Visualizing Popularity Predictions: A Logistic Regression Approach with Custom Color Aesthetics

Avery Holloman

2024-08-07

```
# libraries
library(MASS)
library(pls)
```

```
##
## Attaching package: 'pls'
```

```
## The following object is masked from 'package:stats':
##
##     loadings
```

```
library(readxl)
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.0     ✓ stringr   1.5.1
## ✓ ggplot2   3.5.1     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.1
## ✓ purrr     1.0.2
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ✖ dplyr::select() masks MASS::select()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
e errors
```

```
# datset
data <-read_excel("C:/Users/jacob/Downloads/large_song_dataset.xlsx")
head(data)
```

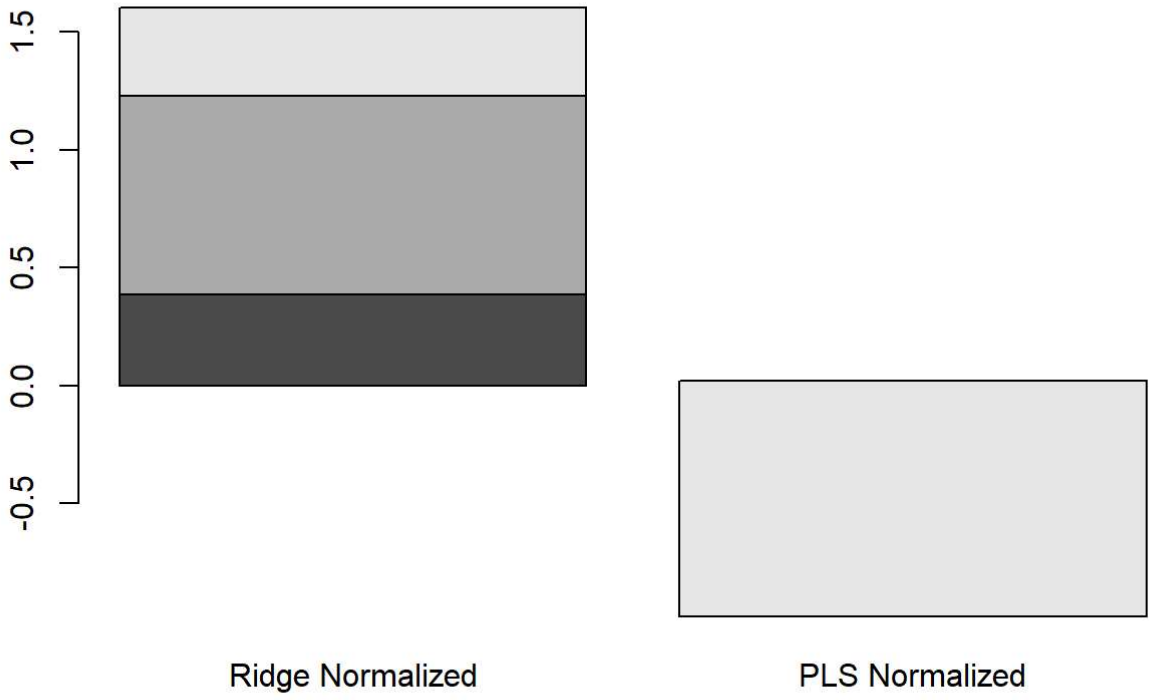
```
## # A tibble: 6 × 20
##   song_id                 bitrate review_comments songwriter created_on
##   <chr>                     <dbl> <chr>           <chr>      <dttm>
## 1 ae15dc06-c222-469f-b1c…     320 Then region hi… Charles J… 2021-05-11 15:23:48
## 2 d97d144b-b95a-4bee-b4b…     320 My partner how… Alexander… 2021-07-07 20:50:45
## 3 3932ef71-220e-4963-9e2…     256 Alone answer v… Kathleen … 2022-10-08 16:18:54
## 4 25bb3810-4d31-4da6-ae9…     320 Office easy do… Christoph… 2022-02-19 21:52:47
## 5 84588900-d720-4523-a6d…     128 Away PM attorn… Alexis Gr… 2020-11-12 13:07:12
## 6 957612e2-02a1-4d4e-b92…     320 Prepare exampl… Jordan Ch… 2020-12-03 12:16:23
## # ℹ 15 more variables: recorded_on <dttm>, song_duration <dbl>,
## #   favorites_count <dbl>, primary_genre <chr>, genre_list <chr>,
## #   song_info <chr>, popularity_index <dbl>, language <chr>,
## #   usage_license <chr>, play_count <dbl>, lyric_writer <chr>,
## #   track_number <dbl>, music_publisher <chr>, song_tags <chr>,
## #   song_title <chr>
```

```
#Data for Modeling

#My most relevant numerical columns
X <- as.matrix(data[, c("bitrate", "favorites_count", "play_count")])
y <- as.vector(data$popularity_index)

# Checking for missing values in my dataset
X <- na.omit(X)
y <- na.omit(y)
```

