Hypothesis Testing

2025-07-29

Adding the BC to data

```
bins <- c(0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5)

# Simple BC function
bimodality_coefficient_from_counts <- function(counts) {
  counts <- as.numeric(counts)
  W  <- sum(counts)
  mu <- sum(counts * bins) / W
  xc <- bins - mu
  m2 <- sum(counts * xc^2) / W
  m3 <- sum(counts * xc^3) / W
  m4 <- sum(counts * xc^4) / W
  g1 <- m3 / (m2^(3/2))         # skewness
  g2 <- (m4 / (m2^2)) - 3       # excess kurtosis
  kp <- g2 + 3                  # Pearson kurtosis
  (g1^2 + 1) / kp
}

# Apply to each row
data$BC <- apply(data[, c("X0.5","X1","X1.5","X2","X2.5","X3","X3.5","X4","X4.5")],
                 1, bimodality_coefficient_from_counts)
```

Experimental Hypothesis

Assumptions check

```
# define the model
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
data_experimental <- data %>%
  filter(experimental != "Unknown")
nrow(data_experimental) # 4407
```

```
## [1] 2948
```

```r
# models
model_experimental <- lm(polarization ~ experimental, data = data_experimental)
BC_experimental <- lm(BC ~ experimental, data = data_experimental)

# Homoscedasticity
library(lmtest)
bptest(model_experimental) # violated
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_experimental
## BP = 0.27421, df = 1, p-value = 0.6005
```

```r
bptest(BC_experimental)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  BC_experimental
## BP = 5.021, df = 1, p-value = 0.02504
```

```r
# Independence of Errors
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode

## The following object is masked from 'package:purrr':
##
##     some
```

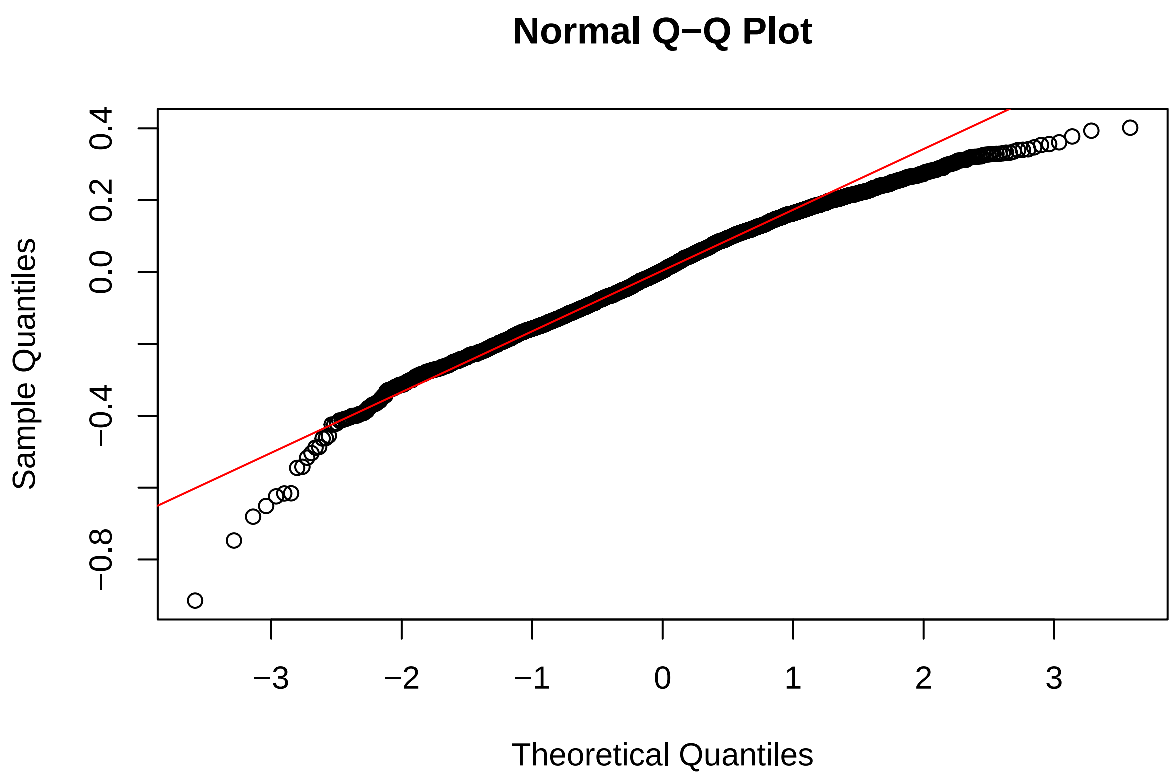
```r
dwtest(model_experimental) # passed
```

```
##
##  Durbin-Watson test
##
## data:  model_experimental
## DW = 1.9125, p-value = 0.008778
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
dwtest(BC_experimental)
```

```
##
##  Durbin-Watson test
##
## data:  BC_experimental
## DW = 1.9986, p-value = 0.485
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
# Normality of Residuals
qqnorm(residuals(model_experimental))
qqline(residuals(model_experimental), col = "red")
```




```r
shapiro.test(model_experimental$residuals) # not passed but sample is very large
```

```
##
##  Shapiro-Wilk normality test
##
## data:  model_experimental$residuals
## W = 0.98709, p-value = 9.745e-16
```

