Spotify Data Bayesian Analysis

Nathaniel Maxwell, Jessie Bierschenk

```
Not_skipped_tracks <- data.frame(read_csv("data/tf_0_reduced.csv"))
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   track_id = col_character(),
##   release_year = col_double(),
##   duration = col_double(),
##   us_popularity_estimate = col_double(),
##   acousticness = col_double(),
##   beat_strength = col_double(),
##   bounciness = col_double(),
##   danceability = col_double(),
##   energy = col_double(),
##   instrumental = col_double(),
##   liveness = col_double(),
##   loudness = col_double(),
##   mode = col_character(),
##   speechiness = col_double(),
##   tempo = col_double(),
##   valence = col_double(),
##   skipped = col_double()
## )
```

```
Skipped_tracks <- data.frame(read_csv("data/tf_1_reduced.csv"))
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   track_id = col_character(),
##   release_year = col_double(),
##   duration = col_double(),
##   us_popularity_estimate = col_double(),
##   acousticness = col_double(),
##   beat_strength = col_double(),
##   bounciness = col_double(),
##   danceability = col_double(),
##   energy = col_double(),
##   instrumental = col_double(),
##   liveness = col_double(),
##   loudness = col_double(),
##   mode = col_character(),
##   speechiness = col_double(),
```

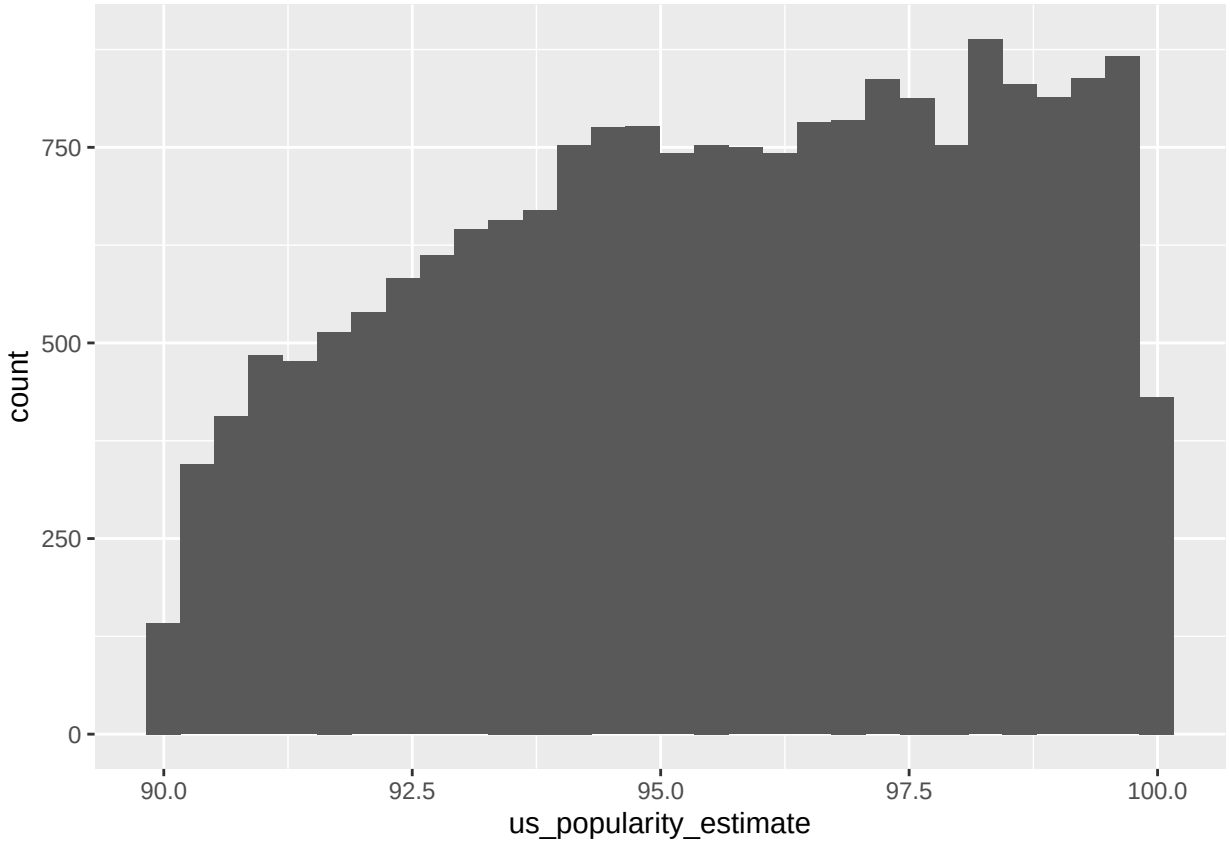
```
##   tempo = col_double(),
##   valence = col_double()
## )
```

```
All_tracks <- data.frame(read_csv("data/tf_reduced.csv"))
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   track_id = col_character(),
##   release_year = col_double(),
##   duration = col_double(),
##   us_popularity_estimate = col_double(),
##   acousticness = col_double(),
##   beat_strength = col_double(),
##   bounciness = col_double(),
##   danceability = col_double(),
##   energy = col_double(),
##   instrumental = col_double(),
##   liveness = col_double(),
##   loudness = col_double(),
##   mode = col_character(),
##   speechiness = col_double(),
##   tempo = col_double(),
##   valence = col_double(),
##   skipped = col_double()
## )
```

