Trailer recommendations exploration

Gaurav Modi

January 25, 2018

```
## Conflicts with tidy packages ----------------------------------------------
```

This dataset was created by Eva Zhong, based on a movielens experiment led by Tahir Sousa.

The dataset contains one observation per user-login. A row appears for any user who viewed one or more movie trailers during their session. The Algorithm represents a recommendation algorithm they were given, which was randomly assigned on each login.

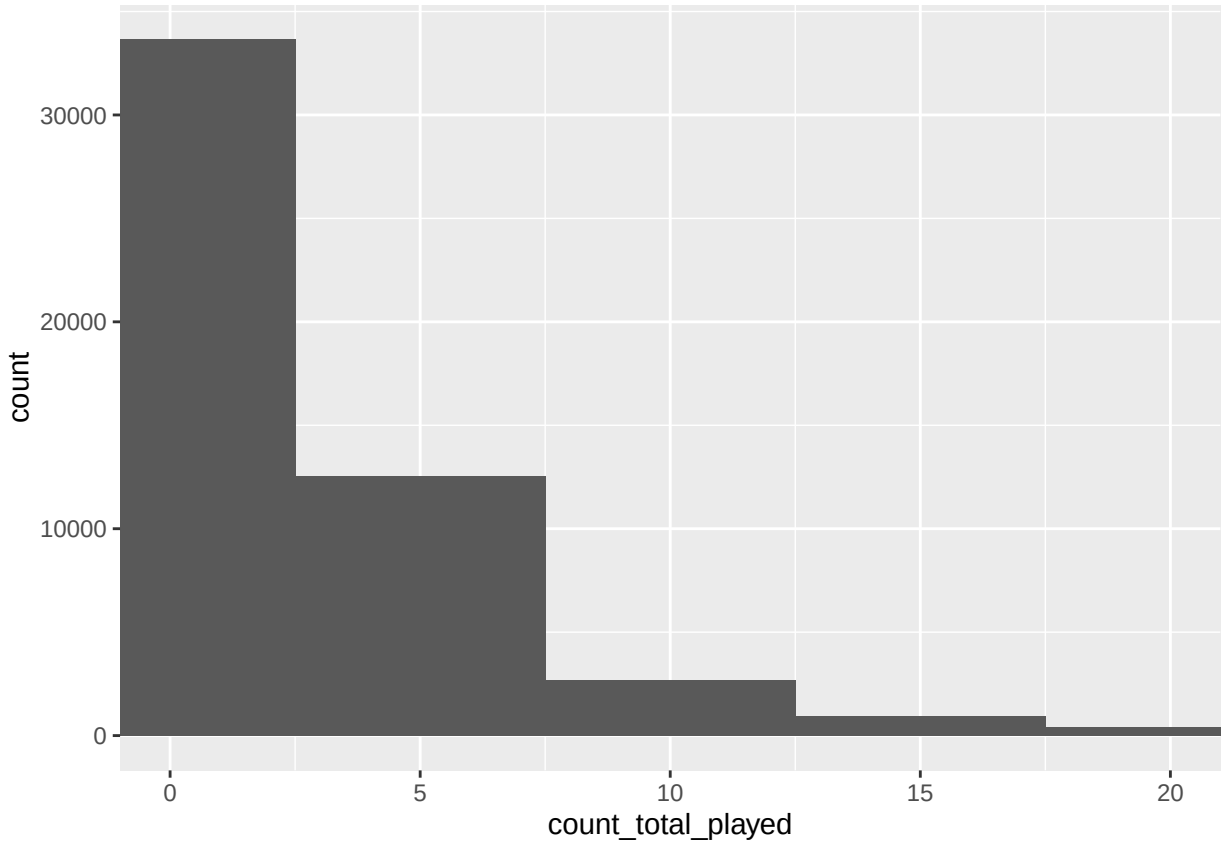
- `count_total_played` – total number of movie trailers started during the login session
- `count_rec_played` – total number of movie trailers played *from a list of recommendations* started during the login session

```
## # A tibble: 50,881 × 5
##    userId loginId        Algorithm count_total_played count_rec_played
##     <chr>   <chr>            <ord>               <int>            <dbl>
## 1    1892 Nm0bajY    TagSimilarity                   1                0
## 2   12337 OnHpLST  PredictedRating                   1                0
## 3   16783 2bqCVrC    TagSimilarity                  19                0
## 4   22005 M1nww5f ShuffledTopPicks                   1                0
## 5   26229 t7hZ2Ty  PredictedRating                   1                0
## 6   36452 XpQ2umM  FilmReleaseDate                   8                0
## 7   41965 1gT5zeZ  PredictedRating                   1                0
## 8   41965 1xaQ8Fy    TagSimilarity                   2                0
## 9   41965 95ylvvy  PredictedRating                   1                0
## 10  41965 Fca2XFK ShuffledTopPicks                   5                0
## # ... with 50,871 more rows
```