Results

```
library(sandwich)
coeftest(model_experimental, vcov = vcovHC(model_experimental, type = "HC1")) # robust standard error d
```

```
##
## t test of coefficients:
##
##                  Estimate Std. Error  t value Pr(>|t|)
## (Intercept)     1.8482097  0.0030676 602.4977   <2e-16 ***
## experimentalYes 0.0134445  0.0094204   1.4272   0.1536
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
confint.default(model_experimental, vcov. = robust_vcov)
```

```
##                         2.5 %     97.5 %
## (Intercept)      1.842181385 1.85423809
## experimentalYes -0.004683623 0.03157268
```

```
summary(model_experimental)$r.squared
```

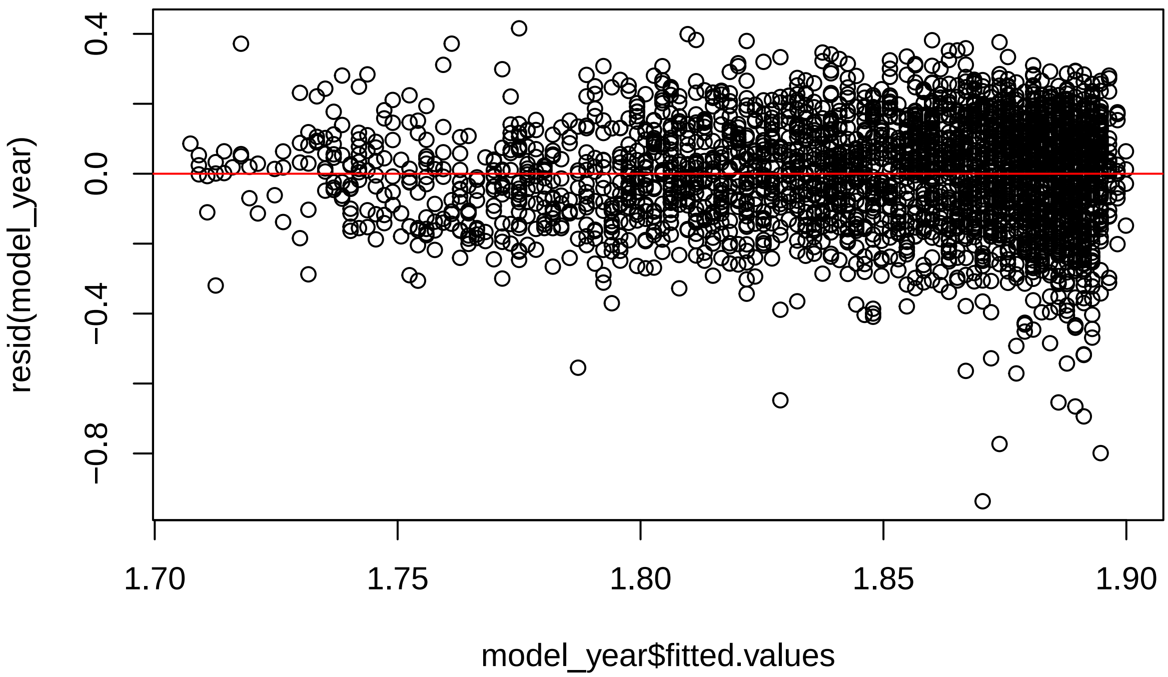
```
## [1] 0.0007166979
```

```
coeftest(BC_experimental, vcov = vcovHC(BC_experimental, type = "HC1"))
```