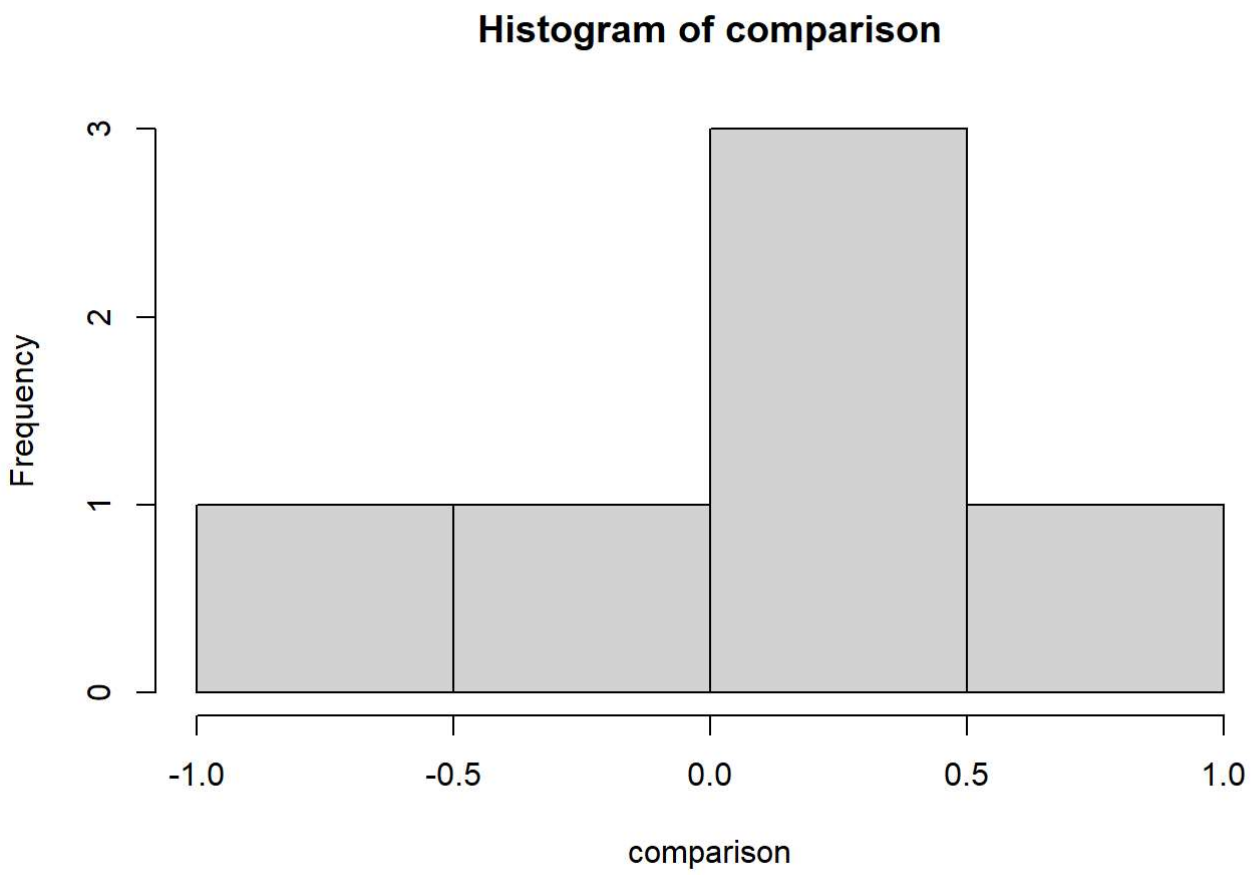
```
# Ridge Regression
lambda <- 1e10
beta_ridge <- solve(t(X) %*% X + lambda * diag(ncol(X))) %*% t(X) %*% y
beta_ridge_normalized <- beta_ridge / sqrt(sum(beta_ridge^2))

# Partial Least Squares (PLS)
pls_model <- plsr(y ~ X, ncomp = 1, validation = "none")
beta_pls <- coef(pls_model, ncomp = 1)
beta_pls_normalized <- beta_pls / sqrt(sum(beta_pls^2))

# Comparing the estimates
comparison <- cbind(beta_ridge_normalized, beta_pls_normalized)
colnames(comparison) <- c("Ridge Normalized", "PLS Normalized")

print(comparison)
```

```
##                 Ridge Normalized PLS Normalized
## bitrate                0.3861142   -0.003158576
## favorites_count        0.8430689    0.025057827
## play_count             0.3743670   -0.999681013
```