Histogram of `count_total_played`

```
g <- ggplot(data = trailer_activities)
```

```
p1 <- g + geom_histogram(mapping = aes(x = count_total_played), binwidth = 5) +
  coord_cartesian(xlim = c(0, 20))

p1
```

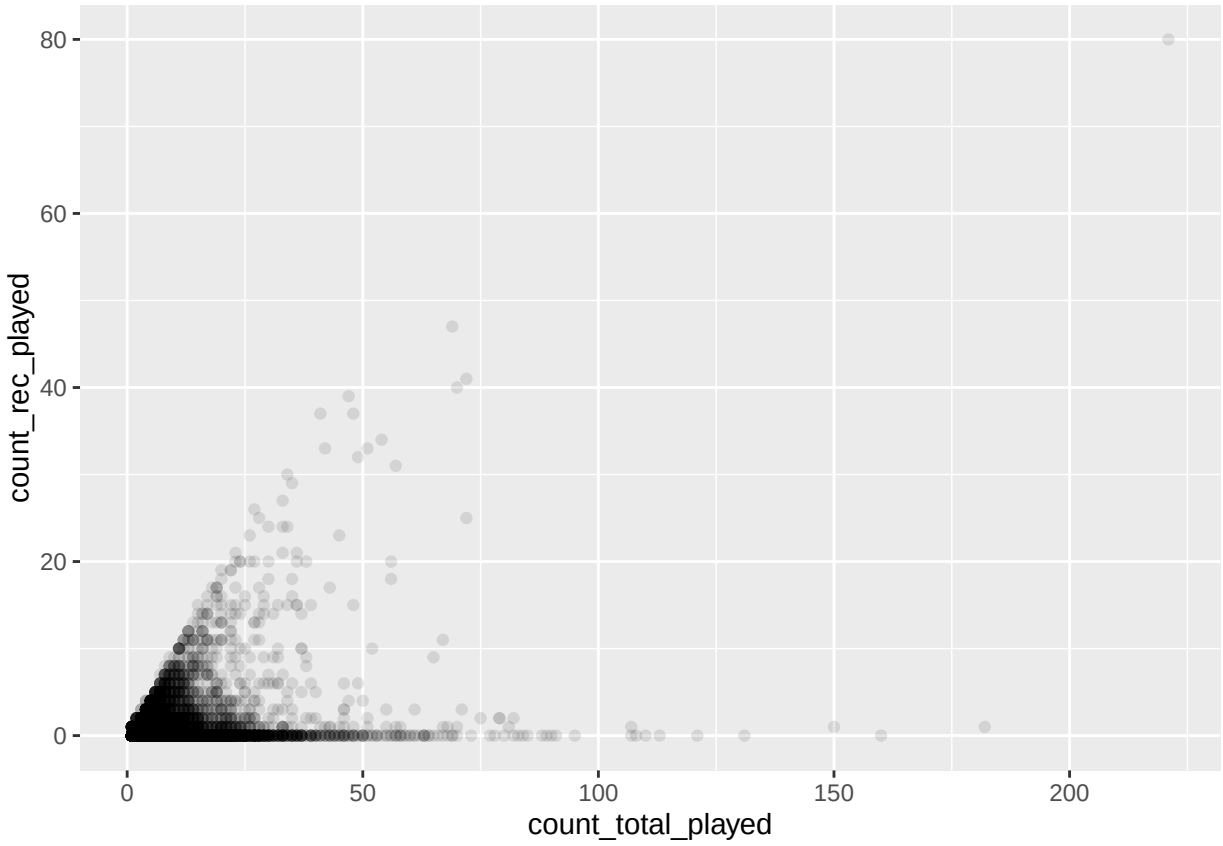


Scatter plots

