Exploratory Data Analysis of the Spotify Dataset

36558993 Ivan Josip Kardum, 36557629 Damjan Crnković

2026-02-10

This report presents an exploratory analysis of the "Spotify songs dataset (Kaggle)", which contains approximately 30,000 songs with described characteristics such as popularity, danceability, tempo, and key. The goal of the analysis is to examine relationships between genre, audio properties, and song popularity through statistical processing and data visualization, as well as to explore how these characteristics have changed over time. Additionally, models have been developed to predict song popularity based on their properties and genre.

When loading data from a CSV file, followed by basic analysis, we examine the structure of the dataset, the first few records, and a summary of variables to gain insight into their characteristics.

```
data <- read_csv("spotify_songs.csv")
```

```
## Rows: 32833 Columns: 23
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (10): track_id, track_name, track_artist, track_album_id, track_album_na...
## dbl (13): track_popularity, danceability, energy, key, loudness, mode, speec...
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
glimpse(data)
```

```
## Rows: 32,833
## Columns: 23
## $ track_id                <chr> "6f807x0ima9a1j3VPbc7VN", "0r7CVbZTWZgbTCYdfa~
## $ track_name              <chr> "I Don't Care (with Justin Bieber) - Loud Lux~
## $ track_artist            <chr> "Ed Sheeran", "Maroon 5", "Zara Larsson", "Th~
## $ track_popularity        <dbl> 66, 67, 70, 60, 69, 67, 62, 69, 68, 67, 58, 6~
## $ track_album_id          <chr> "2oCs0DGTsRO98Gh5ZSl2Cx", "63rPSO264uRjW1X5E6~
## $ track_album_name        <chr> "I Don't Care (with Justin Bieber) [Loud Luxu~
## $ track_album_release_date <chr> "2019-06-14", "2019-12-13", "2019-07-05", "20~
## $ playlist_name           <chr> "Pop Remix", "Pop Remix", "Pop Remix", "Pop R~
## $ playlist_id             <chr> "37i9dQZF1DXcZDD7cfEKhW", "37i9dQZF1DXcZDD7cf~
## $ playlist_genre          <chr> "pop", "pop", "pop", "pop", "pop", "pop", "po~
## $ playlist_subgenre       <chr> "dance pop", "dance pop", "dance pop", "dance~
## $ danceability            <dbl> 0.748, 0.726, 0.675, 0.718, 0.650, 0.675, 0.4~
## $ energy                  <dbl> 0.916, 0.815, 0.931, 0.930, 0.833, 0.919, 0.8~
## $ key                     <dbl> 6, 11, 1, 7, 1, 8, 5, 4, 8, 2, 6, 8, 1, 5, 5,~
## $ loudness                <dbl> -2.634, -4.969, -3.432, -3.778, -4.672, -5.38~
## $ mode                    <dbl> 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, ~
```

```
## $ speechiness              <dbl> 0.0583, 0.0373, 0.0742, 0.1020, 0.0359, 0.127~
## $ acousticness             <dbl> 0.10200, 0.07240, 0.07940, 0.02870, 0.08030, ~
## $ instrumentalness         <dbl> 0.00e+00, 4.21e-03, 2.33e-05, 9.43e-06, 0.00e~
## $ liveness                 <dbl> 0.0653, 0.3570, 0.1100, 0.2040, 0.0833, 0.143~
## $ valence                  <dbl> 0.518, 0.693, 0.613, 0.277, 0.725, 0.585, 0.1~
## $ tempo                    <dbl> 122.036, 99.972, 124.008, 121.956, 123.976, 1~
## $ duration_ms              <dbl> 194754, 162600, 176616, 169093, 189052, 16304~
```

```
# head(data, 10)
# summary(data)
```

The dataset contains 23 variables that describe songs, including audio features, genre, popularity, and album information. These variables enable analysis of relationships between song properties and their popularity.

