Short Report 1

Dongyoung Kim and Kunwu Lyu

2024-10-18

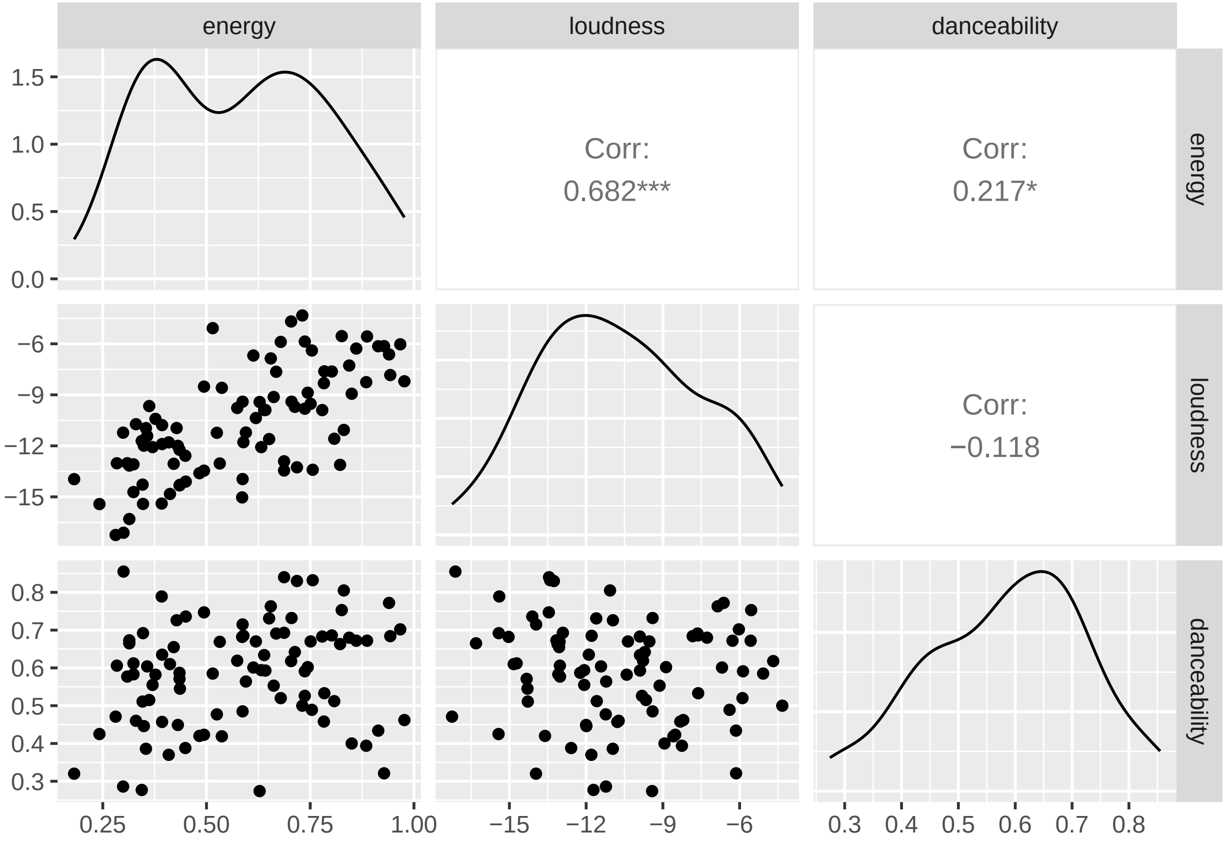
Data Wrangling

```
#first spotify data, with crossover variable
#BE SURE TO DELETE THE LINE BREAK BEFORE RUNNING THIS
spotify_data_crossover <- read.csv(file =
                                     "https://www.math.carleton.edu/ckelling/regression/report_cross_spo
  mutate(mode = as.factor(mode),
         key = as.factor(key),
         crossover_categ = as.factor(crossover_categ))

#second spotify dataset- all data, crossover labelled or not
#BE SURE TO DELETE THE LINE BREAK BEFORE RUNNING THIS
full_musc_data <- read.csv(file = "https://www.math.carleton.edu/ckelling/regression/report_nocross_spot
  mutate(mode = as.factor(mode),
         key = as.factor(key))

# select specified artists and create a new data frame
artists <- c("Gladys Knight & The Pips", "Stevie Wonder", "The Temptations")
spotify_data_crossover_filtered <- spotify_data_crossover %>%
  filter(artist_name %in%
           c("Gladys Knight & The Pips", "Stevie Wonder", "The Temptations"))

glimpse(spotify_data_crossover_filtered)
```

```