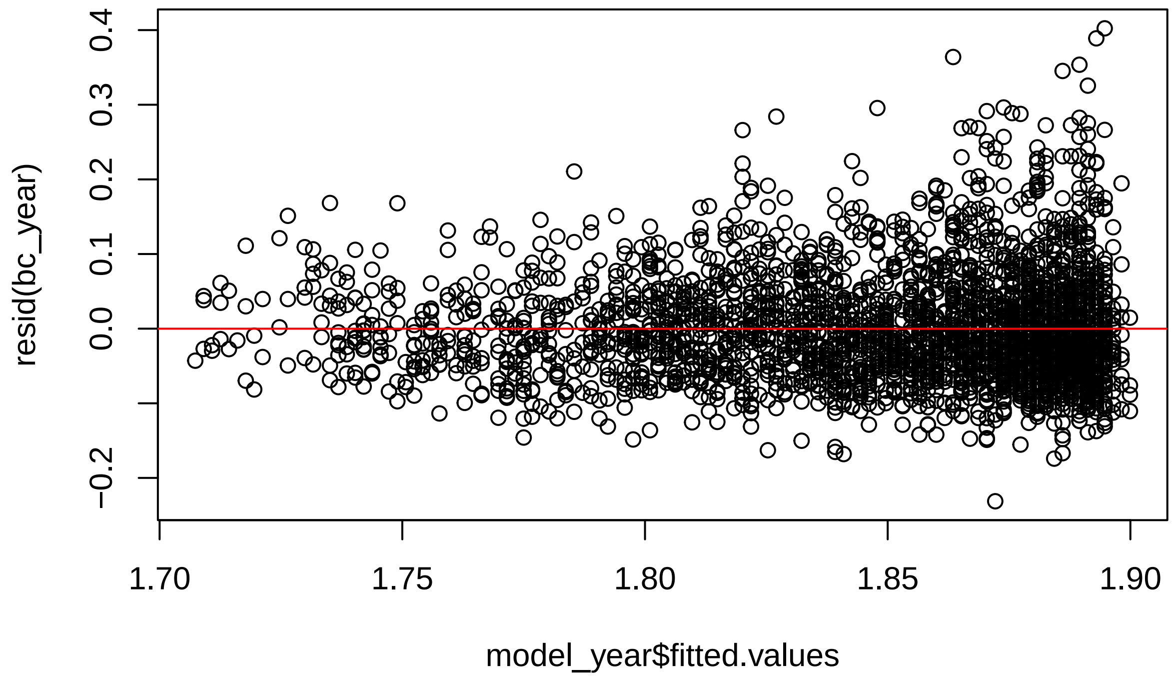
```
##
## t test of coefficients:
##
##                  Estimate Std. Error  t value  Pr(>|t|)
## (Intercept)     0.4326586  0.0015003 288.3871 < 2.2e-16 ***
## experimentalYes 0.0352279  0.0039356   8.9511 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Movie Age Hypothesis

```
model_year <- lm(polarization ~ release_year, data)
bc_year <- lm(BC ~ release_year, data = data)

#linearity
plot(model_year$fitted.values, resid(model_year))
abline(h = 0, col = "red")
```




```
plot(model_year$fitted.values, resid(bc_year))
abline(h = 0, col = "red")
```