The relationship between `count_total_played` and `count_rec_played` using:

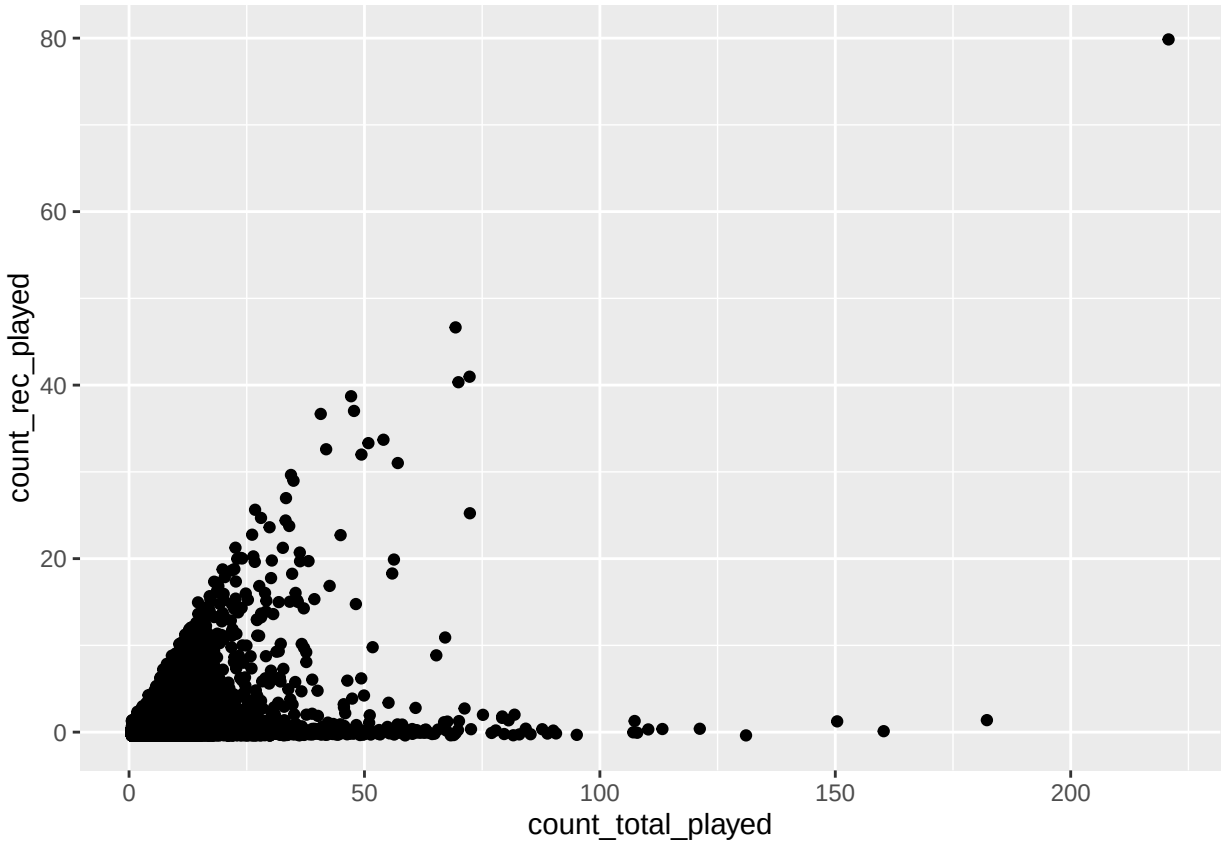
- geom_point (setting `alpha=0.1`)

```
p2 <- g + geom_point(mapping = aes(x = count_total_played, y = count_rec_played), alpha = 0.1)

p2
```