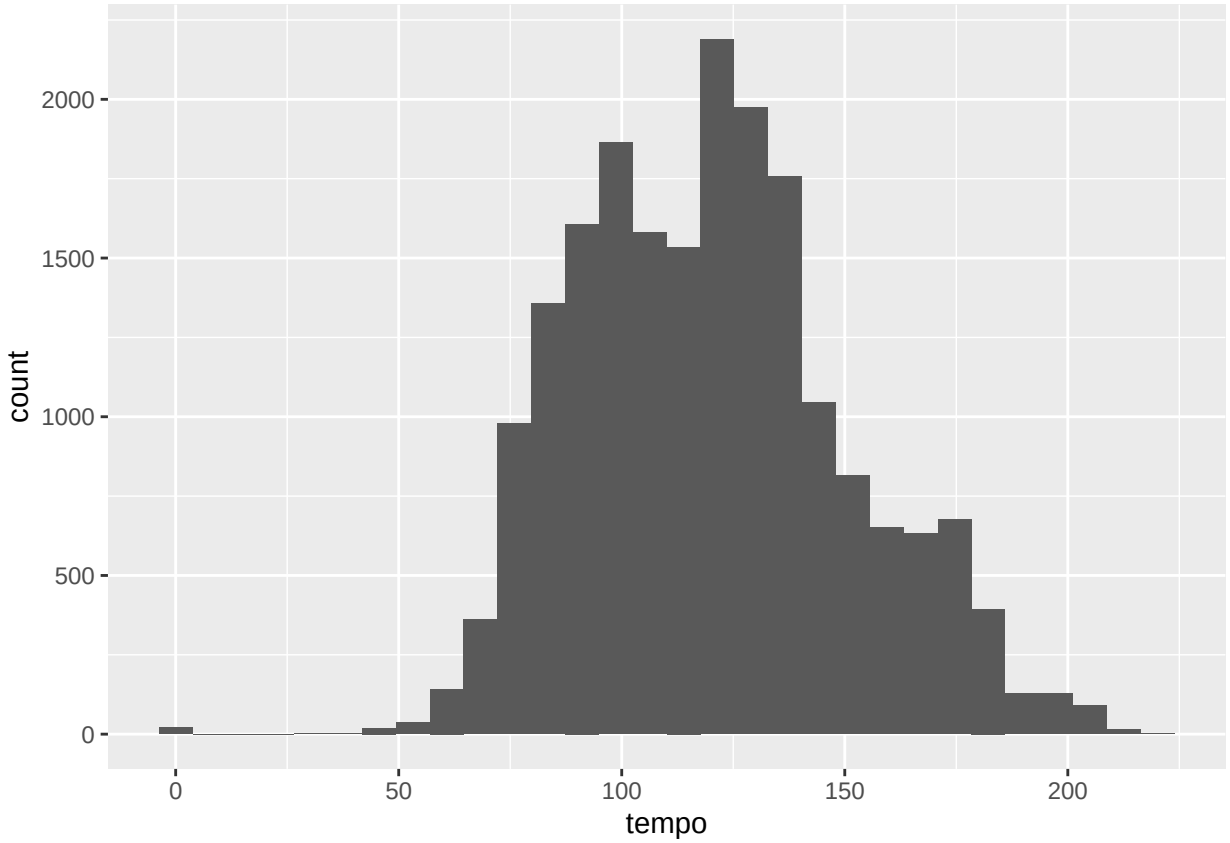
```
ggplot(data = All_tracks, aes(x = us_popularity_estimate)) +
  geom_histogram()
```

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



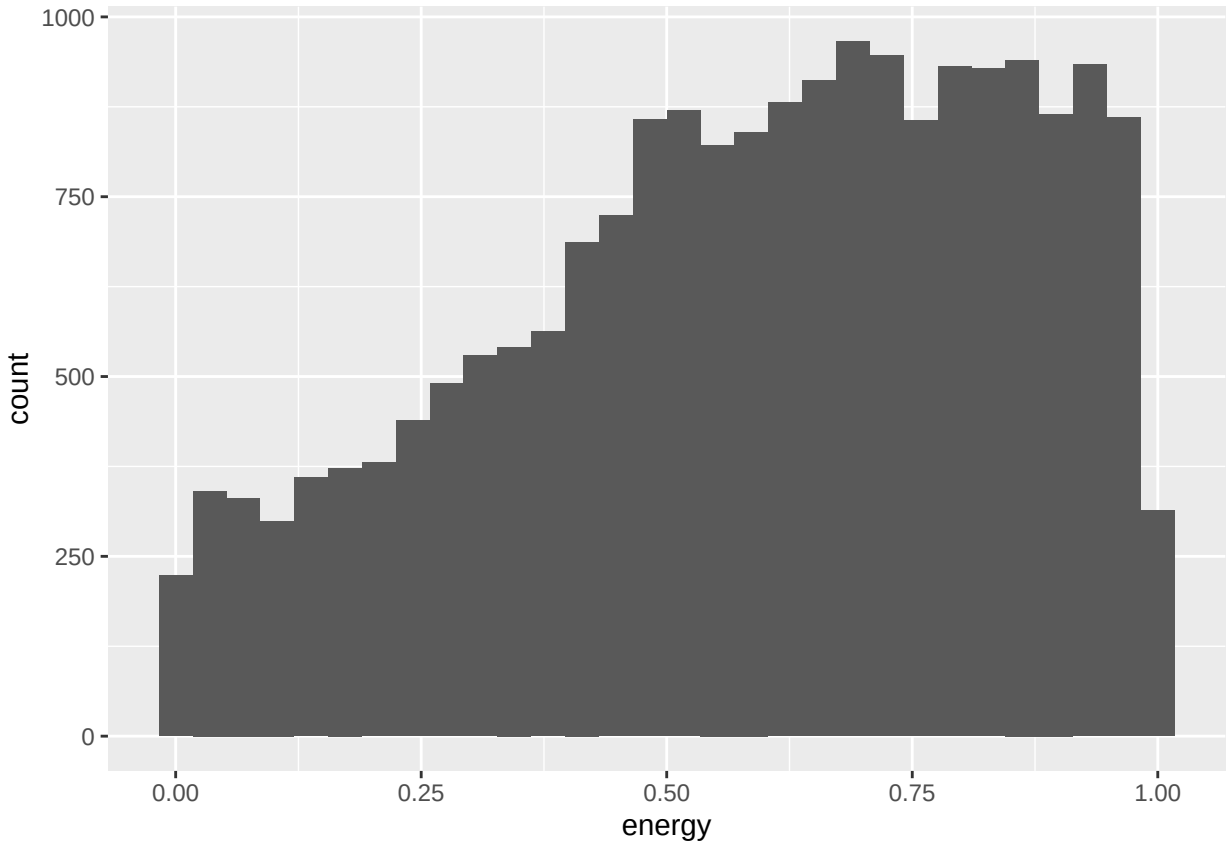

```
ggplot(data = All_tracks, aes(x = tempo)) +
  geom_histogram()
```