## Rows: 90
## Columns: 21
## $ artist_name        <chr> "The Temptations", "The Temptations", "The Temptati~
## $ album_release_year <int> 1980, 1980, 1980, 1978, 1978, 1977, 1977, 1976, 197~
## $ danceability       <dbl> 0.515, 0.832, 0.731, 0.805, 0.577, 0.634, 0.726, 0.~
## $ energy             <dbl> 0.362, 0.756, 0.651, 0.831, 0.309, 0.639, 0.428, 0.~
## $ key                <fct> 0, 1, 3, 5, 10, 9, 0, 7, 1, 8, 6, 10, 2, 8, 7, 10, ~
## $ loudness           <dbl> -9.661, -13.407, -11.603, -11.065, -13.021, -9.893,~
## $ mode               <fct> 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0, ~
## $ speechiness        <dbl> 0.0267, 0.0941, 0.0311, 0.0761, 0.0315, 0.1720, 0.0~
## $ acousticness       <dbl> 0.5450, 0.2620, 0.2040, 0.1040, 0.4770, 0.5950, 0.1~
## $ instrumentalness   <dbl> 1.33e-03, 1.30e-02, 4.12e-06, 2.82e-03, 1.28e-04, 1~
## $ liveness           <dbl> 0.0813, 0.1970, 0.1250, 0.0700, 0.1800, 0.3540, 0.1~
## $ time_signature     <int> 3, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, ~
## $ valence            <dbl> 0.364, 0.763, 0.878, 0.769, 0.759, 0.824, 0.939, 0.