Data Preparation

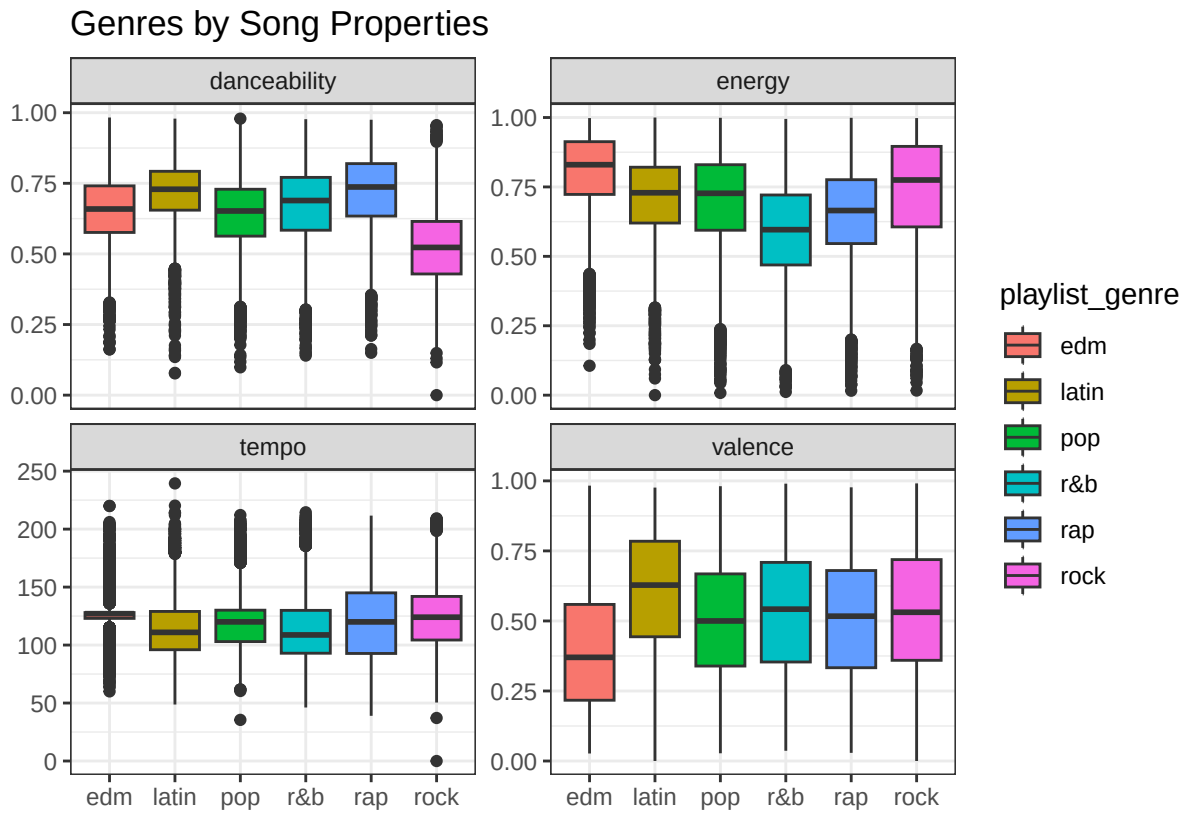
We notice that the date column needs to be converted to the correct type and that we have several categorical variables that need to be factorized so we can use them for grouping in visualizations and analysis as needed.

```
data$track_album_release_date <- as.Date(data$track_album_release_date)

data$mode <- factor(
  data$mode,
  levels = c(0, 1),
  labels = c("Minor", "Major")
)

factcols <- c("playlist_genre", "playlist_subgenre", "playlist_name", "track_artist")
data[factcols] <- lapply(data[factcols], as.factor)
```

1) How do audio properties differ across genres?

We are interested in which audio features characterize genres and how much genres differ in these properties. We examine the relationship between energy, danceability, valence, and tempo with genre.

```
data %>%
  dplyr::select(playlist_genre, energy, danceability, valence, tempo) %>%
  pivot_longer(-playlist_genre, names_to = "svojstva", values_to = "vrijednosti") %>%
  ggplot(aes(x = playlist_genre, y = vrijednosti, fill = playlist_genre)) +
  geom_boxplot() +
  facet_wrap(~ svojstva, scales = "free_y") +
  labs(title = "Genres by Song Properties",
       x = "",
       y = "") +
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5)) +
  theme_bw()
```

