Project 3

Cristian Sigala crs4565

This is the dataset used in this project:

```
spotify_songs <- readr::read_csv(
  'https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2020/2020-01-21/spotify_so
```

Link to the dataset: https://github.com/rfordatascience/tidytuesday/blob/master/data/2020/2020-01-21/readme.md

Part 1

Question: Is there a linear relationship between track popularity and duration of a song ? Which genres are typically most popular?

Introduction: Today we'll be using the spotify dataset in order to explore some data about songs performance. We'll be trying to see if there is any linear relationship between song popularity and duration of the song. Our second question is which music genre is most popular. For both of these problems we will be looking at these variables "track_popularity" which is the rank of a songs popularity in a 0-100 scale. The next variable is "playlist_genre" which describes the genre of the songs in six groups "edm, latin, rock, pop, r&b, and rap."

Approach: To answer the question we first need to further isolate variables by separating the 'playlist_genre' into its respective groups and with its associated data. Once we are able to do that, we will perform linear regressions on all these groups and get a summary table. In that summary table we will be able to see different variables that explain how well our model fits to the data. After that we will then graph these results to visually determine if there is a linear relationship between track popularity and duration. For our next question, we will be using a violin graph to visually determine which genre is typically the most popular. Ridgeline plots are the most effective for this question because it allows us to see the distribution of the data and easily compare with the other groups.

Analysis:

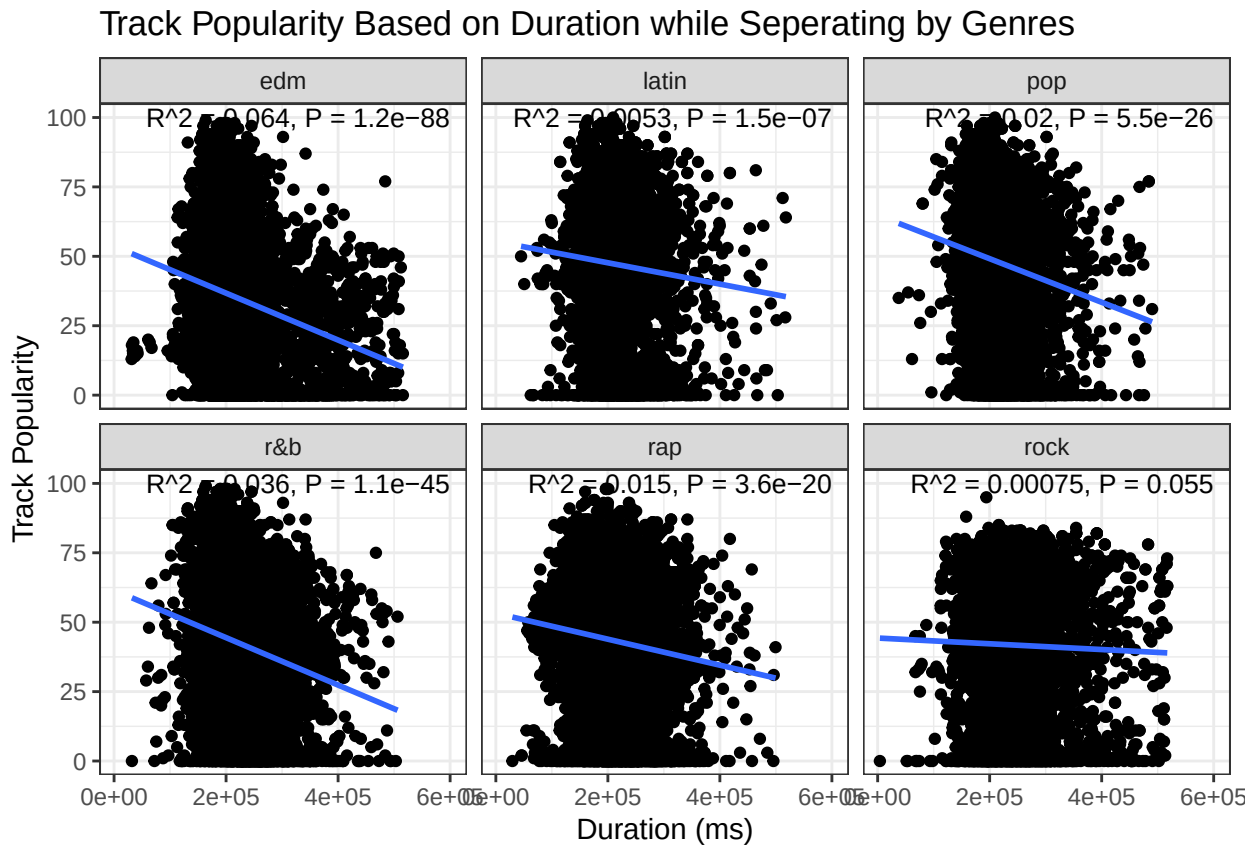
```
lm_data <- spotify_songs %>%
  nest(data = -playlist_genre) %>% #list-column the data frames
  mutate(
    # apply linear model to each nested data frame
    fit = map(data, ~lm(track_popularity ~ duration_ms, data = .x)),
  glance_out = map(fit, glance) #model-wide summary estimates
  ) %>%
  select(playlist_genre, glance_out) %>% #selecting variables
  unnest(cols = glance_out) # transforms data back to normal
lm_data
```

```
## # A tibble: 6 x 13
##   playlist_genre r.squared adj.r.squared sigma statistic  p.value    df  logLik
##   <chr>              <dbl>         <dbl> <dbl>     <dbl>    <dbl> <dbl>   <dbl>
## 1 pop             0.0200        0.0198    24.9    112.   5.52e-26     1 -25519.
## 2 rap             0.0146        0.0145    23.1     85.3  3.58e-20     1 -26202.
## 3 rock            0.000746      0.000544  24.8      3.69 5.47e- 2     1 -22925.
```

```
## 4 latin           0.00534       0.00515   25.4     27.7  1.49e- 7     1 -23980.
## 5 r&b             0.0364        0.0362    25.4    205.   1.14e-45     1 -25278.
## 6 edm             0.0638        0.0637    22.4    412.   1.16e-88     1 -27363.
## # ... with 5 more variables: AIC <dbl>, BIC <dbl>, deviance <dbl>,
## #   df.residual <int>, nobs <int>
```

