Main_project_file

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.0
v ggplot2   3.4.4     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becor
```

```
library(ggplot2)
library(knitr)
```

Importing data into R

```
taylor_album_songs <- read_csv("/Users/donyabehroozi/Desktop/stat365/stat-365/data/taylor_
```

```
Rows: 194 Columns: 29
-- Column specification --------------------------------------------------------
Delimiter: ","
chr   (7): album_name, track_name, artist, featuring, key_name, mode_name, k...
dbl  (14): track_number, danceability, energy, key, loudness, mode, speechin...
lgl   (4): ep, bonus_track, explicit, lyrics
date  (4): album_release, promotional_release, single_release, track_release
```

```
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.
```

```
taylor_ratings <- read_csv("/Users/donyabehroozi/Desktop/stat365/stat-365/data/taylor_albu
```

```
Rows: 14 Columns: 5
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (1): album_name
dbl  (2): metacritic_score, user_score
lgl  (1): ep
date (1): album_release

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.
```

Merging data sets together

```
taylor_albums_joined <- taylor_album_songs |>
  left_join(taylor_ratings, by = c("album_name", ep = "ep", album_release = "album_release

head(taylor_albums_joined)
```

```
# A tibble: 6 x 31
  album_name   ep    album_release track_number track_name      artist featuring
  <chr>        <lgl> <date>                <dbl> <chr>           <chr>  <chr>
1 Taylor Swift FALSE 2006-10-24                1 Tim McGraw      Taylo~ <NA>
2 Taylor Swift FALSE 2006-10-24                2 Picture To Burn Taylo~ <NA>
3 Taylor Swift FALSE 2006-10-24                3 Teardrops On M~ Taylo~ <NA>
4 Taylor Swift FALSE 2006-10-24                4 A Place In Thi~ Taylo~ <NA>
5 Taylor Swift FALSE 2006-10-24                5 Cold As You     Taylo~ <NA>
6 Taylor Swift FALSE 2006-10-24                6 The Outside     Taylo~ <NA>
# i 24 more variables: bonus_track <lgl>, promotional_release <date>,
#   single_release <date>, track_release <date>, danceability <dbl>,
#   energy <dbl>, key <dbl>, loudness <dbl>, mode <dbl>, speechiness <dbl>,
#   acousticness <dbl>, instrumentalness <dbl>, liveness <dbl>, valence <dbl>,
#   tempo <dbl>, time_signature <dbl>, duration_ms <dbl>, explicit <lgl>,
#   key_name <chr>, mode_name <chr>, key_mode <chr>, lyrics <lgl>,
#   metacritic_score <dbl>, user_score <dbl>
```

Linear regression analysis (danceability vs. user scores)

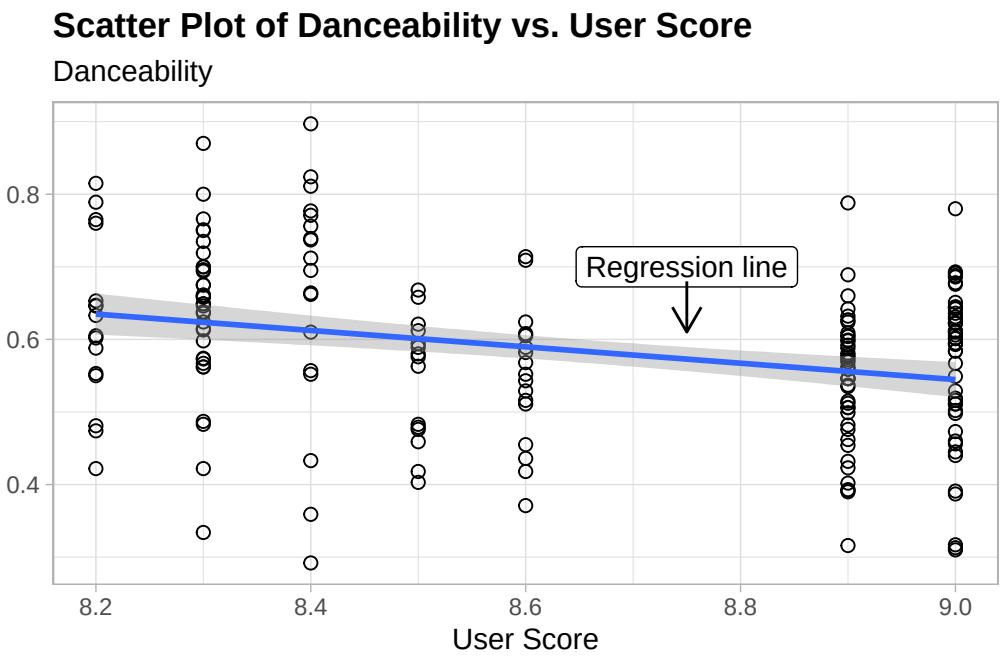
```
taylor.fit1=lm(danceability~user_score, data = taylor_albums_joined)