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



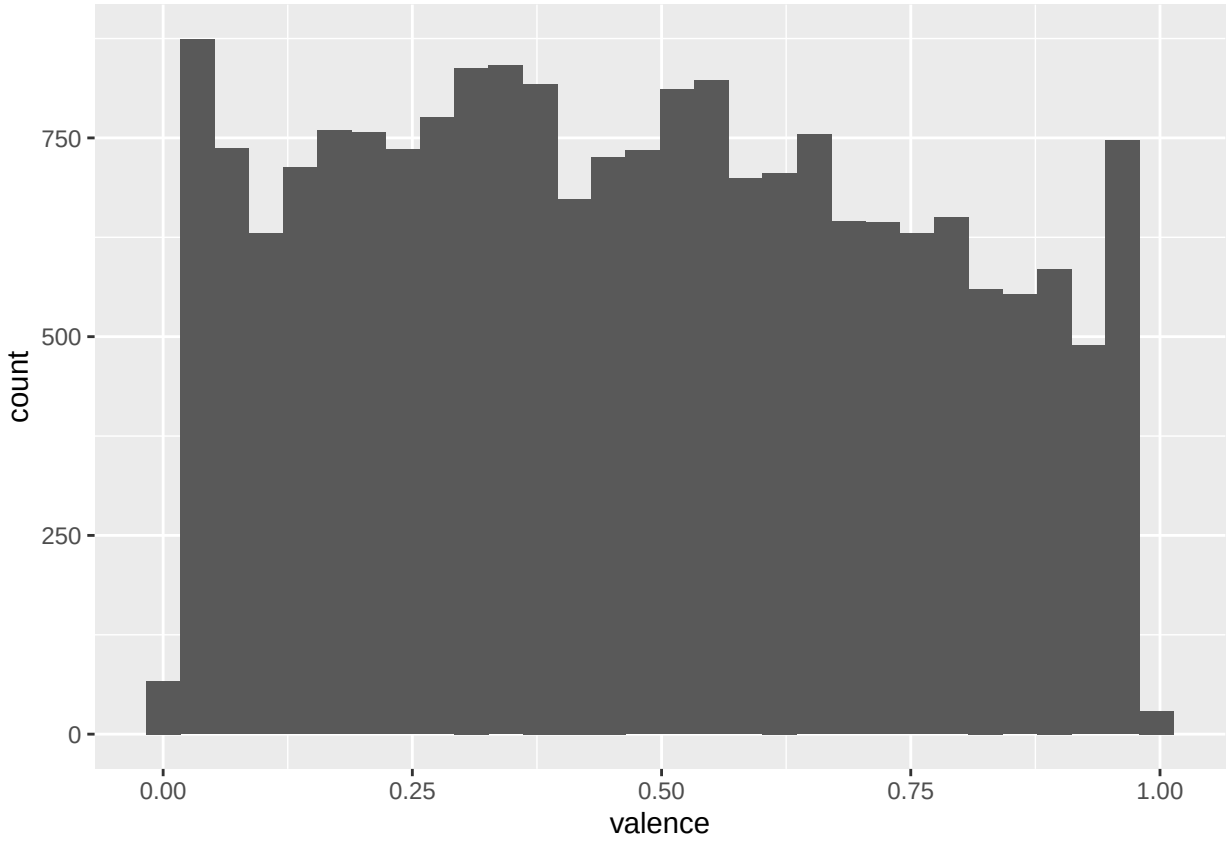

```
ggplot(data = All_tracks, aes(x = energy)) +
  geom_histogram()
```