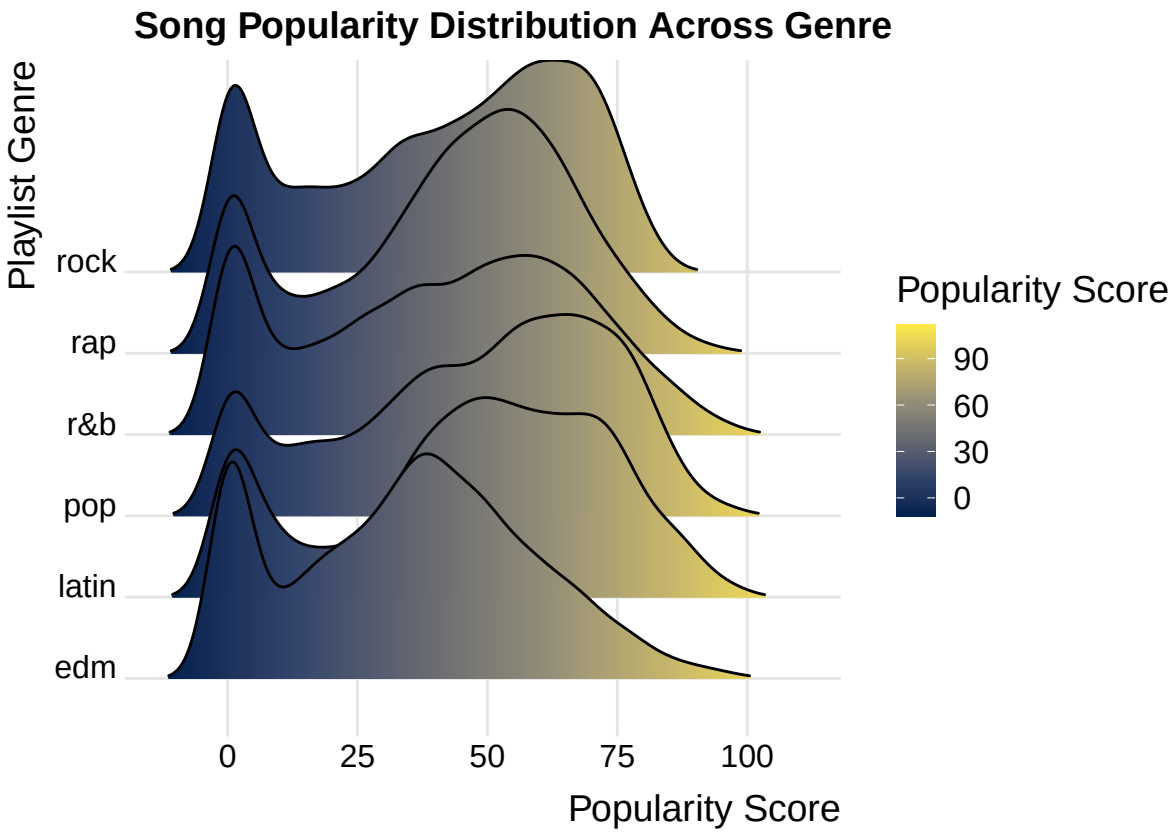
```
label_data <- lm_data %>%
  mutate(
    rsqr = signif(r.squared, 2),  # r^2
    pval = signif(p.value, 2), # p value
    label = glue("R^2 = {rsqr}, P = {pval}"), #attaching labels together
    duration_ms = 6e+05, track_popularity = 100 # label position in plot
  ) %>%
  select(playlist_genre, label, duration_ms, track_popularity) #selecting data

ggplot(spotify_songs, aes(duration_ms, track_popularity)) + #graph specifications
  geom_point() + #show points
  geom_text(
    data = label_data, aes(label = label), #using label data
    size = 10/.pt, hjust = 1  # placement of labels
  ) +
  geom_smooth(method = "lm", se = FALSE) + #applying linear regression
  facet_wrap(vars(playlist_genre)) +  #facetwrap over playlist_genre
  ylab("Track Popularity")+ #change y axis title
  xlab("Duration (ms)") + # change x axis title
  ggtitle("Track Popularity Based on Duration while Seperating by Genres") + #add title
  theme_bw() #change theme
```