```r
# Homoscedasticity
library(lmtest)
bptest(model_year) # passed
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_year
## BP = 18.064, df = 1, p-value = 2.136e-05
```

```r
bptest(bc_year)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  bc_year
## BP = 26.65, df = 1, p-value = 2.439e-07
```

```r
# Independence of Errors
library(car)
dwtest(model_year) # passed
```

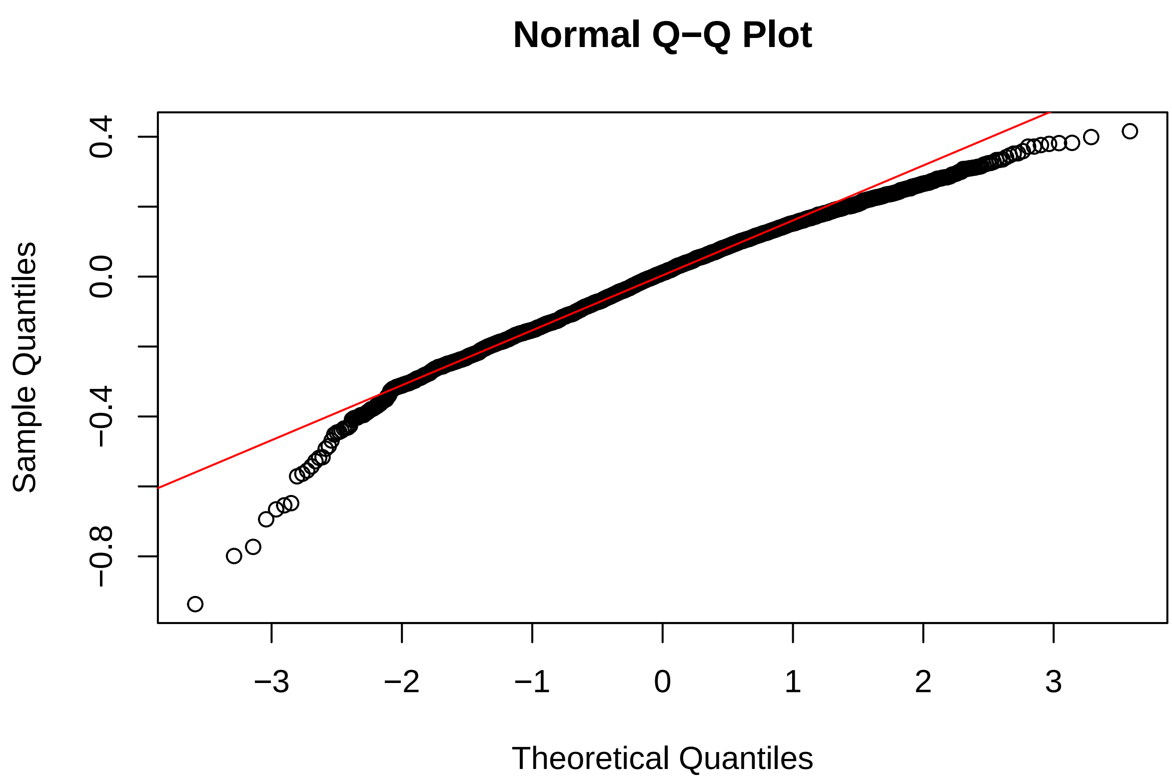
```
##
##  Durbin-Watson test
```

```
##
## data:  model_year
## DW = 1.9197, p-value = 0.01427
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
dwtest(bc_year)
```

```
##
##  Durbin-Watson test
##
## data:  bc_year
## DW = 2.0145, p-value = 0.6539
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
# Normality of Residuals
qqnorm(residuals(model_year))
qqline(residuals(model_year), col = "red")
```




```r
shapiro.test(model_year$residuals) # not passed but sample is very large
```

```
##
##  Shapiro-Wilk normality test
##
## data:  model_year$residuals
## W = 0.98281, p-value < 2.2e-16
```

Results