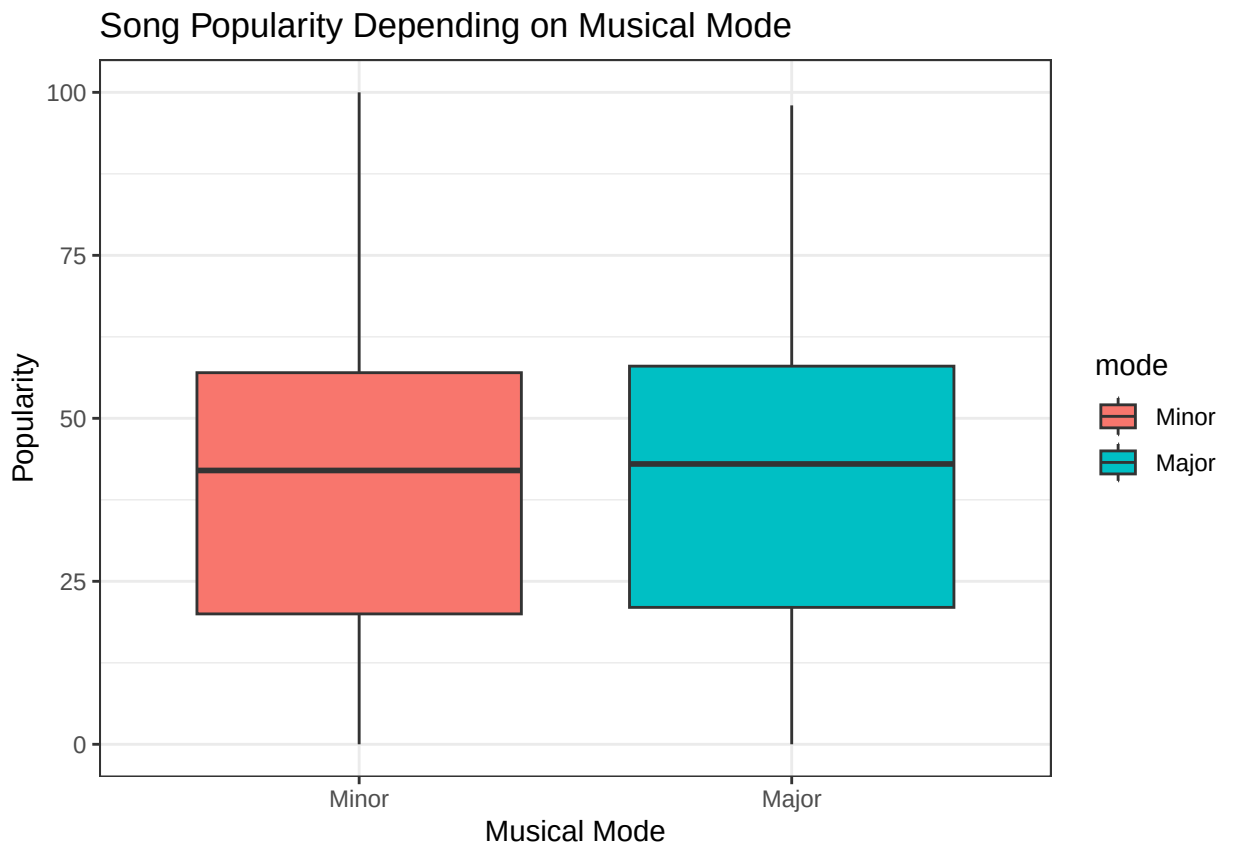


Reviewing the graphs, we can see that genres differ most in danceability and energy, where the differences in medians are visually obvious, while variability is approximately the same (visible from the interquartile range). Tempo shows small differences among genres and appears to have no significant influence in classifying a song into a particular genre. Valence is also similar among most genres, with the exception of the EDM genre which deviates with a low level of valence and the Latin genre which has a slightly higher level than others. The analysis was conducted at the playlist level, where the same song can appear in multiple genres, which is acceptable since the goal of the analysis is to compare characteristics by genre rather than individual songs.

2) Does mode (major/minor) affect song popularity?

Major scales in music sound happier and brighter than their corresponding minor scales, so we analyze whether there is a difference in the popularity of songs based on musical mode. We will remove multiple occurrences of specific songs so they don't negatively affect the results.

```
data2 <- distinct(data, track_id, .keep_all = T)

data2 %>% ggplot(aes(x = mode, y = track_popularity, fill = mode)) +
  geom_boxplot() +
  labs(title = "Song Popularity Depending on Musical Mode",
       x = "Musical Mode",
       y = "Popularity") +
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5)) +
  theme_bw()
```



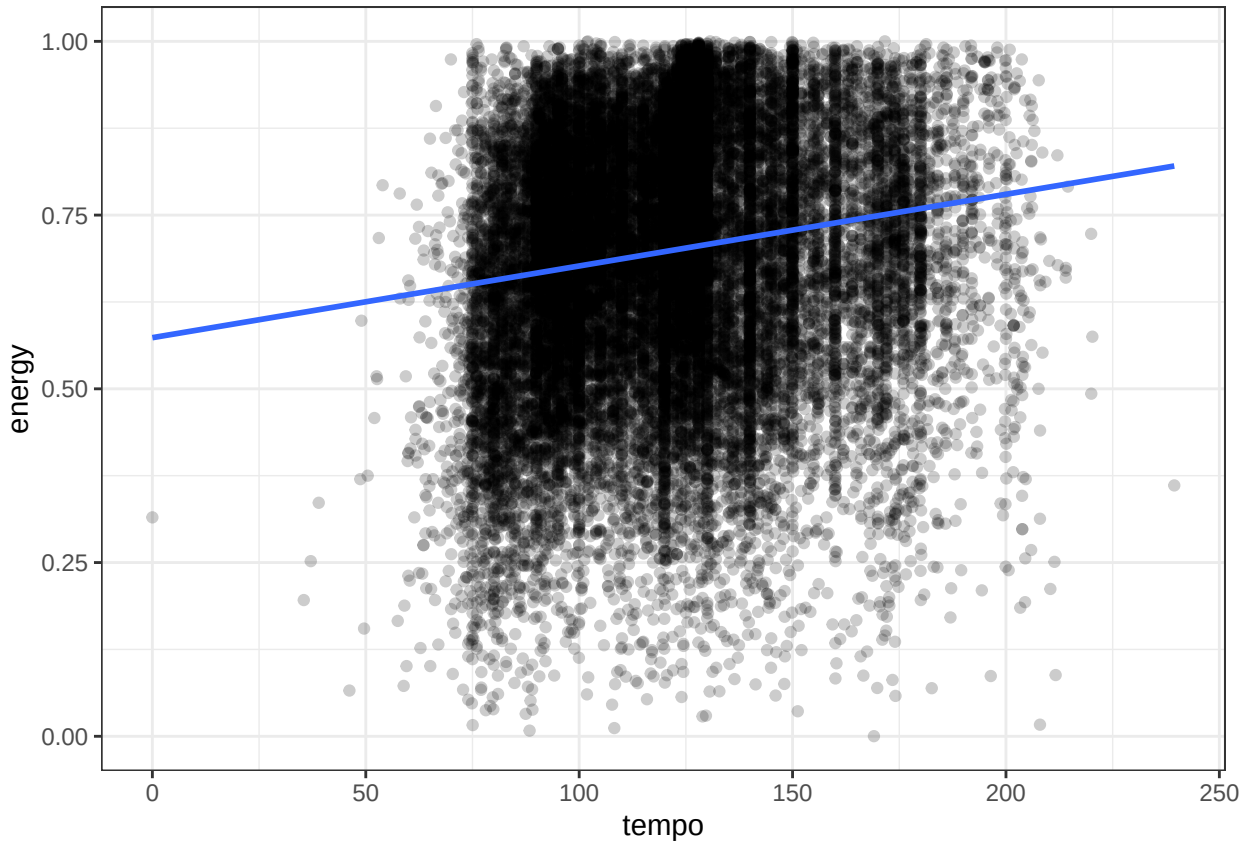
Reviewing the graphical representation, no clear difference in popularity is observed between songs written in major and minor. The popularity medians are very similar, and the distributions largely overlap, suggesting that musical mode by itself does not have a pronounced effect on song popularity.