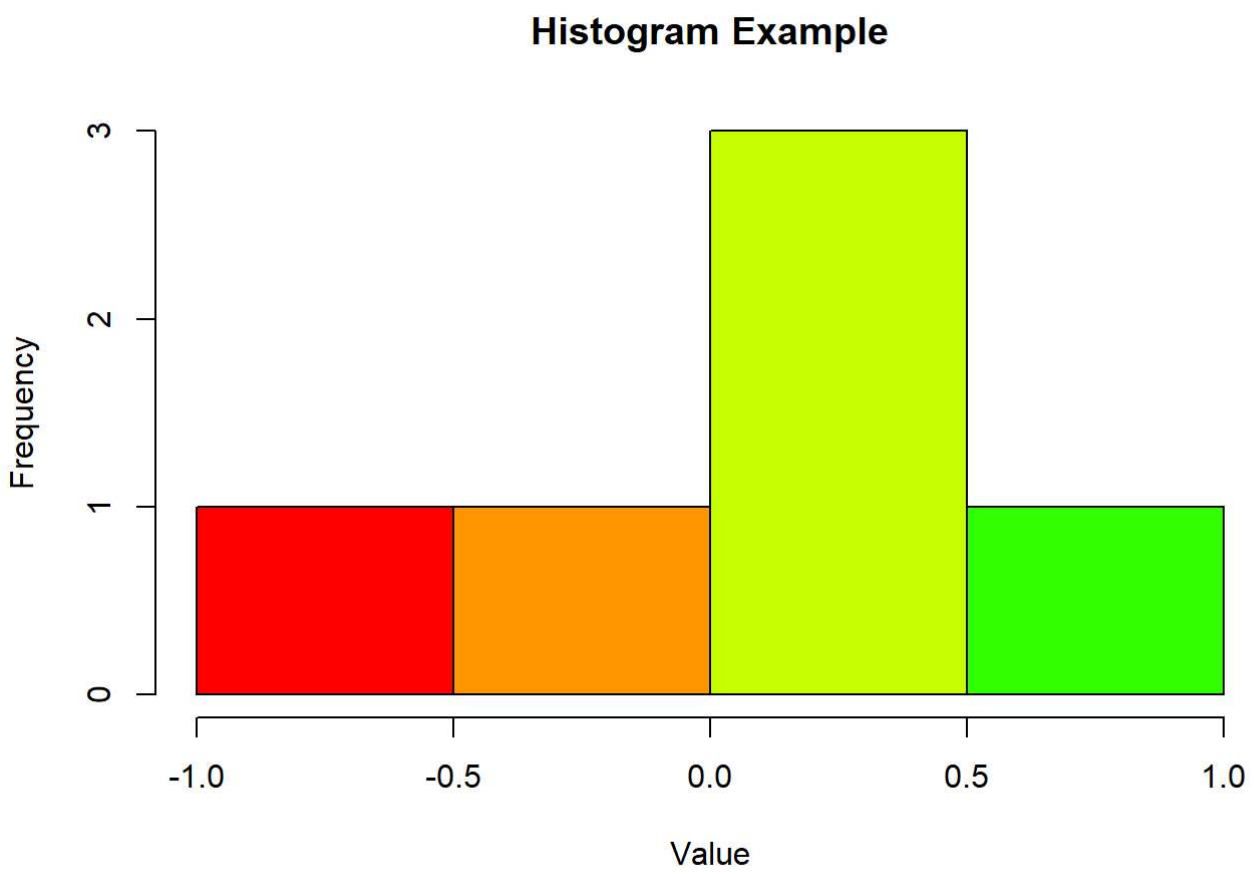
```
#bar plot to see the comparison of the results
barplot(comparison)
```