```r
coeftest(model_year, vcov = vcovHC(model_year, type = "HC1")) # robust standard error due to heterosced
```

```
##
## t test of coefficients:
##
##                 Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  -1.61314788  0.21660090 -7.4476 1.239e-13 ***
## release_year  0.00173485  0.00010872 15.9572 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
confint.default(model_year, vcov. = robust_vcov)
```

```
##                    2.5 %       97.5 %
## (Intercept)  -2.06806891 -1.158226861
## release_year  0.00150694  0.001962751
```

```r
summary(model_year)$r.squared
```

```
## [1] 0.06967767
```

```r
coeftest(bc_year, vcov = vcovHC(bc_year, type = "HC1"))
```

```
##
## t test of coefficients:
##
##                 Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  -1.4637e+00  1.0184e-01 -14.373 < 2.2e-16 ***
## release_year  9.5231e-04  5.1159e-05  18.614 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Moral Themes Hypothesis

Assumption check

```r
data_moral <- data %>%
  filter(!is.na(moral_themes))

model_moral <- lm(polarization ~ moral_themes, data = data_moral)
bc_moral <- lm(BC ~ moral_themes, data = data_moral)
nrow(data_moral) # 4481
```

```
## [1] 2966
```

```r
# Homoscedasticity
library(lmtest)
bptest(model_moral) # violated
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_moral
## BP = 7.1936, df = 1, p-value = 0.007316
```

```r
bptest(bc_moral)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  bc_moral
## BP = 5.7004, df = 1, p-value = 0.01696
```

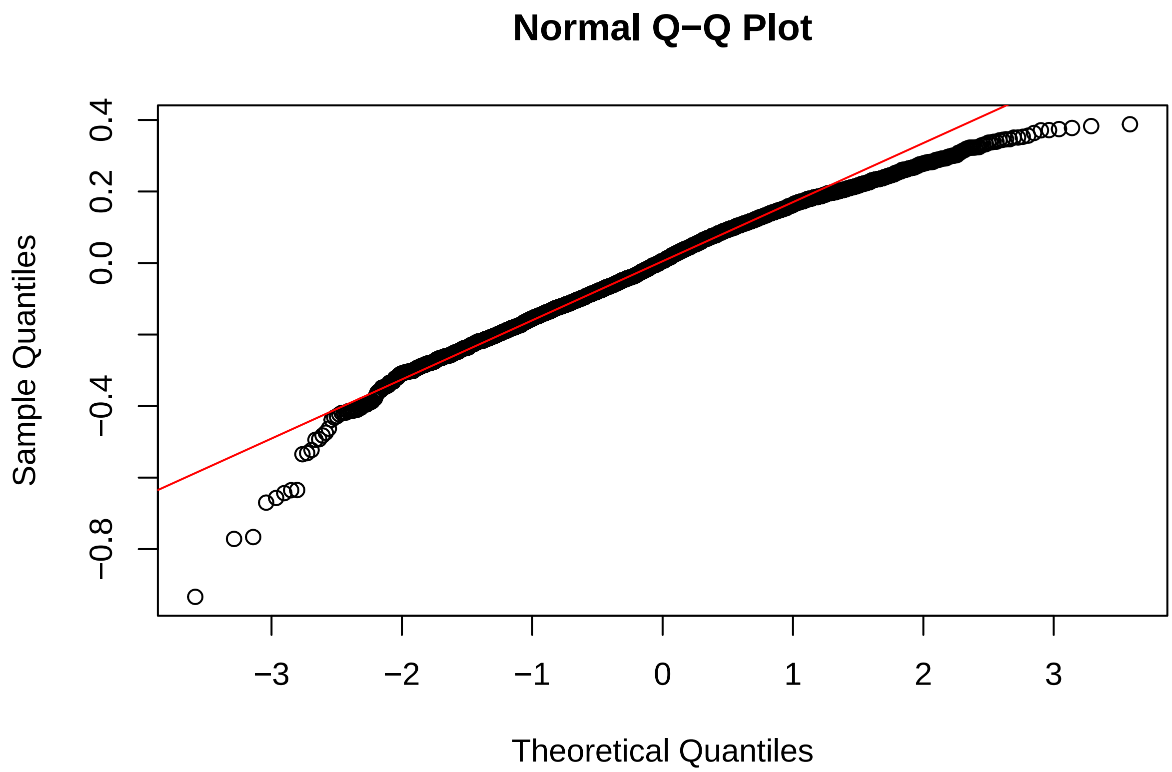
```r
# Independence of Errors
library(car)
dwtest(model_moral) # passed
```

```
##
##  Durbin-Watson test
##
## data:  model_moral
## DW = 1.9228, p-value = 0.01766
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
dwtest(bc_moral)
```

```
##
##  Durbin-Watson test
##
## data:  bc_moral
## DW = 1.9981, p-value = 0.4787
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
# Normality of Residuals
qqnorm(residuals(model_moral))
qqline(residuals(model_moral), col = "red")
```