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



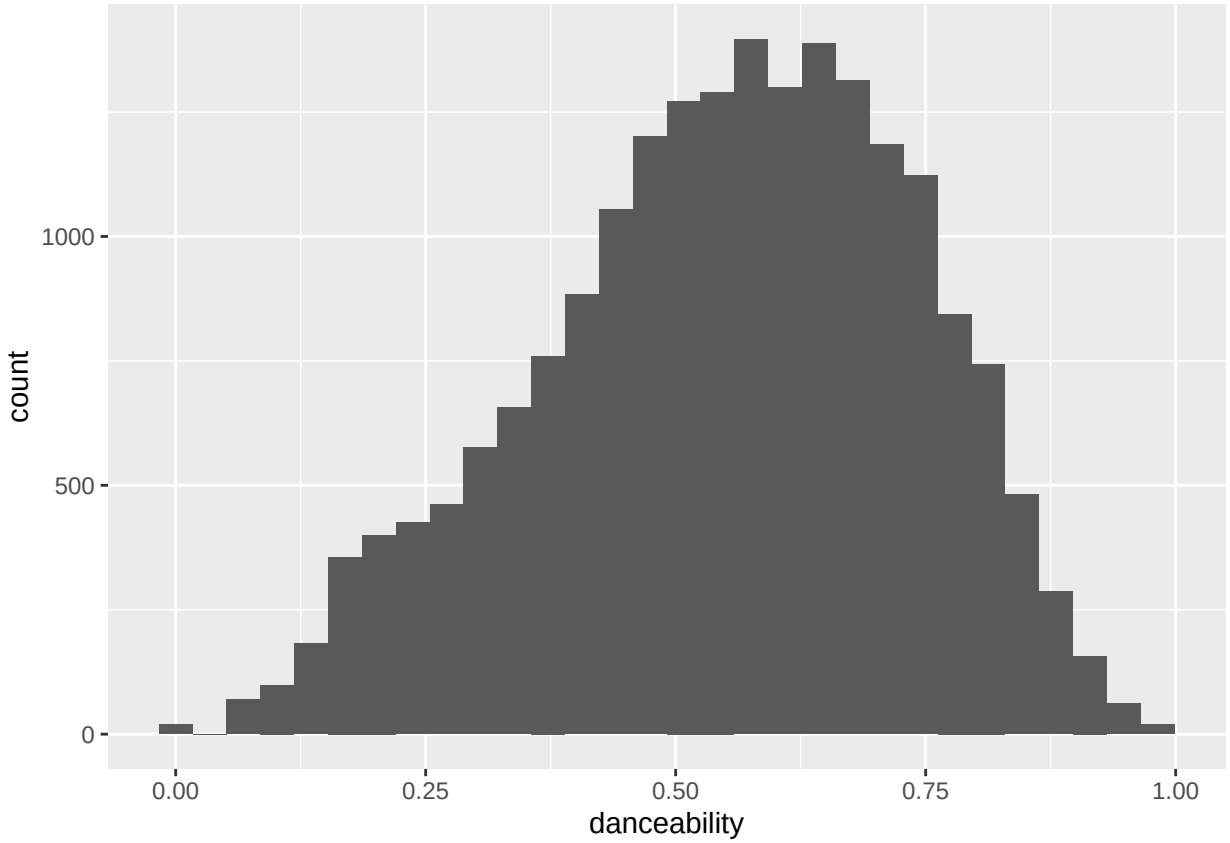

```
ggplot(data = All_tracks, aes(x = valence)) +
  geom_histogram()
```

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



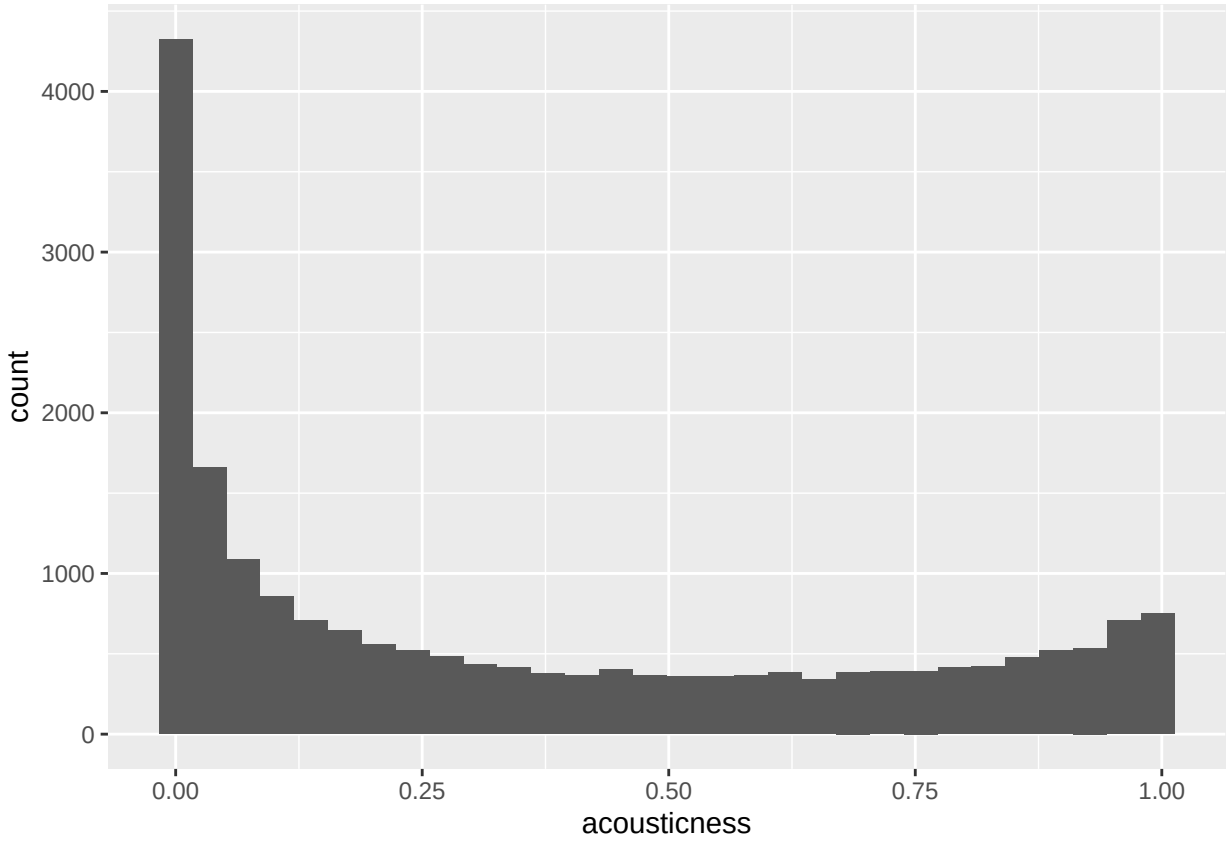

```
ggplot(data = All_tracks, aes(x = danceability)) +
  geom_histogram()
```