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




```
ggplot(spotify_songs, aes(x = track_popularity, y = playlist_genre, fill = stat(x))) +
  #graph specifications
  geom_density_ridges_gradient(scale = 3, rel_min_height = 0.01) +
  #applying ridgeline gradient
  scale_fill_viridis_c(name = "Popularity Score", option = "E") +
  #adding color scale + editing legend
  theme_ridges() + #changing theme
  ylab("Playlist Genre") + #change y axis label
  xlab("Popularity Score") + # change X axis label
  #change title
  ggtitle("Song Popularity Distribution Across Genre")
```

```
## Picking joint bandwidth of 3.94
```



Discussion: After constructing our summary table, we can see that there is definitely some type of relationship between duration of a song and its popularity based on the very low p-values. However, when looking at the r-squared values, we see that none of the models really have a great fit. This tells us that although there is a relationship between the variables, there is NOT a linear relationship.

For the second question, we see that there is definitely distribution differences between the genres. To answer the question, I believe that the pop genre is typically most popular because it has a higher distribution closer to 100 than all the other groups. I also noticed that pop had the lowest distribution of songs in the 0 ranking of popularity.

Part 2

Question: Are there any variables that can predict track popularity and song duration?

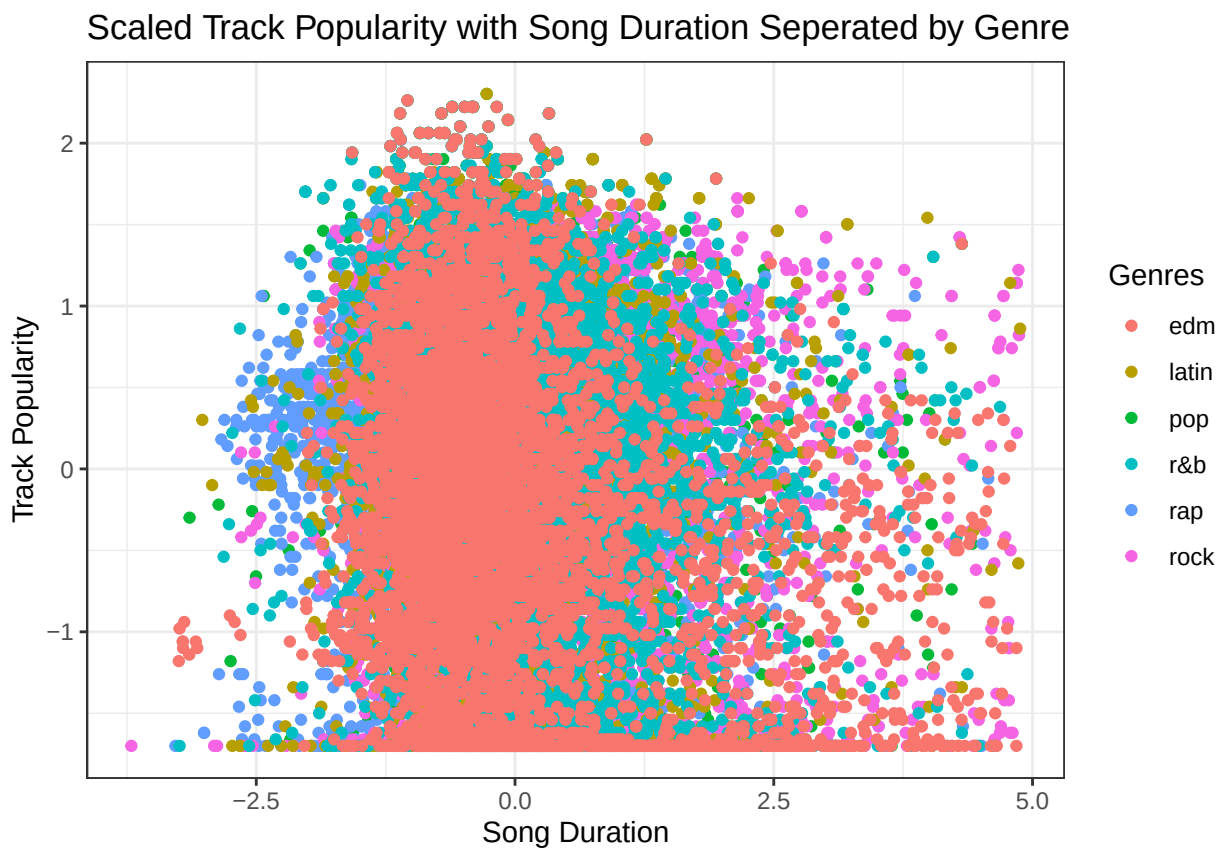
Introduction: While still using the spotify_songs dataset, we will now use all numeric variables in order to see which ones have weight in explaining track_popularity and song duration. These variables include "danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, and tempo". Each of these variables are numeric with their own type of scaling.

Approach: Since there are a lot of variables that have their own associated meanings and scaling, we must perform dimension reduction to effectively compare them. Principle component analysis (PCA) will be used because it reduces the dimension of the variables while still showing us the weight they have. Once plotted, we can identify which variables have the most weight in influencing track_popularity and duration_ms.