3) Is song energy related to tempo and loudness?

Song energy is a property that is difficult to objectively define and is often defined using other properties. We are interested in whether there is a relationship between tempo and loudness and the perceived energy of songs. Again, we will analyze unique songs (without repetition) in the analysis. In the second graph, we limit ourselves to 99.9% of the data so that outliers don't disrupt the appearance of the graph.

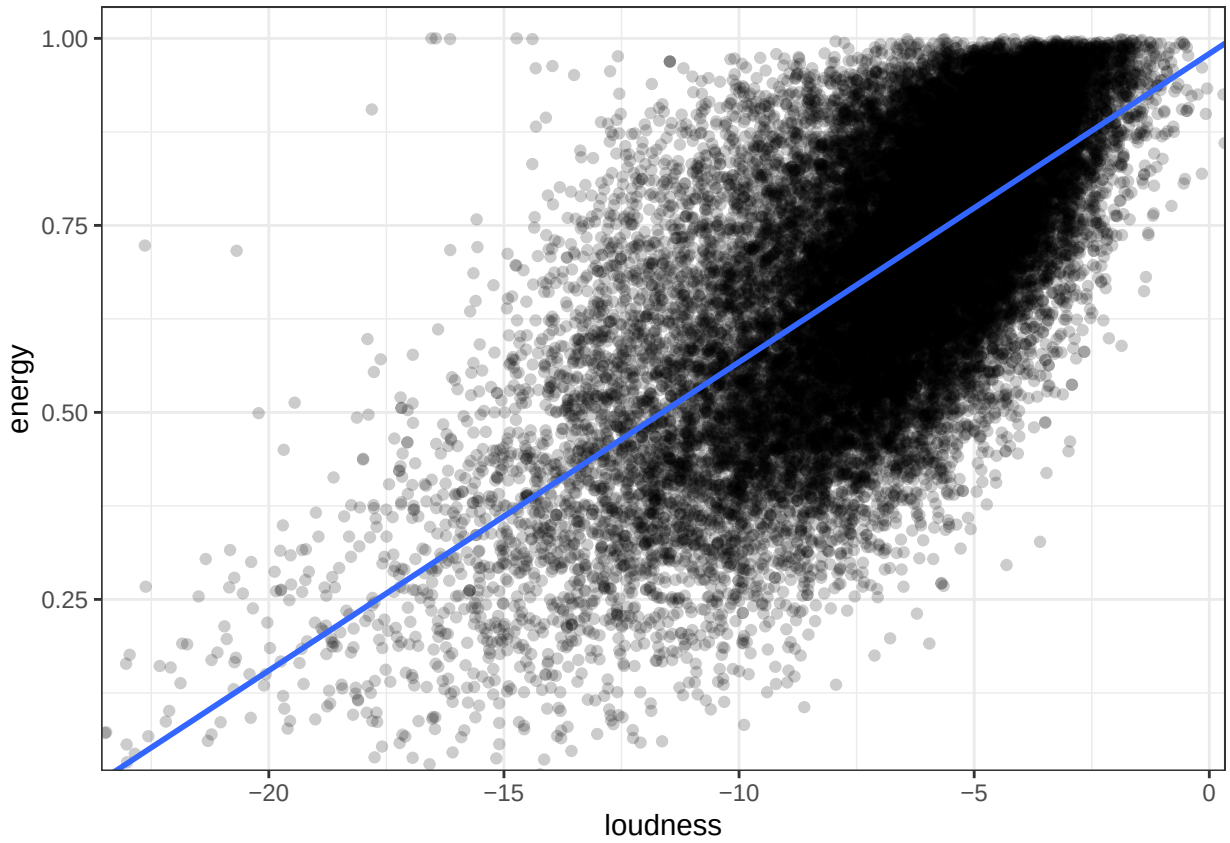
```
ggplot(data2, aes(x = tempo, y = energy)) +
  geom_point(alpha = 0.2) +
  geom_smooth(method = "lm", se = F) +
  theme_bw()
```

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




```
ggplot(data2, aes(x = loudness, y = energy)) +
  geom_point(alpha = 0.2) +
  coord_cartesian(
    xlim = quantile(data2$loudness, c(0.001, 0.999)),
    ylim = quantile(data2$energy, c(0.001, 0.999))
  ) +
  geom_smooth(method = "lm", se = F) +
  theme_bw()
```

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




```
model <- lm(energy ~ tempo + loudness, data = data2)
summary(model)
```

```
##
## Call:
## lm(formula = energy ~ tempo + loudness, data = data2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.54061 -0.08780  0.00655  0.09096  1.34283
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.9048821  0.0042160  214.63   <2e-16 ***
## tempo       0.0005881  0.0000295   19.94   <2e-16 ***
## loudness    0.0407218  0.0002619  155.51   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1333 on 28353 degrees of freedom
## Multiple R-squared:  0.4727, Adjusted R-squared:  0.4727
## F-statistic: 1.271e+04 on 2 and 28353 DF,  p-value: < 2.2e-16
```

The linear regression model confirms the observed relationships from the graphs. Loudness has a strong and positive effect on song energy, while the relationship between tempo and energy is statistically significant but

of much weaker intensity. The R^2 value shows that tempo and loudness together explain about 47% of energy variability, suggesting that song energy also depends on other properties not analyzed here. The results are consistent with expectations, since loudness is one of the key components of music energy perception, although the effect of tempo may have been smaller than expected.