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



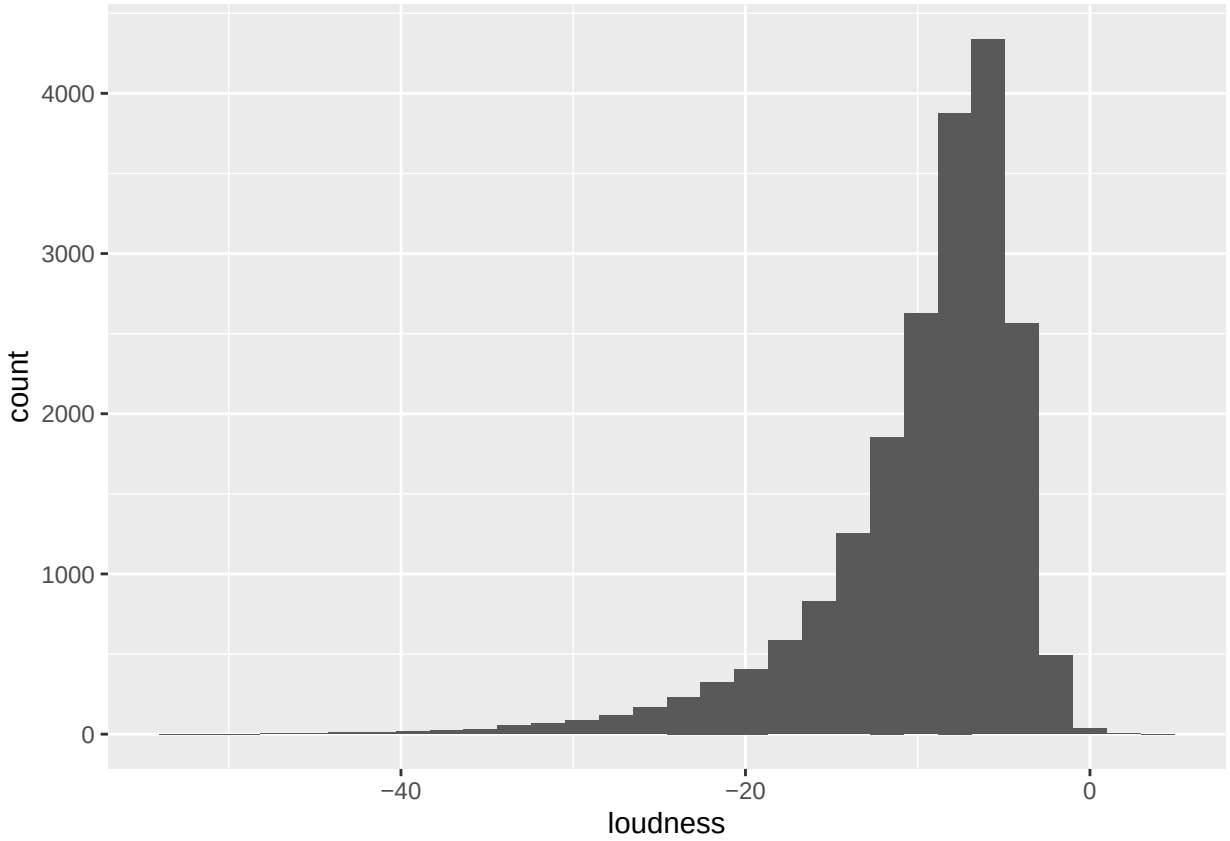

```
ggplot(data = All_tracks, aes(x = acousticness)) +
  geom_histogram()
```