~
## $ tempo              <dbl> 131.433, 125.105, 101.383, 115.842, 173.073, 118.98~
## $ explicit           <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FA~
## $ key_name           <chr> "C", "C#", "D#", "F", "A#", "A", "C", "G", "C#", "G~
## $ mode_name          <chr> "major", "major", "major", "minor", "major", "minor~
## $ rnb_chart          <int> 19, 11, 55, 42, 31, 58, 21, 21, 22, 3, 1, 9, 1, 1, ~
## $ pop_chart          <int> NA, 43, NA, NA, NA, NA, NA, 94, NA, 54, 40, 37, 26,~
## $ duration_ms        <int> 365720, 367093, 254280, 219160, 248720, 300413, 195~
## $ crossover_categ    <fct> N, N, N, N, N, N, N, N, N, N, N, N, N, N, N, N, Y, ~
```

EDA

```
# scatterplot matrix
ggpairs(spotify_data_crossover_filtered,
        columns = c("energy", "loudness", "danceability"))
```

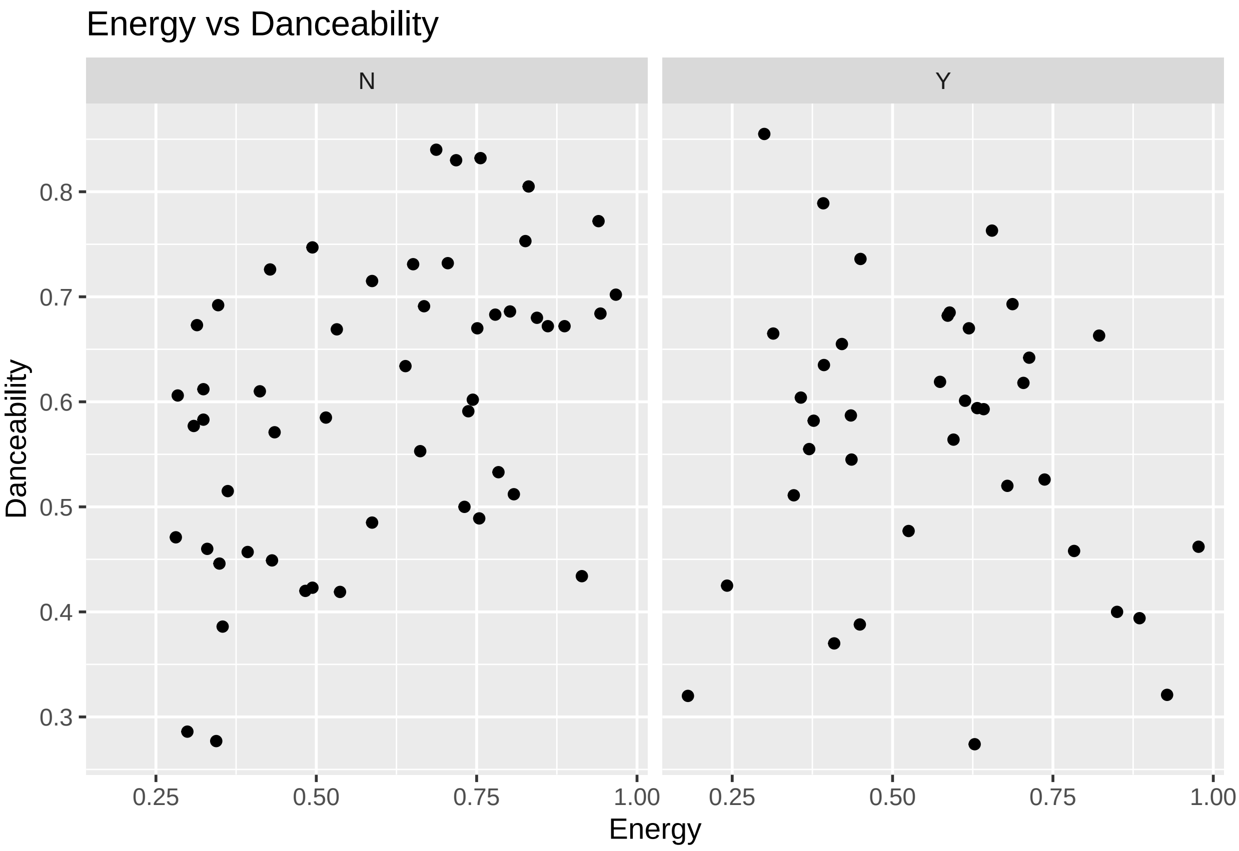


```
ggsave("scat_mtrx.png") # save as png for better importing
```

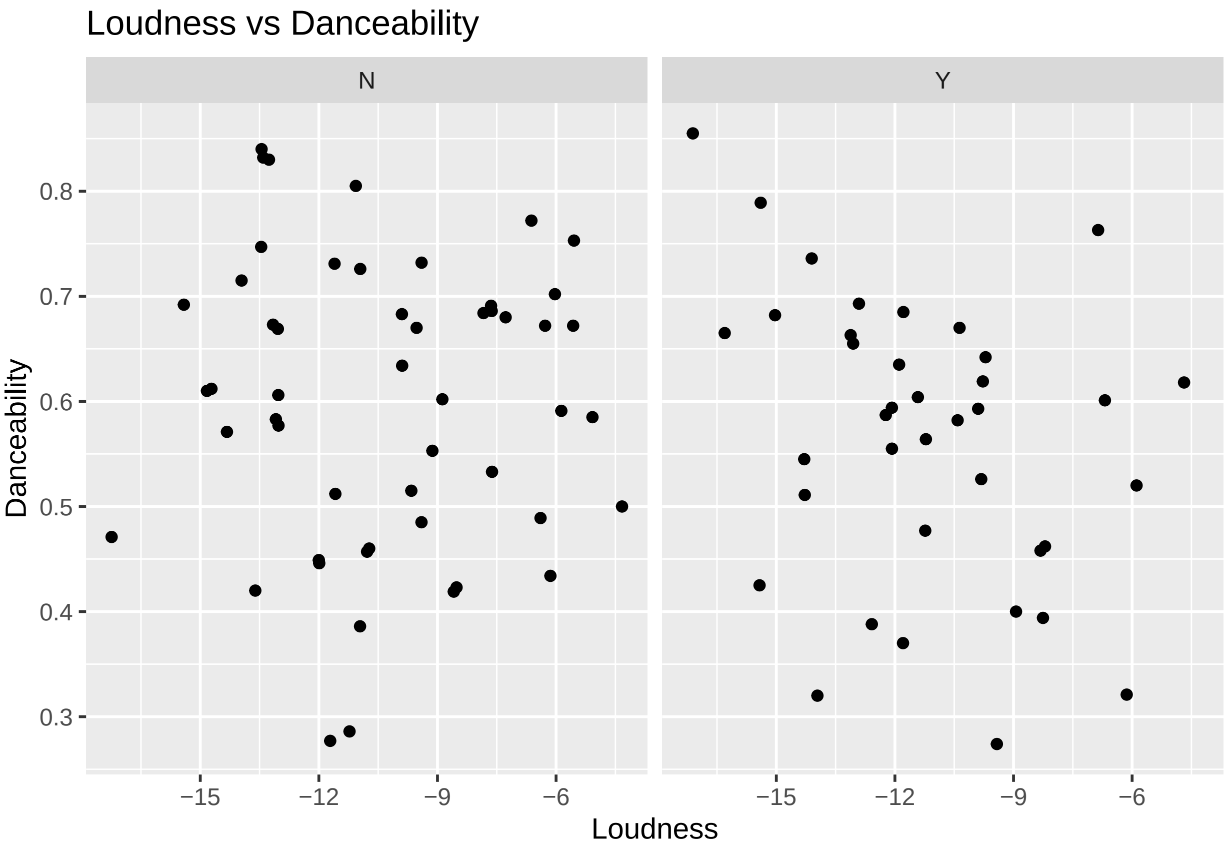
```
## Saving 6.5 x 4.5 in image
```

```
# Vis
# EDA with energy vs danceability
eda_energy <- ggplot(data = spotify_data_crossover_filtered,
                     aes(x = energy, y = danceability)) +
  geom_point() +
  facet_grid(cols = vars(crossover_categ)) + # separate by crossover
