diff < 300)

ggplot(chess_games_filt, aes(avg_rating)) +
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


1500
1000
500
0
count
500
1000
1500
2000
avg_rating