4) How have song audio characteristics changed over time?

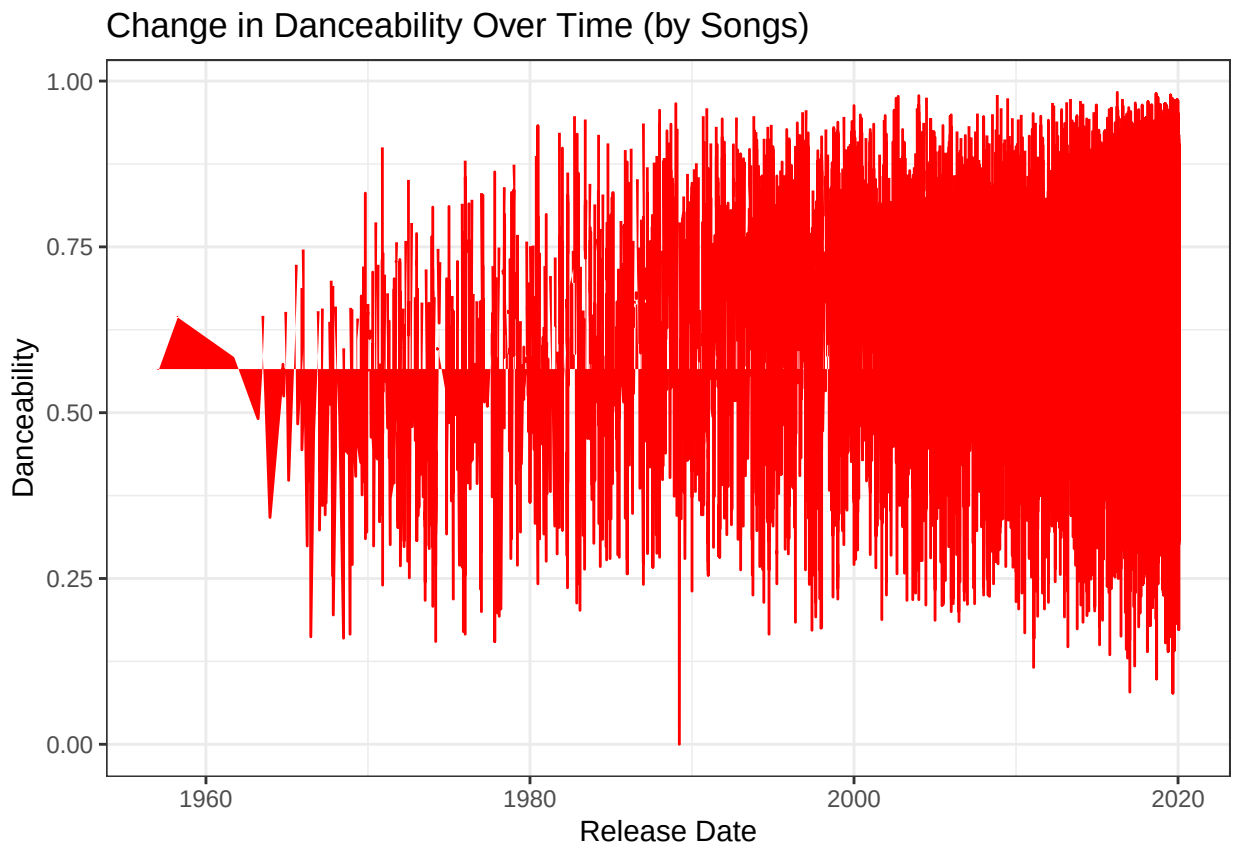
In this part of the analysis, the goal is to examine how key audio characteristics of songs (danceability, energy, and valence) have changed over the years. Since we are interested in the temporal trend, it is first necessary to prepare the data and extract the year from the album release date.

We remove records without a known release date from the dataset and extract the release year from the track_album_release_date variable.

```
data_modified <- data %>%
  filter(!is.na(data$track_album_release_date)) %>%
  mutate(relase_year = year(track_album_release_date))
```

As a starting step, we show the change in danceability. Such a display shows great variability because it contains all individual songs.

```
ggplot(data_modified, aes(x = track_album_release_date, y = danceability)) +
  geom_line(color = "red") +
  labs(
    title = "Change in Danceability Over Time (by Songs)",
    x = "Release Date",
    y = "Danceability"
  ) +
  theme_bw()
```



Although the graph contains a large amount of noise, it serves as motivation for aggregating data at the annual level.