  labs(x = "Energy", y = "Danceability", title = "Energy vs Danceability")

# EDA with loudness vs danceability
eda_loudness <- ggplot(data = spotify_data_crossover_filtered,
                       aes(x = loudness, y = danceability)) +
  geom_point() +
  labs(x = "Loudness", y = "Danceability", title = "Loudness vs Danceability") +
  facet_grid(cols = vars(crossover_categ))  # separate by crossover
```

```
eda_energy
```



```
eda_loudness
```



```
ggsave("eda_energy.png", eda_energy)
```

```
## Saving 6.5 x 4.5 in image
```

```
ggsave("eda_loudness.png", eda_loudness)
```

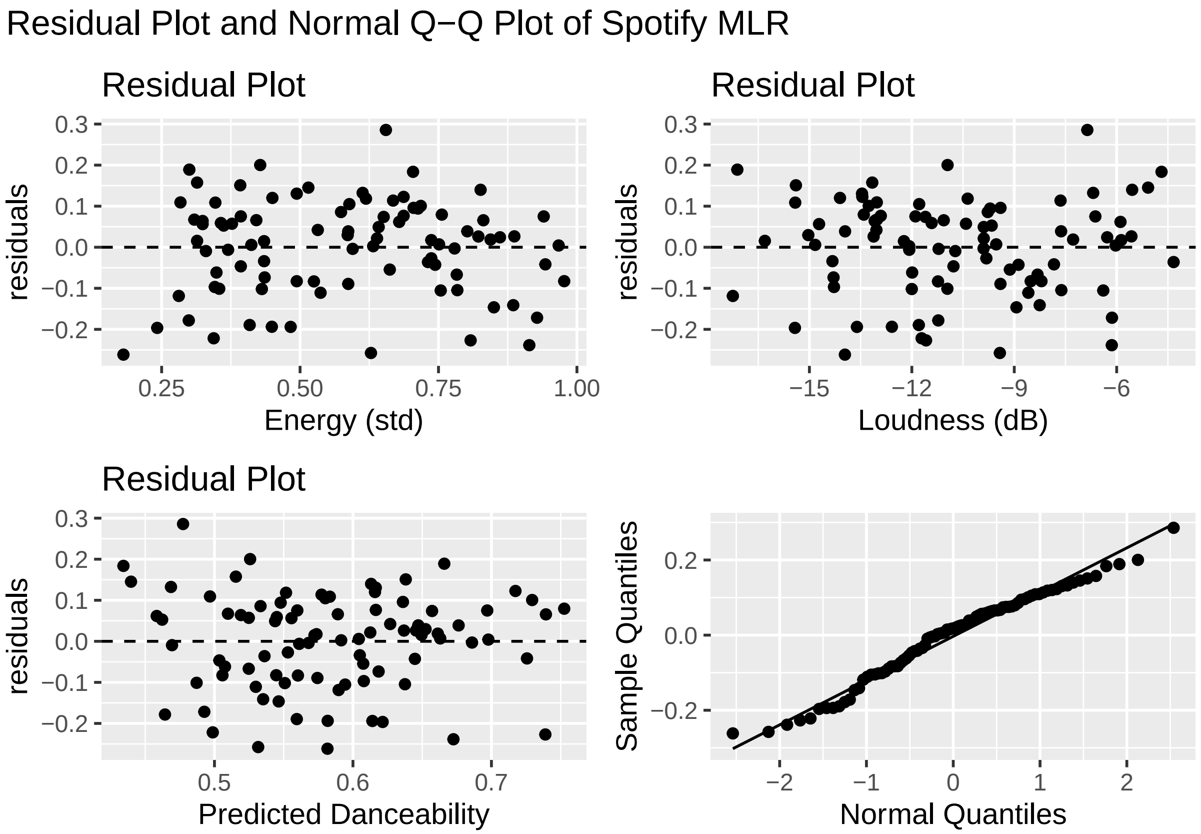
```
## Saving 6.5 x 4.5 in image
```

MLR and Assumptions

```
# MLR Model
danceability_lm <- lm(danceability ~ loudness + energy * crossover_categ,
                      data = spotify_data_crossover_filtered)
summary(danceability_lm)
```

```
##
## Call:
## lm(formula = danceability ~ loudness + energy * crossover_categ,
##     data = spotify_data_crossover_filtered)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.26160 -0.08273  0.01803  0.07615  0.28569
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)             0.056170   0.102524   0.548 0.585213