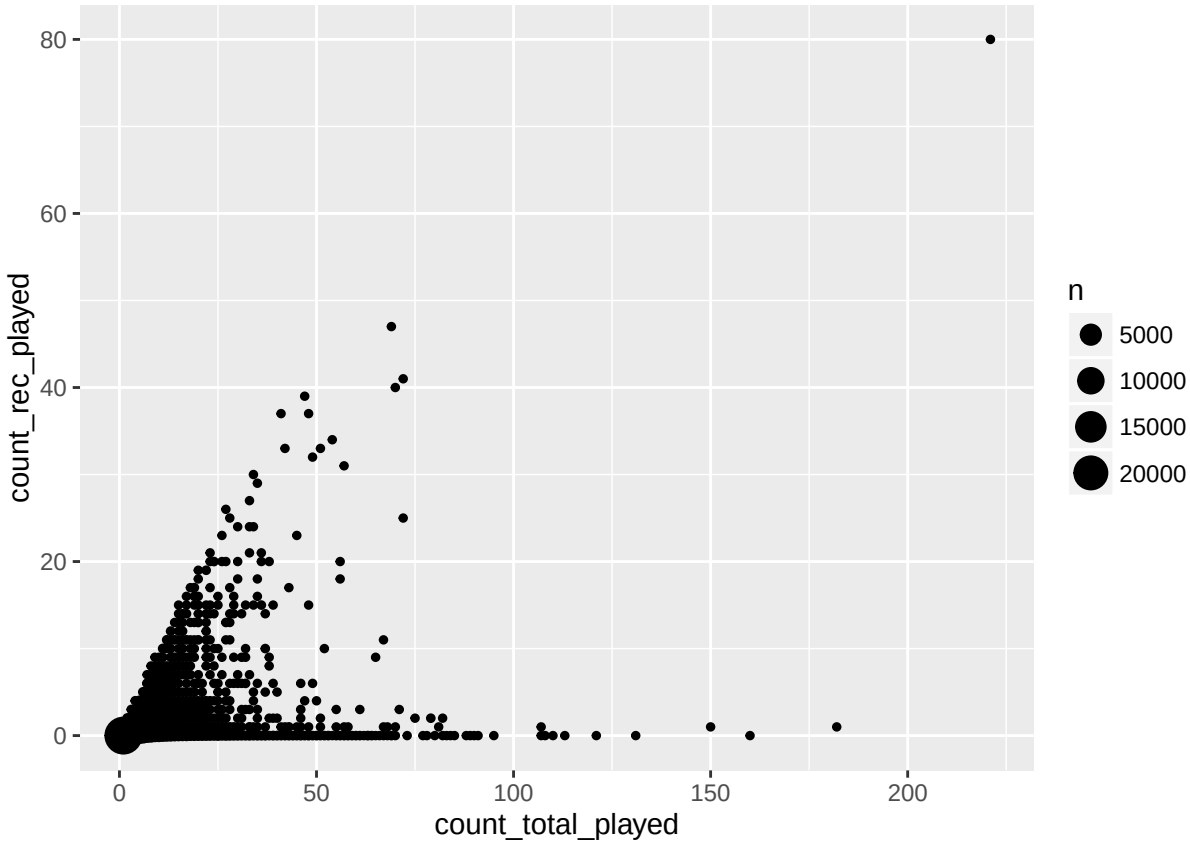
- geom_jitter

```
p3 <- g + geom_jitter(mapping = aes(x = count_total_played, y = count_rec_played))

p3
```