taylor_albums_joined |>
  ggplot(aes(x = user_score, y = danceability)) +
  geom_point(size = 2, shape = 1) +
  geom_smooth(method = "lm") +
  labs(title = "Scatter Plot of Danceability vs. User Score", x = "User Score", y = " ", s
  theme_light() +
  theme(plot.title = element_text(face = "bold")) +
  annotate(geom = "label",
           x = 8.75,
           y = 0.7,
           label = "Regression line") +
  annotate(geom = "segment",
           x = 8.75,
           y = 0.68,
           xend = 8.75,
           yend = 0.61,
           arrow = arrow(length = unit(0.15, "inches"))) +
  scale_y_continuous(breaks = c(0.2, 0.4, 0.6,  0.8, 1.0))
```

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

```
Warning: Removed 3 rows containing non-finite values (`stat_smooth()`).
```

```
Warning: Removed 3 rows containing missing values (`geom_point()`).
```



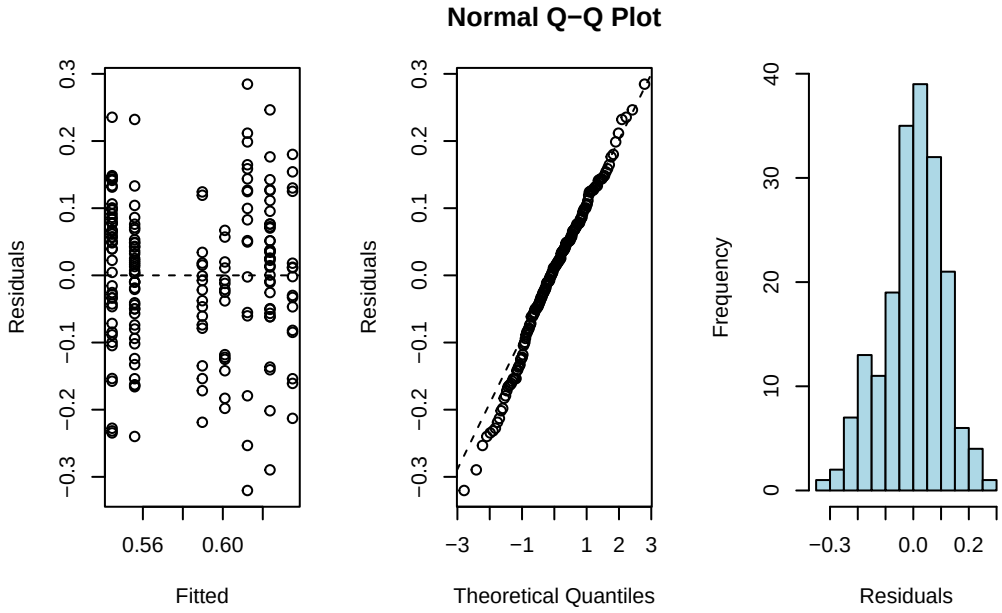
Hypotheses

H_0: Audience rating is not linearly associated with danceability score.

H_A: Audience rating is linearly associated with danceability score.

Checking simple linear regression assumptions

```
par(mfrow=c(1,3))
plot(resid(taylor.fit1)~fitted(taylor.fit1), xlab='Fitted', ylab='Residuals');abline(h=0,l
qqnorm(resid(taylor.fit1),ylab='Residuals'); qqline(resid(taylor.fit1),lty=2)
hist(resid(taylor.fit1),main="", xlab="Residuals",col='lightblue')
```



Results and conclusion

```
summary(taylor.fit1)
```

```
Call:
lm(formula = danceability ~ user_score, data = taylor_albums_joined)