## loudness               -0.022404   0.005566  -4.025 0.000123 ***
## energy                    0.523810   0.095190   5.503 3.88e-07 ***
## crossover_categY          0.191747   0.076673   2.501 0.014309 *
## energy:crossover_categY -0.408225   0.125784  -3.245 0.001678 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1184 on 85 degrees of freedom
## Multiple R-squared:  0.291,  Adjusted R-squared:  0.2577
## F-statistic: 8.723 on 4 and 85 DF,  p-value: 5.993e-06
```

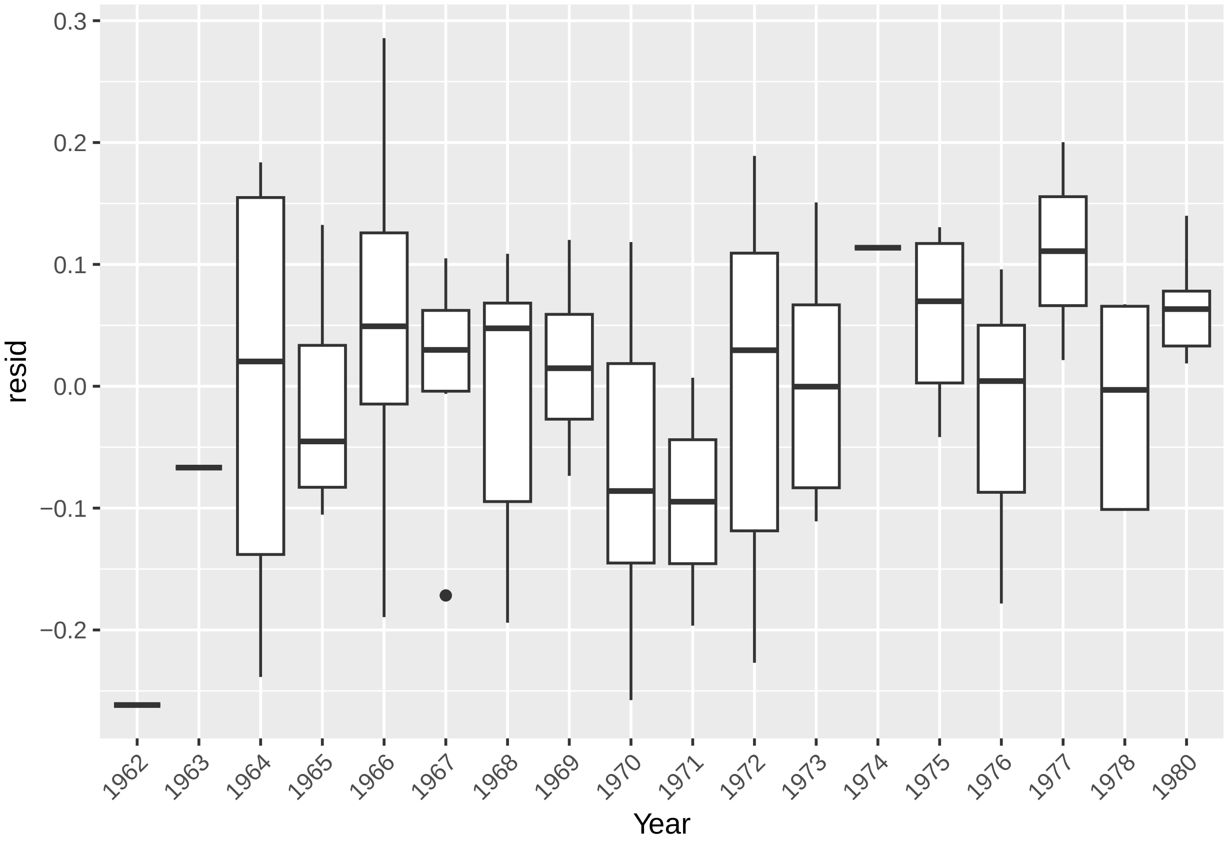
```r
# Assumptions Checking
danceability_lm_aug <- augment(danceability_lm, # add original data for later
                                                # checking assumptions
                               data = spotify_data_crossover_filtered)

# residual plot
danceability_lm_res1 <- ggplot(danceability_lm_aug, aes(x = energy, y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  labs(x = "Energy (std)", y = "residuals",
       title = "Residual Plot")

danceability_lm_res2 <- ggplot(danceability_lm_aug, aes(x = loudness, y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  labs(x = "Loudness (dB)", y = "residuals",