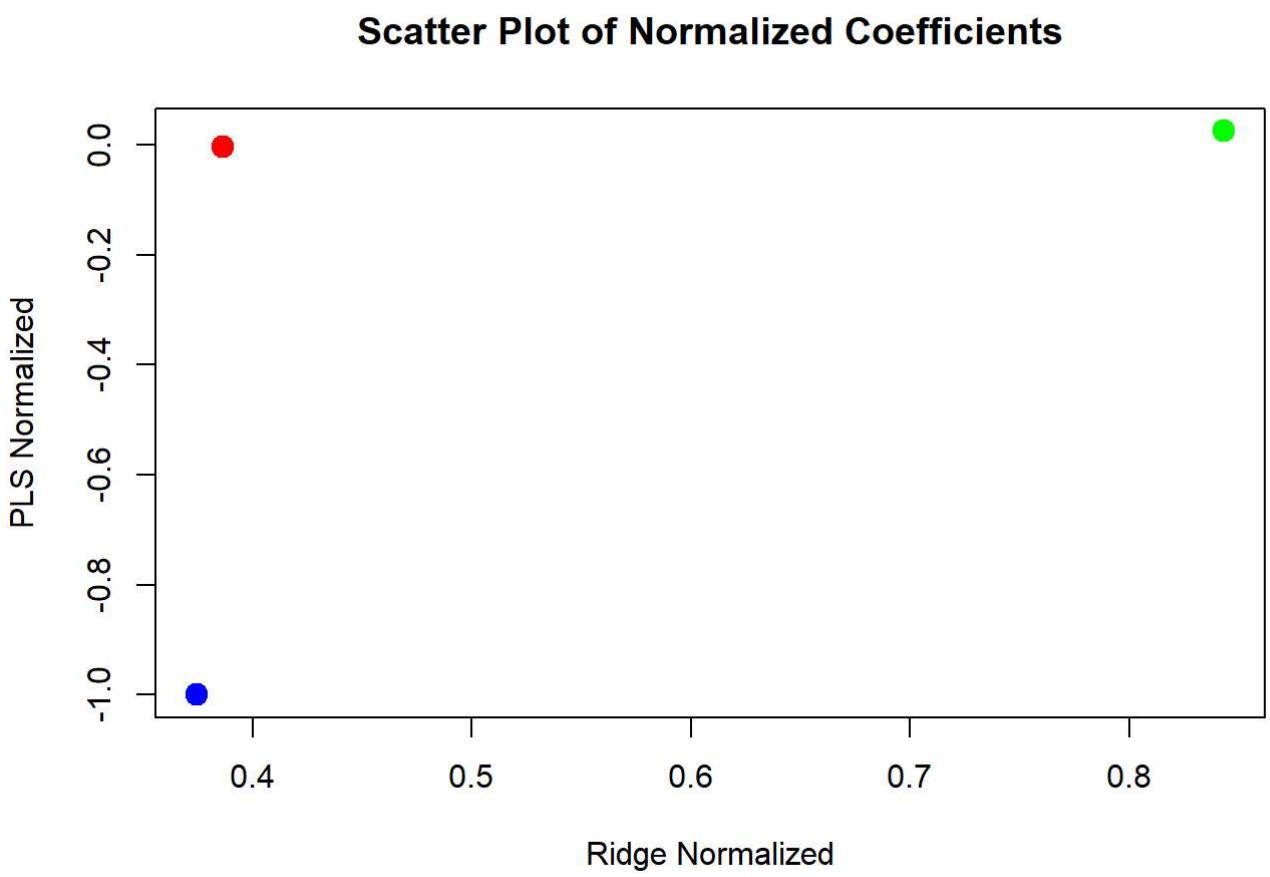

```
#Histogram to get a better understanding
hist(comparison)
```




```
#Simply adding color to get a better understanding of my results
hist(comparison,
                  main = "Histogram Example",
                  xlab = "Value",
                  ylab = "Frequency",
                  col = rainbow(10),    # Fill color using rainbow palette
                  border = "black")     # Border color
```




```
# Testing where on the graph are the points located with a scatter plot
#sample results
ridge_normalized <- c(0.3861142, 0.8430689, 0.3743670)
pls_normalized <- c(-0.003158576, 0.025057827, -0.999681013)

# Scatter plot
plot(ridge_normalized, pls_normalized,
     main = "Scatter Plot of Normalized Coefficients",
     xlab = "Ridge Normalized",
     ylab = "PLS Normalized",
     pch = 19,
     col = rainbow(length(ridge_normalized)),
     cex = 1.5)
```


Scatter Plot of Normalized Coefficients
PLS Normalized
0.0
-0.2
-0.4
-0.6
-0.8
-1.0
0.4
0.5
0.6
0.7
0.8
Ridge Normalized