To get a clearer picture of long-term trends, we aggregate data by year and calculate mean values of selected audio characteristics:

```
yearly <- data_modified %>%
  group_by(relase_year) %>%
  summarise(
    mean_d = mean(danceability),
    mean_e = mean(energy),
    mean_v = mean(valence)
  )
```

Before displaying, let's look at the number of data points per decade:

```
data_modified %>%
  mutate(decade = floor(relase_year / 10) * 10) %>%
  count(decade)
```

```
## # A tibble: 8 x 2
##   decade     n
##    <dbl> <int>
## 1   1950     2
## 2   1960   131
## 3   1970   646
## 4   1980   954
```

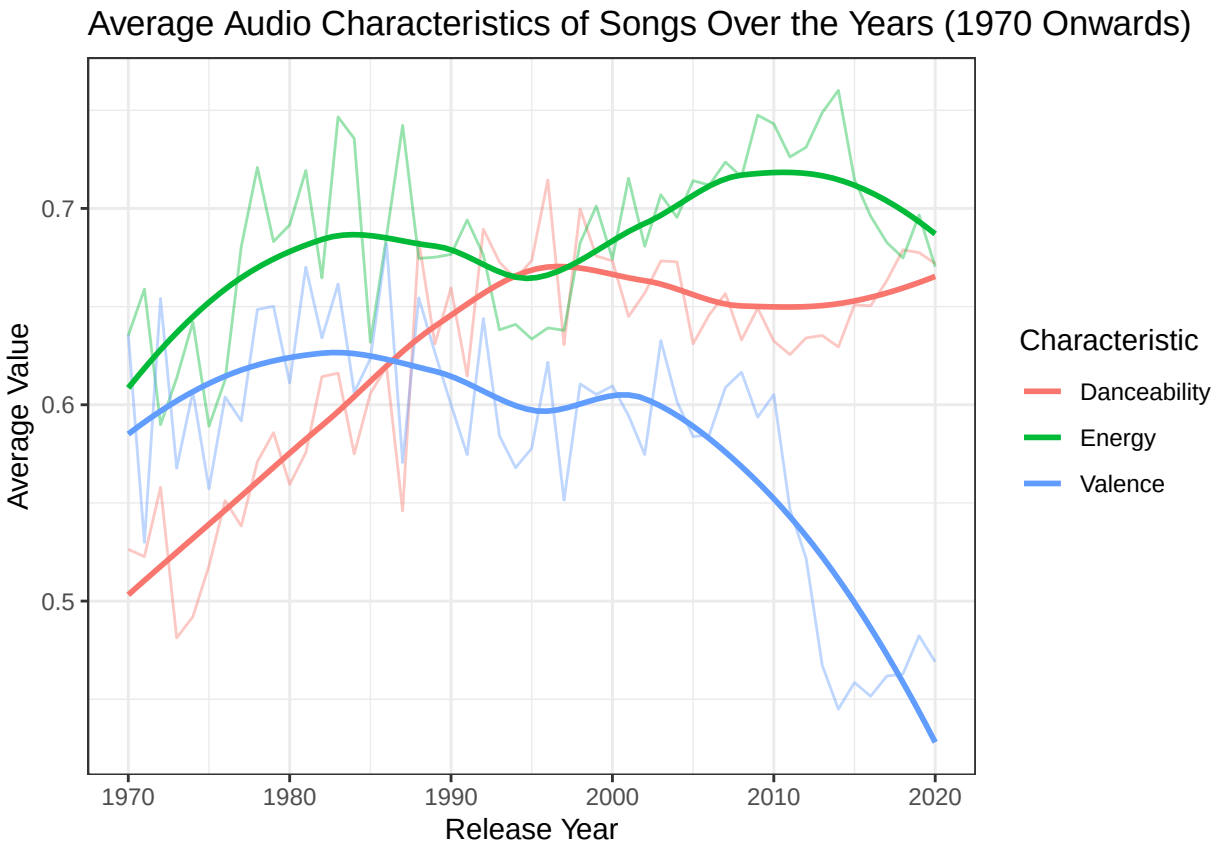
```
## 5   1990  1879
## 6   2000  3565
## 7   2010 22985
## 8   2020   785
```

We notice a very small number of data points in the fifties and sixties, which can cause inconsistency and noise in the graphical display.

Below, we simultaneously show changes in average danceability, energy, and emotional valence over the years, limiting the interval to 1970 onwards. The data is additionally smoothed to reduce the effect of annual oscillations and clearly highlight the long-term trend, while still maintaining an overview of actual changes over time.

```
ggplot(yearly %>% filter(relase_year >= 1970), aes(x = relase_year)) +
  geom_line(aes(y = mean_d, color = "Danceability"), alpha = 0.4) +
  geom_line(aes(y = mean_e, color = "Energy"), alpha = 0.4) +
  geom_line(aes(y = mean_v, color = "Valence"), alpha = 0.4) +
  geom_smooth(aes(y = mean_d, color = "Danceability"), se = FALSE) +
  geom_smooth(aes(y = mean_e, color = "Energy"), se = FALSE) +
  geom_smooth(aes(y = mean_v, color = "Valence"), se = FALSE) +
  labs(
    title = "Average Audio Characteristics of Songs Over the Years (1970 Onwards)",
    x = "Release Year",
    y = "Average Value",
    color = "Characteristic"
  ) +
  theme_bw()
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



Danceability experienced a significant increase in the period from 1970 to 1995, which is likely related to the popularity of dance and disco rhythms of that era, while after 1995 there is stagnation.

Energy gradually increases throughout the observed period, showing a continuous trend of more intense and dynamic productions in songs.

Emotional valence has been plummeting since 2000, suggesting that newer songs, although rhythmically and energetically stronger, are becoming emotionally more serious or less cheerful.

5) Which variables affect song popularity?

The goal of this part of the analysis is to explore which song characteristics affect its popularity. We use **track_popularity** as the target (dependent) variable, and all relevant numerical and categorical variables as predictors. First, we apply traditional linear regression, then more complex neural methods to identify the best model for prediction.

The first step in building a predictive model is to split the data into a training set and a test set. We use the training set to train the model, and the test set to check how well the model performs on new, unseen data.