Residuals:
     Min       1Q   Median       3Q      Max
-0.32032 -0.06053  0.01211  0.07175  0.28468

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.56037    0.22873   6.822 1.18e-10 ***
user_score  -0.11286    0.02642  -4.271 3.08e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1105 on 189 degrees of freedom
```

```
  (3 observations deleted due to missingness)
Multiple R-squared:  0.08803,   Adjusted R-squared:  0.0832
F-statistic: 18.24 on 1 and 189 DF,  p-value: 3.076e-05
```

```r
confint(taylor.fit1, level = 0.95)
```

```
                2.5 %      97.5 %
(Intercept)  1.109183  2.01155811
user_score  -0.164987 -0.06073933
```

Linear regression analysis (danceability vs. metacritic scores)

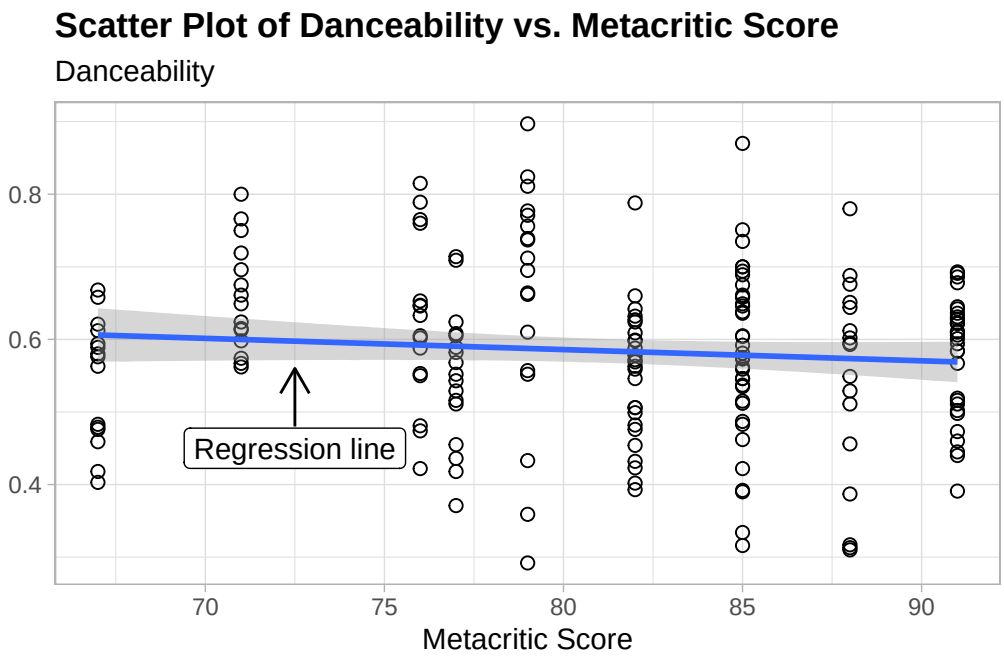
```r
taylor.fit2=lm(danceability~metacritic_score, data = taylor_albums_joined)

taylor_albums_joined |>
  ggplot(aes(x = metacritic_score, y = danceability)) +
  geom_point(size = 2, shape = 1) +
  geom_smooth(method = "lm") +
  labs(title = "Scatter Plot of Danceability vs. Metacritic Score", x = "Metacritic Score"
  theme_light() +
  theme(plot.title = element_text(face = "bold")) +
  annotate(geom = "label",
           x = 72.5,
           y = 0.45,
           label = "Regression line")  +
  annotate(geom = "segment",
           x = 72.5,
           y = 0.48,
           xend = 72.5,
           yend = 0.56,
           arrow = arrow(length = unit(0.15, "inches"))) +
  scale_y_continuous(breaks = c(0.2, 0.4, 0.6,  0.8, 1.0))
```

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