       title = "Residual Plot")

danceability_lm_res3 <- ggplot(danceability_lm_aug, aes(x = .fitted, y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  labs(x = "Predicted Danceability", y = "residuals",
       title = "Residual Plot")

# normal qq plot
danceability_lm_qq <- ggplot(danceability_lm_aug, aes(sample = .resid))+
  geom_qq() +
  geom_qq_line() +
  labs(y = "Sample Quantiles", x = "Normal Quantiles")

combined_plot <- (danceability_lm_res1 | danceability_lm_res2) /
  (danceability_lm_res3 | danceability_lm_qq)
res_qq <- combined_plot +
  plot_layout(guides = 'collect') +
  plot_annotation(title = "Residual Plot and Normal Q-Q Plot of Spotify MLR")
res_qq
```



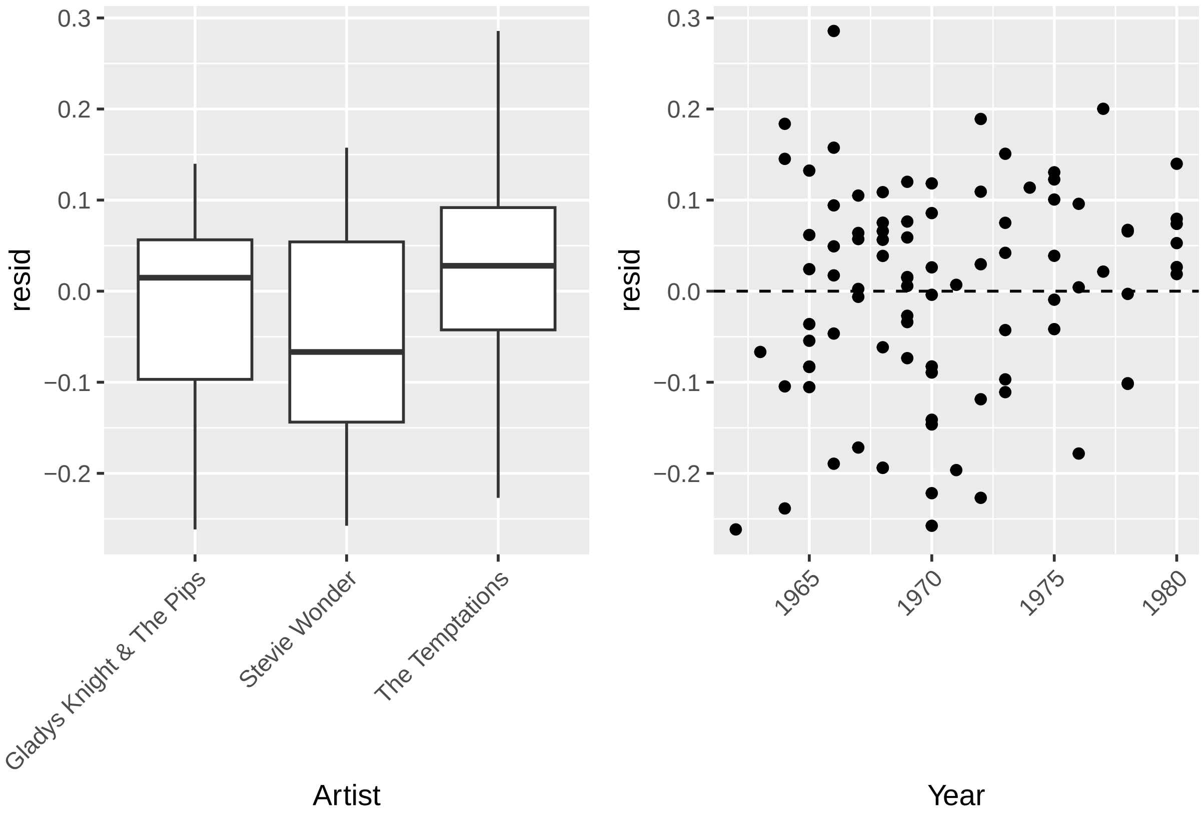
```
res_qq <- ggsave("res+qq.png", res_qq)
```

```
## Saving 6.5 x 4.5 in image
```

```
# Cluster independence
cluster_dep <- ggplot(danceability_lm_aug, aes(x = artist_name, y = .resid)) +
  geom_boxplot() +
  labs(y = "resid", x = "Artist") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

# Serial independence, w/ boxplot
ggplot(danceability_lm_aug, aes(x = as.factor(album_release_year), y = .resid)) +
  geom_boxplot() +
  labs(y = "resid", x = "Year") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```
# Serial independence, w/ scatterplot (better visuals)
serial_dep <- ggplot(danceability_lm_aug,
                     aes(x = album_release_year, y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  # geom_smooth(method = "lm", se = T) +
  labs(y = "resid", x = "Year") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

cluster_dep | serial_dep -> independence_check
independence_check
```



```
ggsave("independence.png", independence_check)
```

```
## Saving 6.5 x 4.5 in image
```

R^2

```
######## R^2 #########
danceability_lm_red <- lm(danceability ~ loudness + energy, data = spotify_data_crossover_filtered)

anova(danceability_lm_red, danceability_lm)
```

```
## Analysis of Variance Table
##
## Model 1: danceability ~ loudness + energy
## Model 2: danceability ~ loudness + energy * crossover_categ
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1     87 1.3796
## 2     85 1.1924  2   0.18722 6.6728 0.002033 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```