```
set.seed(1234)
train_size <- 0.7 * nrow(data) %>% round
train_ind <- sample(1:nrow(data), train_size)

data_train <- data[train_ind,]
data_test <- data[-train_ind,]
```

Before modeling, we remove variables that don't help predict popularity, such as IDs, names, and dates.

```
data_train <- data_train %>%
  dplyr::select(-track_id, -track_name, -playlist_name, -track_album_id, -track_album_name, -track_artis
data_test <- data_test %>%
  dplyr::select(-track_id, -track_name, -playlist_name, -track_album_id, -track_album_name, -track_artis
```

Now we use multiple linear regression to estimate the effect of each variable on song popularity.

```
lmMod <- lm(track_popularity ~ ., data = data_train)
summary(lmMod)
```

```
##
## Call:
## lm(formula = track_popularity ~ ., data = data_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -59.385 -17.250   3.152  18.531  67.784
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           6.923e+01  2.055e+00  33.684  < 2e-16 ***
## playlist_genrelatin   8.466e+00  5.871e-01  14.421  < 2e-16 ***
## playlist_genrepop     9.965e+00  5.622e-01  17.726  < 2e-16 ***
## playlist_genrer&b     2.930e+00  6.078e-01   4.820 1.45e-06 ***
## playlist_genrerap     4.579e+00  5.918e-01   7.737 1.06e-14 ***
## playlist_genrerock    9.769e+00  6.285e-01  15.542  < 2e-16 ***
## danceability          1.063e+01  1.354e+00   7.851 4.30e-15 ***
## energy               -2.809e+01  1.457e+00 -19.281  < 2e-16 ***
## key                   6.892e-02  4.405e-02   1.564  0.11772
## loudness              1.627e+00  7.683e-02  21.179  < 2e-16 ***
## modeMajor             1.101e-01  3.242e-01   0.340  0.73423
## speechiness          -1.698e+00  1.771e+00  -0.959  0.33773
## acousticness          2.787e+00  8.689e-01   3.207  0.00134 **
## instrumentalness     -9.126e+00  7.741e-01 -11.790  < 2e-16 ***
## liveness             -3.203e+00  1.041e+00  -3.076  0.00210 **
## valence              -1.066e+00  7.932e-01  -1.344  0.17910
## tempo                 2.068e-02  6.049e-03   3.419  0.00063 ***
## duration_ms          -4.566e-05  2.727e-06 -16.743  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 23.76 on 22965 degrees of freedom
## Multiple R-squared:  0.09029,    Adjusted R-squared:  0.08961
## F-statistic: 134.1 on 17 and 22965 DF,  p-value: < 2.2e-16
```

The linear model shows that several variables, such as playlist genre, danceability, loudness, and instrumentalness, significantly affect song popularity. Some variables, such as key, mode, and speechiness, are not statistically significant and do not contribute to popularity prediction.

We define functions to calculate RMSE and R^2 to assess model accuracy:

```r
rmse <- function(pred, stv) {
  sqrt(mean((pred - stv)^2))
}

r_squared <- function(pred, true) {
  1 - sum((pred - true)^2) / sum((true - mean(true))^2)
}
```

We assess the accuracy of linear regression using RMSE and R^2 on the test set:

```r
data_test$predPopularityLM <- predict(lmMod, data_test)

rmse_lm <- rmse(data_test$predPopularityLM, data_test$track_popularity)
r2_lm <- r_squared(data_test$predPopularityLM, data_test$track_popularity)

cat("Linear Regression RMSE:", rmse_lm, "\n")
```

```
## Linear Regression RMSE: 24.00005
```

```r
cat("Linear Regression R²:", r2_lm, "\n")
```

```
## Linear Regression R²: 0.09059274
```

The results show that linear regression poorly predicts song popularity, with a large average error of about 24 (on a 1-100 scale). The low R^2 value (0.09) means the model explains only a small part of the variation in popularity.

Now we use random forest to predict song popularity, because after linear regression we want to improve prediction accuracy. Using cross-validation, we assess the reliability of the model, then predict popularity on the test set and calculate RMSE and R^2 to evaluate accuracy.

```r
ctrl <- trainControl(
  method = "repeatedcv",
  number = 5,
  repeats = 2,
  verboseIter = TRUE
)

rfMod <- train(
  track_popularity ~ .,
  data = data_train,
  method = 'ranger',
  tuneLength = 5,
  trControl = ctrl,
  num.trees = 20
)
```