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



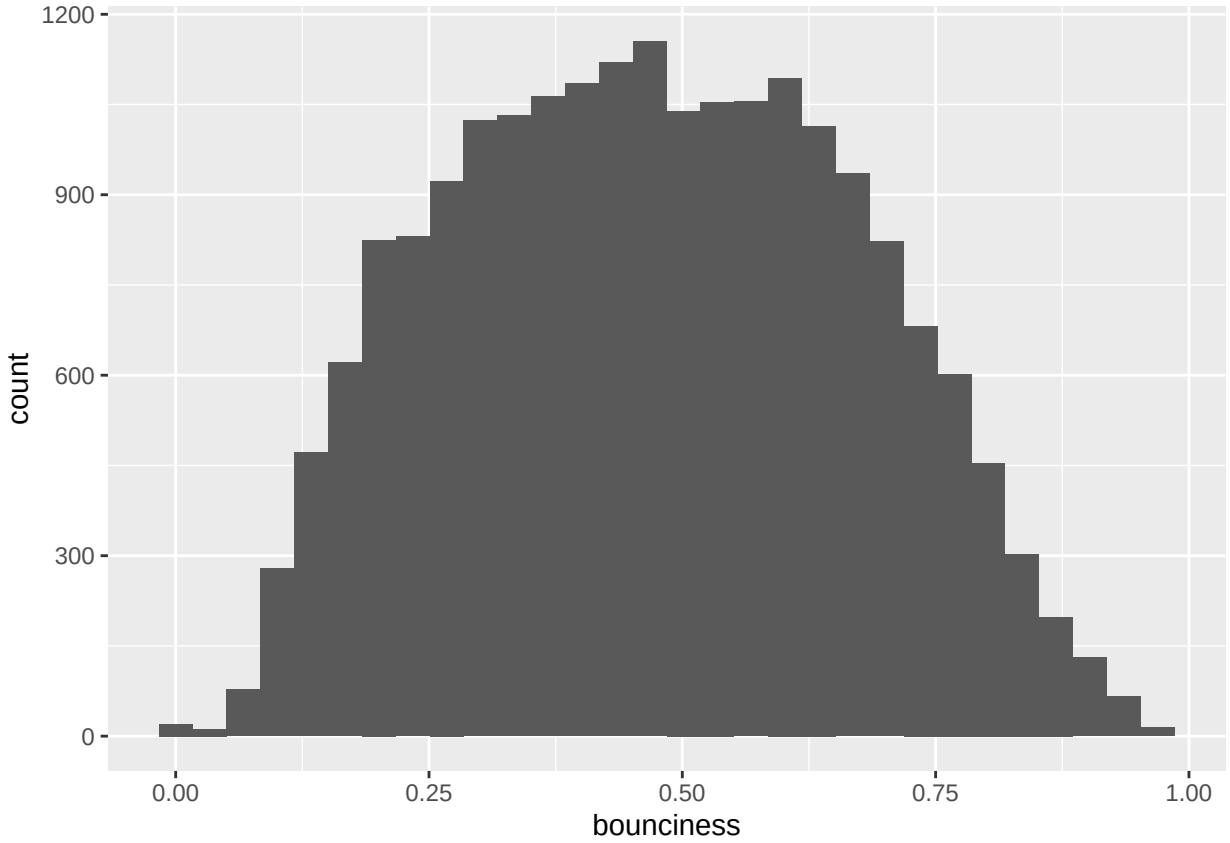

```
ggplot(data = All_tracks, aes(x = loudness)) +
  geom_histogram()
```

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



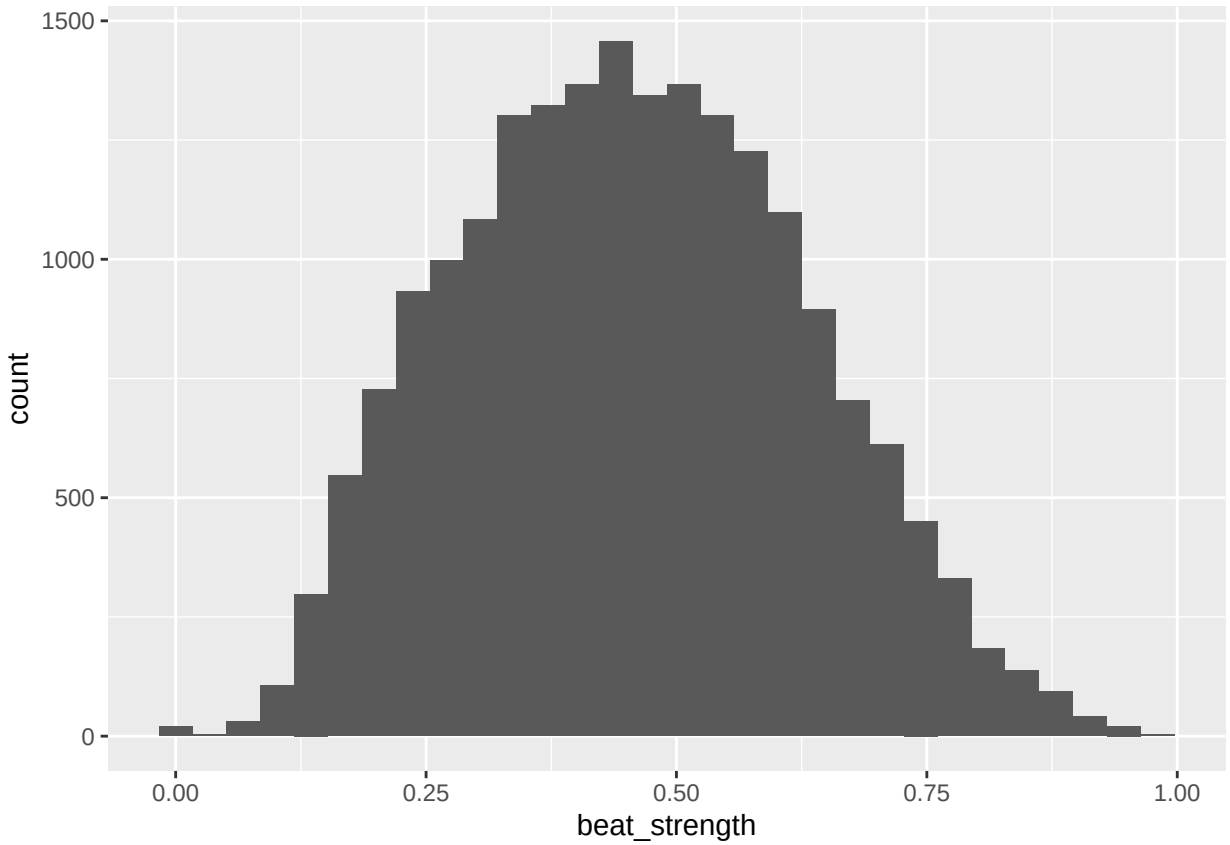

```
ggplot(data = All_tracks, aes(x = bounciness)) +
  geom_histogram()
```