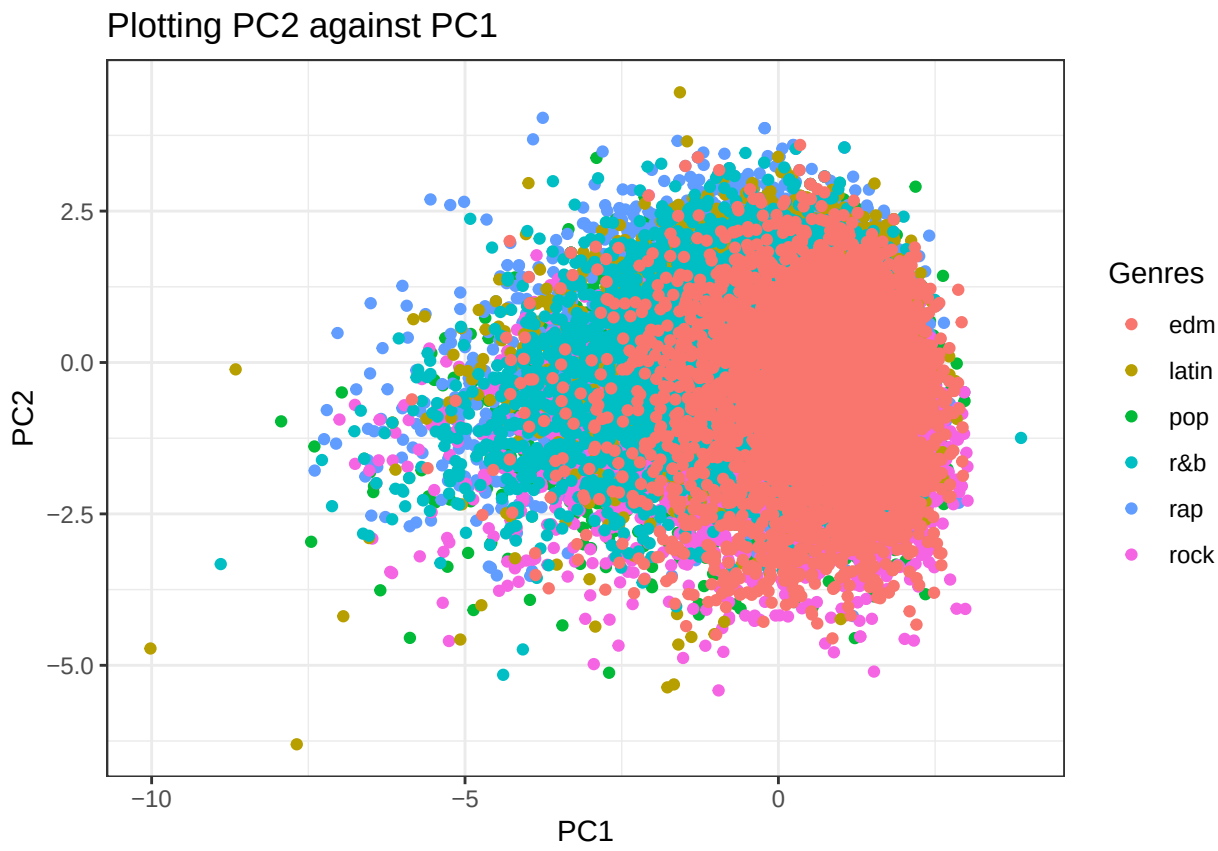
Analysis:

```
spotify_songs %>%
  # scaling numeric columns
```