```
Warning: Removed 3 rows containing non-finite values (`stat_smooth()`).
```

```
Warning: Removed 3 rows containing missing values (`geom_point()`).
```



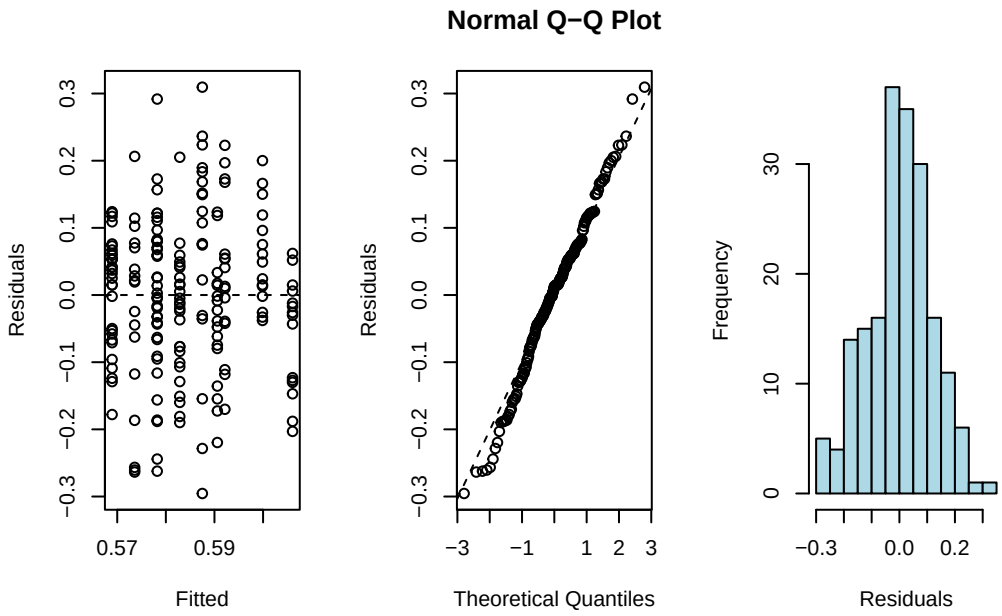
Hypotheses

H_0: Metacritic rating is not linearly associated with danceability score.

H_A: Metacritic rating is linearly associated with danceability score.

Checking simple linear regression assumptions

```
par(mfrow=c(1,3))
plot(resid(taylor.fit2)~fitted(taylor.fit2), xlab='Fitted', ylab='Residuals');abline(h=0,l
qqnorm(resid(taylor.fit2),ylab='Residuals'); qqline(resid(taylor.fit2),lty=2)
hist(resid(taylor.fit2),main="", xlab="Residuals",col='lightblue')
```



Results and conclusion

```
summary(taylor.fit2)
```

```
Call:
lm(formula = danceability ~ metacritic_score, data = taylor_albums_joined)

Residuals:
     Min       1Q   Median       3Q      Max
-0.29551 -0.06659  0.01286  0.07059  0.30950

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       0.709668   0.095748   7.412 4.05e-12 ***
metacritic_score -0.001546   0.001174  -1.317    0.189
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1151 on 189 degrees of freedom
```

```
  (3 observations deleted due to missingness)
Multiple R-squared:  0.009098,  Adjusted R-squared:  0.003855
F-statistic: 1.735 on 1 and 189 DF,  p-value: 0.1893
```

```r
confint(taylor.fit2, level = 0.95)
```

```
                      2.5 %       97.5 %
(Intercept)       0.520796591 0.8985390672
metacritic_score -0.003861928 0.0007692015
```

Summary statistics table

```r
taylor_albums_joined |>
  group_by(album_name) |>
  summarise(mean_dance = mean(danceability, na.rm = TRUE), sd_dance = sd(danceability, na.
  arrange(desc(mean_dance)) |>
  rename("Album Name" = album_name, "Mean Danceability" = mean_dance, "Standard dev. Dance
  kable()
```

Album Name	Mean Danceability	Standard dev. Danceability	Mean User Score	Mean Critic Score
Lover	0.6582222	0.1644029	8.4	79
reputation	0.6579333	0.0751280	8.3	71
Midnights	0.6266500	0.1220605	8.3	85
1989	0.6239375	0.1155938	8.2	76
Red (Taylor's Version)	0.5769667	0.0820633	9.0	91
Fearless (Taylor's Version)	0.5510385	0.0914831	8.9	82
Speak Now	0.5488235	0.0940560	8.6	77
Taylor Swift	0.5452667	0.0852136	8.5	67
folklore	0.5419412	0.1417747	9.0	88
evermore	0.5268235	0.0921006	8.9	85