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



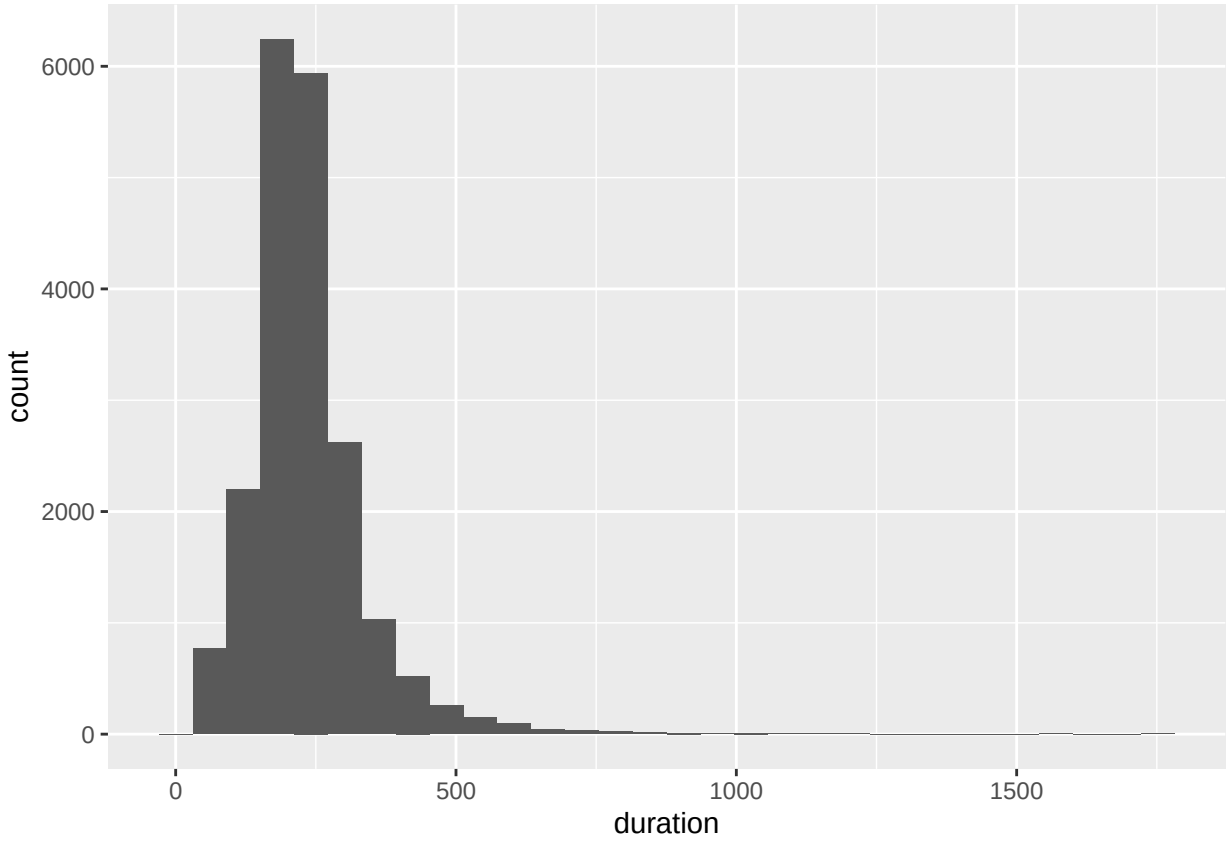

```
ggplot(data = All_tracks, aes(x = beat_strength)) +
  geom_histogram()
```