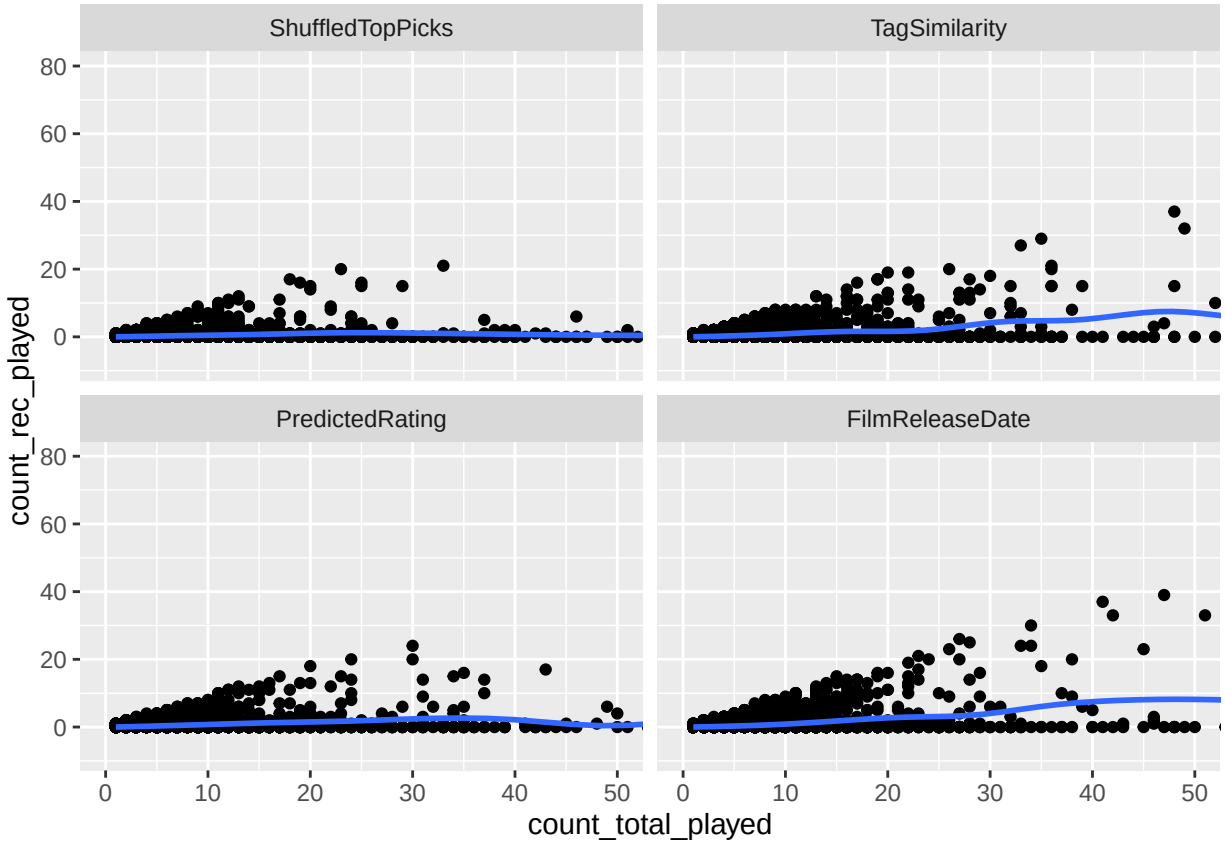
- geom_count

```
p4 <- g + geom_count(mapping = aes(x = count_total_played, y = count_rec_played))

p4
```



- a `facet_wrap` to split the chart by algorithm

```
p5 <- g + geom_point(mapping = aes(x = count_total_played, y = count_rec_played)) +
  geom_smooth(mapping = aes(x = count_total_played, y = count_rec_played)) +
  coord_cartesian(xlim = c(0, 50)) +
  facet_wrap( ~ Algorithm)

p5
```



Summary Statistics

Here are aggregate stats of recommended play actions, grouped by algorithm:

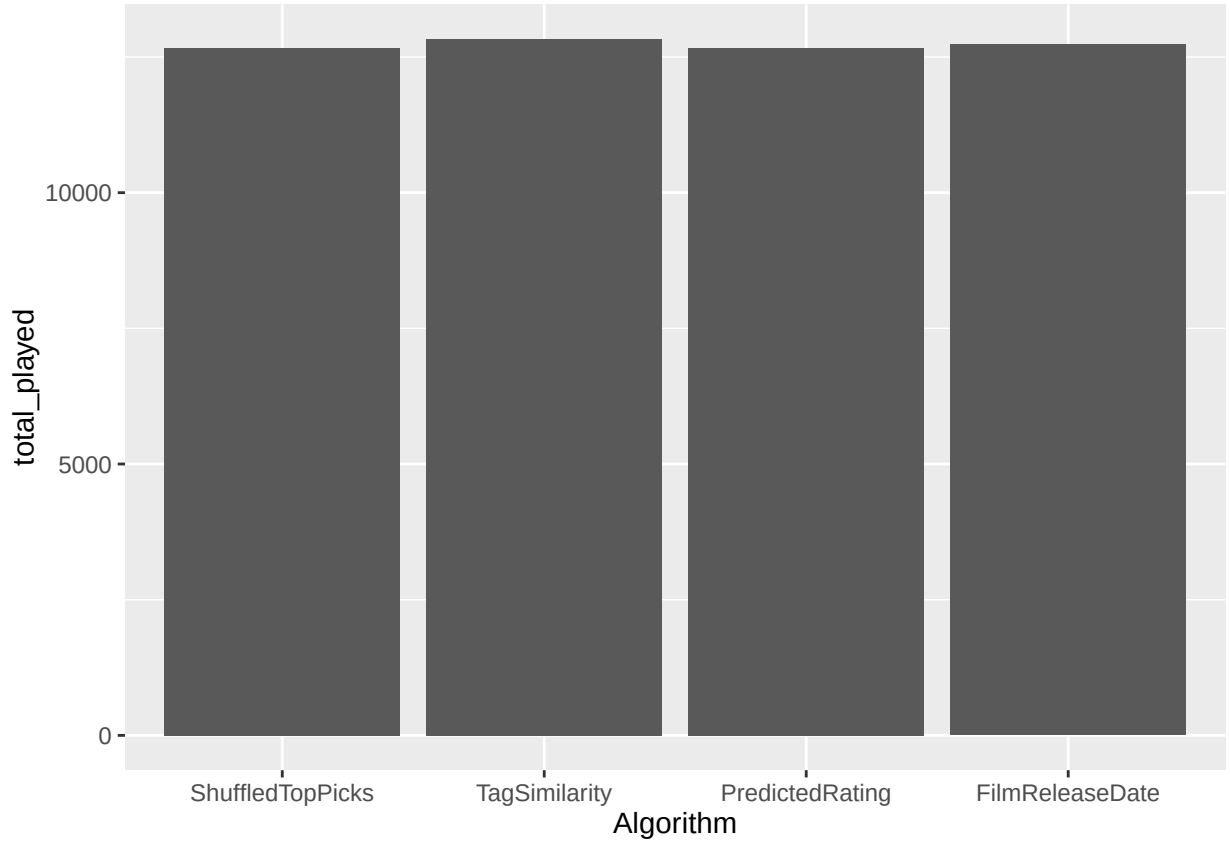
```
trailer_activities_summary <- trailer_activities %>%
  group_by(Algorithm) %>%
  summarise(
    total_played = n(),
    mean_rec_played = mean(count_rec_played),
    max_rec_played = max(count_rec_played),
    min_rec_played = min(count_rec_played),
    std_err_rec_played = sd(count_rec_played)/sqrt(n())
  )
trailer_activities_summary
```

```
## # A tibble: 4 × 6
##          Algorithm total_played mean_rec_played max_rec_played
##              <ord>        <int>           <dbl>          <dbl>
## 1 ShuffledTopPicks        12659       0.1094873             21
## 2    TagSimilarity        12828       0.2172591             47
## 3  PredictedRating        12664       0.1762476             80
## 4  FilmReleaseDate        12730       0.2203456             40
## # ... with 2 more variables: min_rec_played <dbl>,
## #   std_err_rec_played <dbl>
```

Bar chart!