Main resulst

```r
coeftest(model_moral, vcov = vcovHC(model_moral, type = "HC1"))
```

```
##
## t test of coefficients:
##
##                      Estimate Std. Error  t value  Pr(>|t|)
## (Intercept)         1.8672534  0.0047526 392.8899 < 2.2e-16 ***
## moral_themesTRUE -0.0294835  0.0059765  -4.9332 8.535e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
confint.default(model_moral, vcov. = robust_vcov)
```

```
##                        2.5 %     97.5 %
## (Intercept)       1.85836406  1.8761428
## moral_themesTRUE -0.04102004 -0.0179469
```

```r
summary(model_moral)$r.squared
```

```
## [1] 0.00839384
```

```r
coeftest(bc_moral, vcov = vcovHC(bc_moral, type = "HC1"))
```

```
##
## t test of coefficients:
##
##                    Estimate Std. Error  t value Pr(>|t|)
## (Intercept)       0.4404685  0.0023345 188.6776  < 2e-16 ***
## moral_themesTRUE -0.0058298  0.0029253  -1.9928  0.04637 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Marginalized Character Hypothesis

Assuumption check

```r
data_marginalized <- data %>%
  filter(
    !is.na(white_protagonist),
    !is.na(lgbtq_protagonist),
    !is.na(female_protagonist)
    )
nrow(data_marginalized) # 1446
```

```
## [1] 1109
```

```r
data_marginalized$white_protagonist <- relevel(data_marginalized$white_protagonist, ref = "No")

combined_model <- lm(polarization ~ white_protagonist + lgbtq_protagonist + female_protagonist, data = d
BC_combined <- lm(BC ~ white_protagonist + lgbtq_protagonist + female_protagonist, data = data_margina

# Homoscedasticity
library(lmtest)
bptest(combined_model) # passed
```

```
##
##  studentized Breusch-Pagan test
##
## data:  combined_model
## BP = 4.3646, df = 3, p-value = 0.2247
```