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



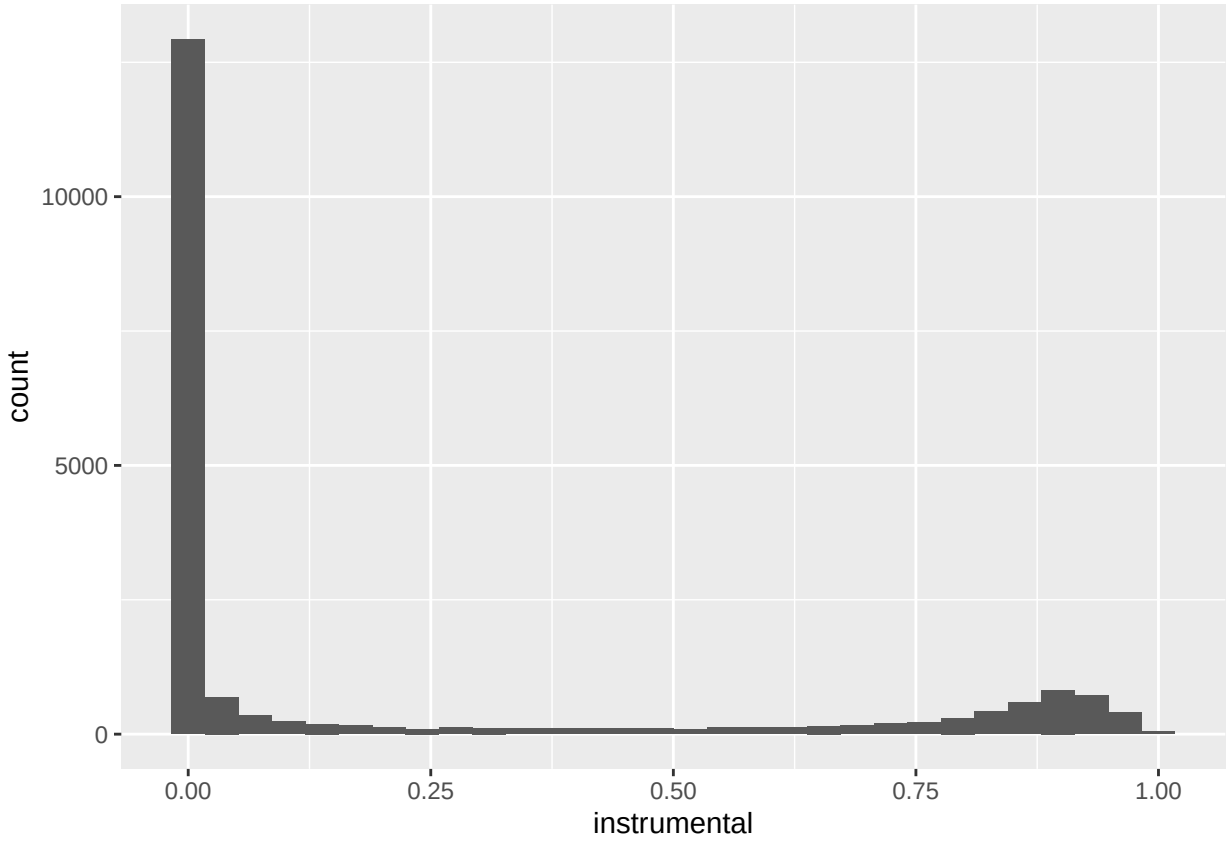

```
ggplot(data = All_tracks, aes(x = duration)) +
  geom_histogram()
```

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



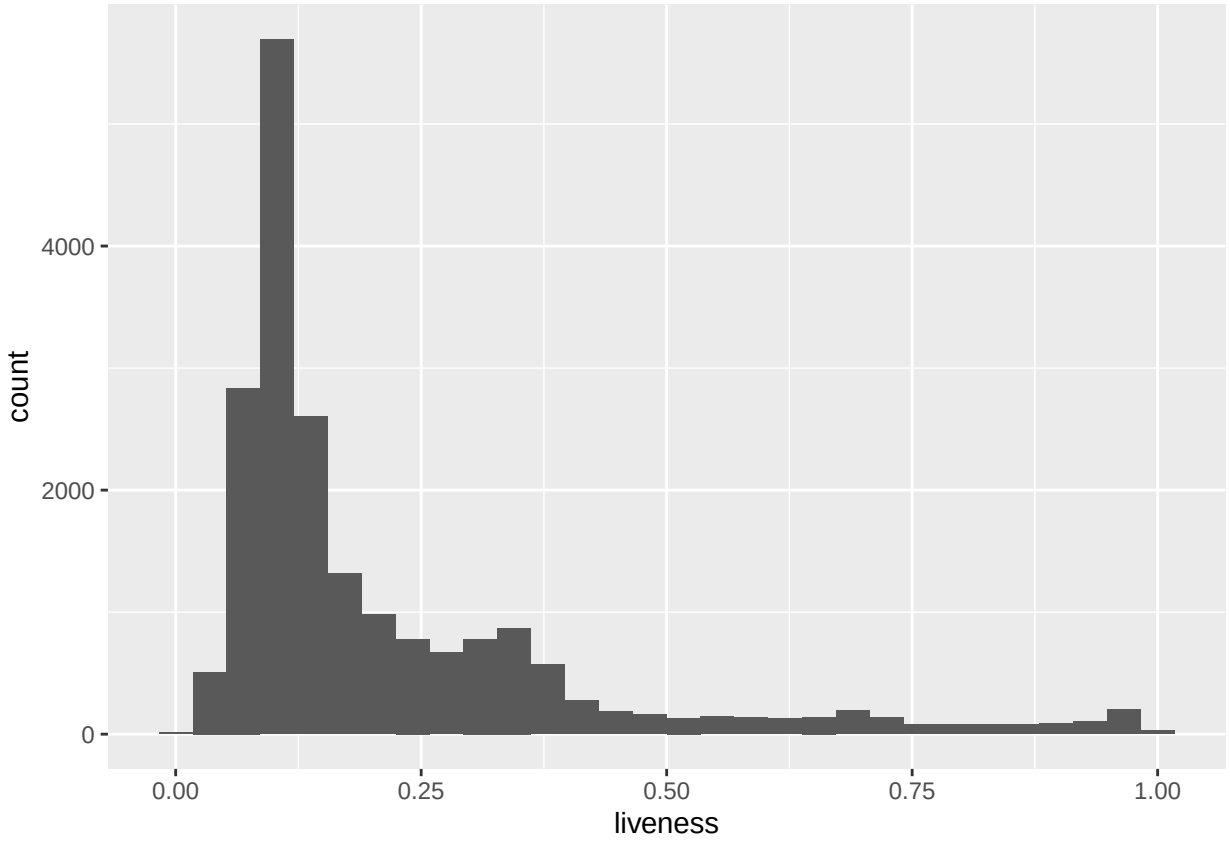

```
ggplot(data = All_tracks, aes(x = instrumental)) +
  geom_histogram()
```

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




```
ggplot(data = All_tracks, aes(x = liveness)) +
  geom_histogram()
```

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


4000
2000
0
count
0.00
0.25
0.50
0.75
1.00
liveness