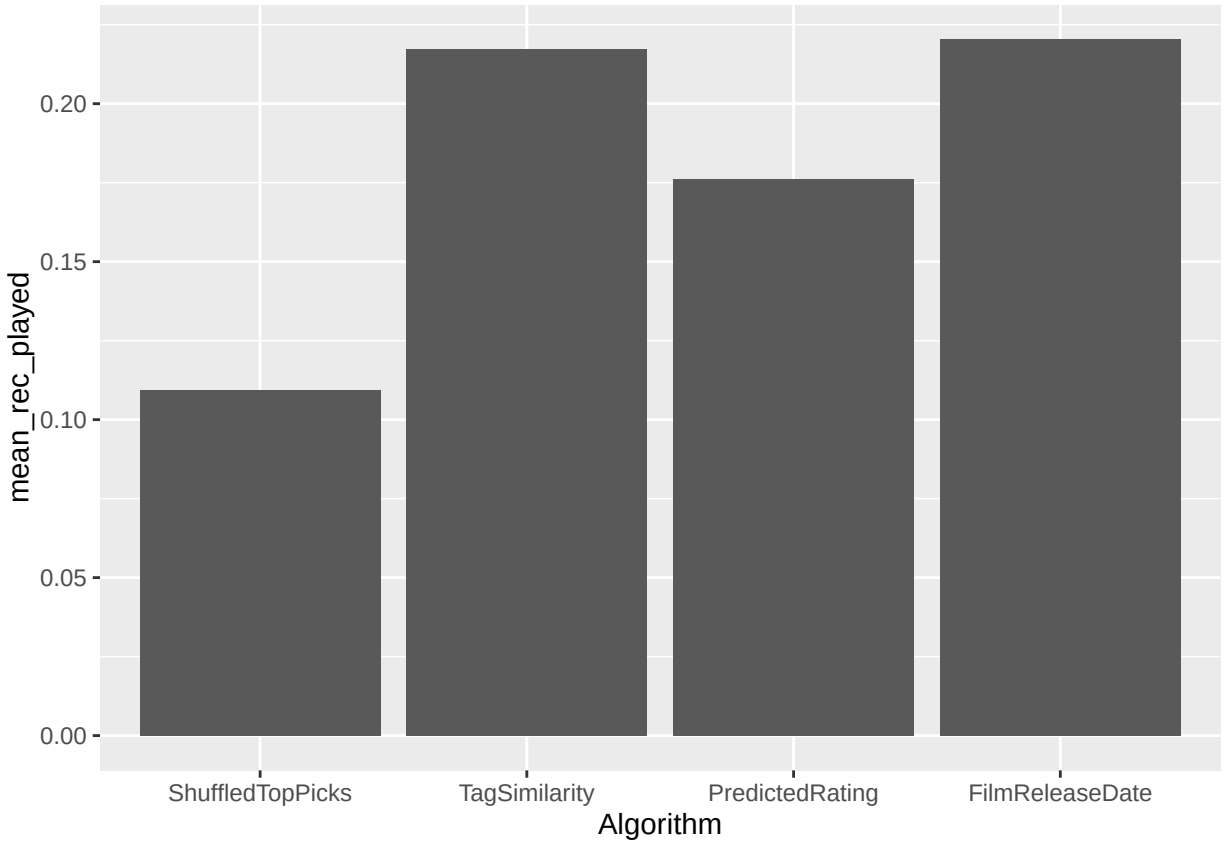
Are the number of observations balanced between the four experimental algorithms? Build a bar chart of this tibble by Algorithm.

```
g1 <- ggplot(data = trailer_activities_summary)

p6 <- g1 + geom_bar(mapping = aes(x = Algorithm, y = total_played), stat = "identity")

p6
```