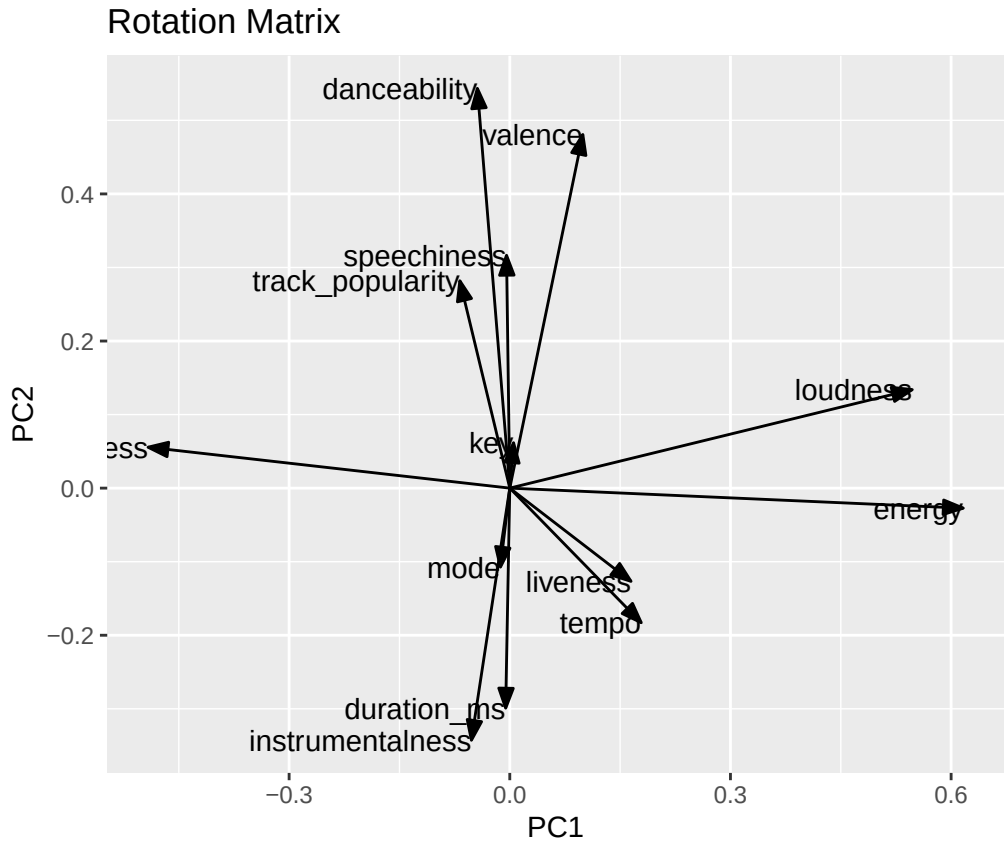
```
  mutate(across(where(is.numeric), scale)) %>%
  ggplot(aes(duration_ms, track_popularity)) + #graph specifications
  geom_point(aes(color = playlist_genre)) + # points separated by genre
  ylab("Track Popularity") + #change y axis title
  xlab("Song Duration") + #change x axis title
  #change title
  ggtitle ("Scaled Track Popularity with Song Duration Seperated by Genre") +
   labs(color='Genres') + #changed legend title
  theme_bw()#changed theme
```




```
pca_fit <- spotify_songs %>%
  select(where(is.numeric)) %>% #keep numeric columns
  scale() %>% # scaling
  prcomp() #PCA

pca_fit %>%
  # add PCs to the original dataset
  augment(spotify_songs) %>%
  ggplot(aes(.fittedPC1, .fittedPC2)) + #graph specifications
  geom_point(aes(color = playlist_genre)) + #show points
  ggtitle("Plotting PC2 against PC1") + #title
  ylab("PC2") + #Y axis title
  xlab("PC1") + # x axis title
  labs(color='Genres') +# legend title
  theme_bw()# change theme
```




```
arrow_style <- arrow(
  angle = 20, length = grid::unit(8, "pt"), #create arrows
  ends = "first", type = "closed"
)
pca_fit %>%
  # extract rotation matrix
  tidy(matrix = "rotation") %>%
  #widening data
  pivot_wider(
    names_from = "PC", values_from = "value",
    names_prefix = "PC"
  ) %>%
  ggplot(aes(PC1, PC2)) + #graph specifications
  #adding arrows to data
  geom_segment(
    xend = 0, yend = 0,
    arrow = arrow_style
  ) +
  #adding variable names to arrows
  geom_text(aes(label = column), hjust = 1) +
  coord_fixed() +#fix coordinates
  ggtitle("Rotation Matrix")
```



Discussion: To answer our question, the variable "acousticness" seems to have a very strong weight in influencing the duration of the song. For track popularity, variables "mode, liveness, tempo, and intrumentalness" have a strong weight on inluencing track popularity. This gives us insight to further test "acousticness" to predict song duration and "mode, liveness, tempo, and intrumentalness" to predict track popularity.