```r
data_test$predPopularityRF <- predict(rfMod, data_test)

rmse_rf <- rmse(data_test$predPopularityRF, data_test$track_popularity)
r2_rf <- r_squared(data_test$predPopularityRF, data_test$track_popularity)

cat("Random Forest RMSE:", rmse_rf, "\n")
```

```
## Random Forest RMSE: 22.0656
```

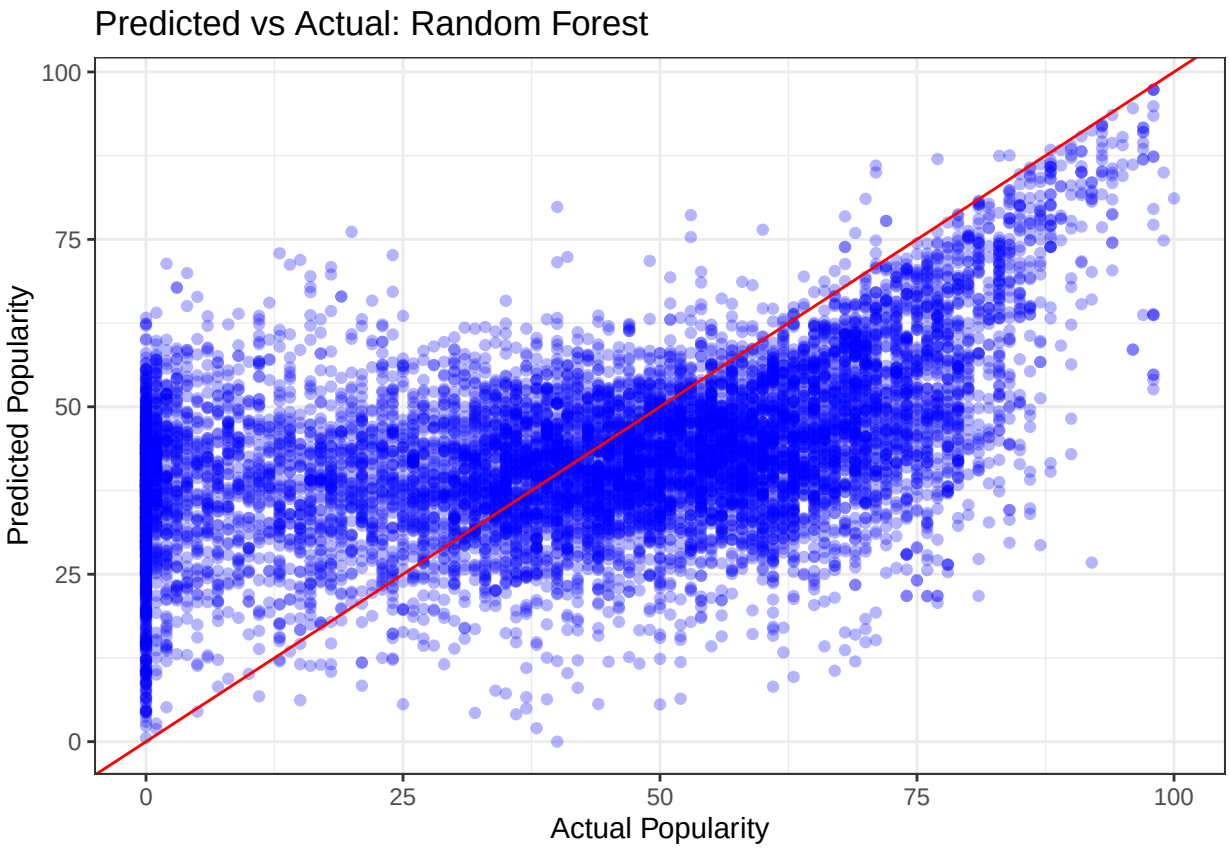
```
cat("Random Forest R²:", r2_rf, "\n")
```

```
## Random Forest R²: 0.2312849
```

The random forest model reduced RMSE compared to linear regression, meaning the model better predicts song popularity on average. The R^2 value more than doubled, showing that random forest explains significantly more variability in popularity. This confirms that a more complex model better captures nonlinear and mutual relationships between variables than the linear regression model.

Finally, let's visualize the comparison of actual and predicted popularity values using the random forest model.

```
ggplot(data_test, aes(x = track_popularity, y = predPopularityRF)) +
  geom_point(alpha = 0.3, color = "blue") +
  geom_abline(intercept = 0, slope = 1, color = "red") +
  labs(
    title = "Predicted vs Actual: Random Forest",
    x = "Actual Popularity",
    y = "Predicted Popularity"
  ) +
  theme_bw()
```



The predicted popularity values are quite scattered; for many songs with popularity close to zero, the model predicts significantly higher values. Although some points cluster around the diagonal, the deviations are large, which is consistent with high RMSE values and moderate R^2.

Conclusion

1) How do audio properties differ across genres?
The largest differences among genres are visible in danceability and energy, while tempo and valence remain relatively similar, with minor deviations of certain genres such as EDM and Latin.

2) Does mode (major/minor) affect song popularity?
Song popularity does not significantly depend on mode, as the medians and distributions of major and minor songs largely overlap.

3) Is song energy related to tempo and loudness?
Loudness strongly affects song energy, tempo has a weaker but statistically significant effect, and together they explain as much as 47% of energy variability, showing that these two characteristics have a significant impact on the perception of song energy.

4) How have song audio characteristics changed over time?
Danceability increased significantly until 1995, energy gradually increases throughout the period, while emotional valence has significantly declined since 2000, showing a trend of rhythmically and energetically stronger but emotionally more serious songs.

5) Which variables affect song popularity?
The linear model shows that playlist genre, danceability, loudness, and instrumentalness significantly affect song popularity, while key, mode, and speechiness have no significant effect. The random forest model better captures nonlinear and mutual relationships among variables and explains significantly more variability in popularity.