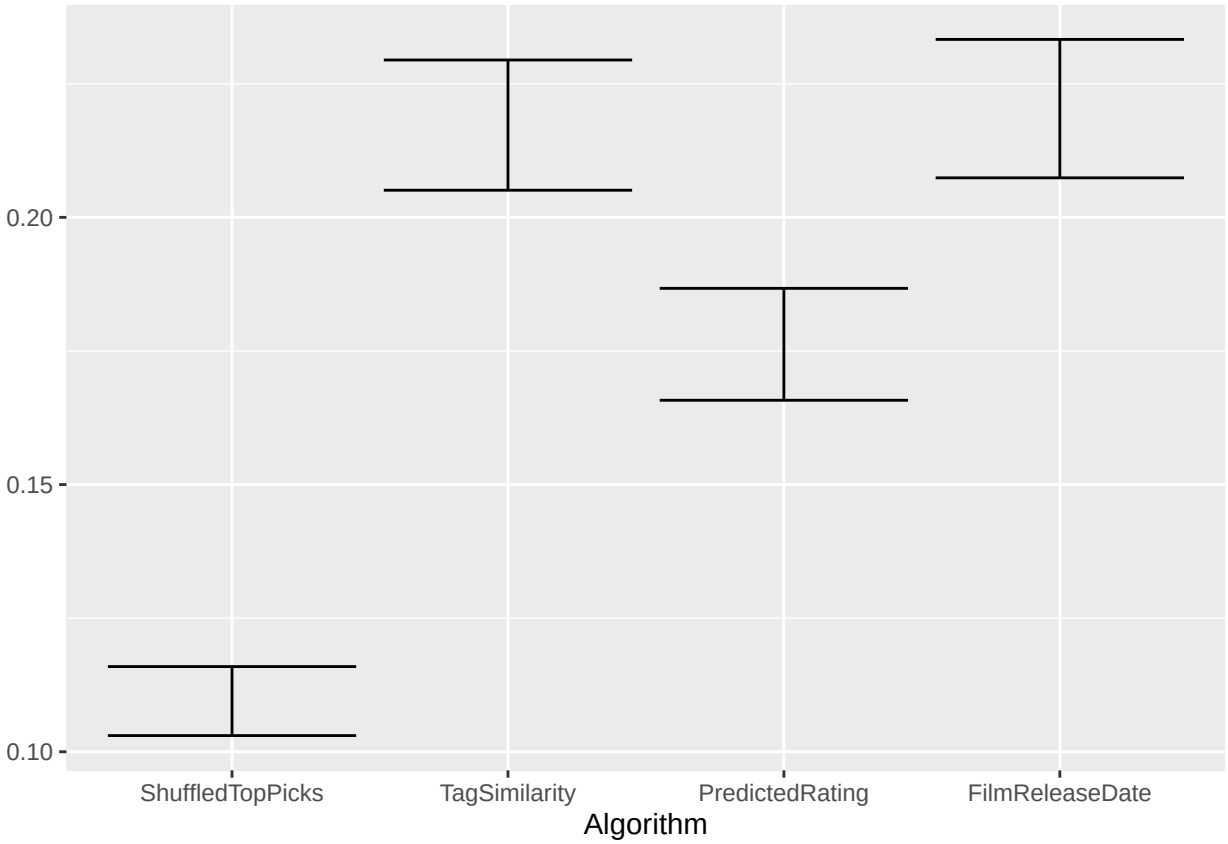
- Charting Algorithm by mean_rec_played

```
p7 <- g1 + geom_bar(mapping = aes(x = Algorithm, y = mean_rec_played), stat = "identity")

p7
```



- Standard error plot for each algorithm

```
p8 <- g1 + geom_errorbar(aes(x = Algorithm, ymin = mean_rec_played - std_err_rec_played, ymax = mean_rec

p8
```


0.20
0.15
0.10
ShuffledTopPicks
TagSimilarity
PredictedRating
FilmReleaseDate
Algorithm