```r
bptest(BC_combined)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  BC_combined
## BP = 8.556, df = 3, p-value = 0.03582
```

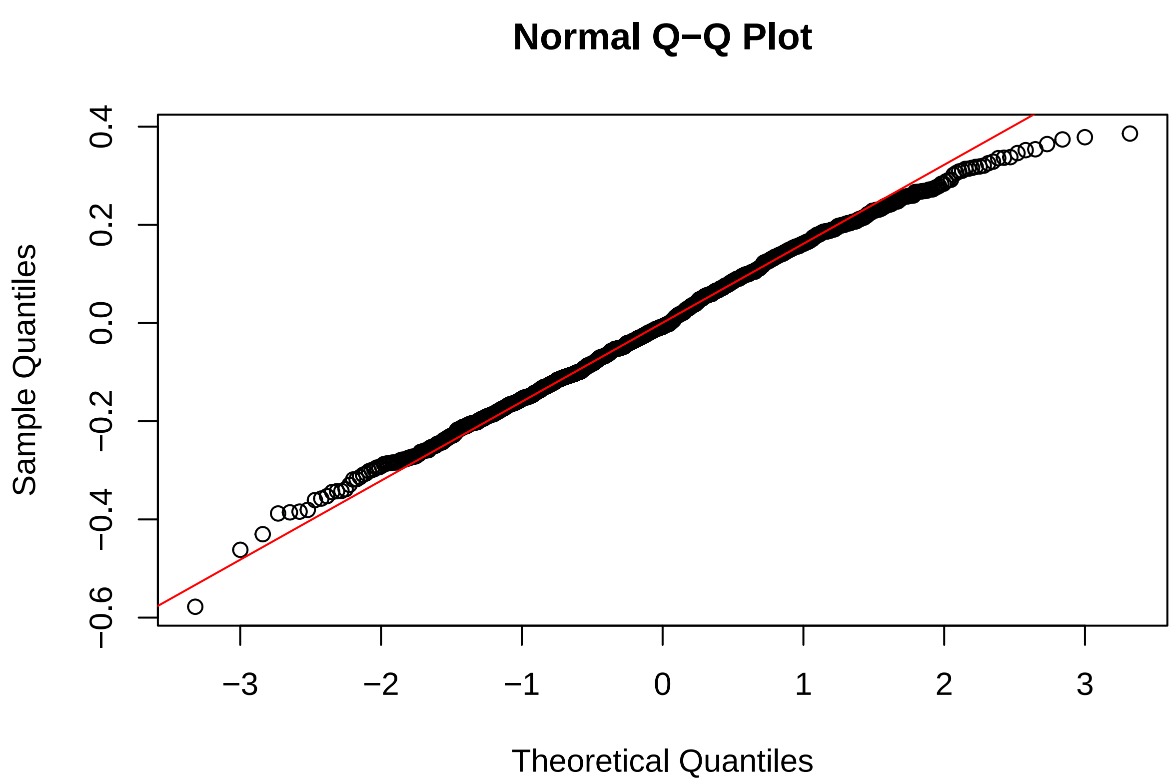
```r
# Independence of Errors
library(car)
dwtest(combined_model) # passed
```

```
##
##  Durbin-Watson test
##
## data:  combined_model
## DW = 2.0213, p-value = 0.6392
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
dwtest(BC_combined)
```

```
##
##  Durbin-Watson test
##
## data:  BC_combined
## DW = 2.0884, p-value = 0.9299
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
# Normality of Residuals
qqnorm(residuals(combined_model))
qqline(residuals(combined_model), col = "red")
```



```r
shapiro.test(combined_model$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  combined_model$residuals
## W = 0.9961, p-value = 0.006662
```

Main results

```r
summary(combined_model)
```

```
##
## Call:
## lm(formula = polarization ~ white_protagonist + lgbtq_protagonist +
##     female_protagonist, data = data_marginalized)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5778 -0.1079 -0.0077  0.1091  0.3859
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.809802   0.009061 199.739  < 2e-16 ***
## white_protagonistYes  0.013122   0.009963   1.317  0.18809
## lgbtq_protagonistYes  0.048893   0.016146   3.028  0.00252 **
## female_protagonistYes 0.037925   0.009401   4.034 5.85e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1526 on 1105 degrees of freedom
## Multiple R-squared:  0.02669,    Adjusted R-squared:  0.02405
## F-statistic:  10.1 on 3 and 1105 DF,  p-value: 1.444e-06
```

```r
confint.default(combined_model, vcov. = robust_vcov)
```

```
##                              2.5 %     97.5 %
## (Intercept)            1.792043271 1.82756101
## white_protagonistYes  -0.006405272 0.03264876
## lgbtq_protagonistYes   0.017247212 0.08053927
## female_protagonistYes  0.019499696 0.05634934
```

```r
summary(combined_model)$r.squared
```

```
## [1] 0.02669004
```

```r
coeftest(BC_combined, vcov = vcovHC(BC_combined, type = "HC1"))
```

```
##
## t test of coefficients:
##
##                          Estimate  Std. Error  t value  Pr(>|t|)
## (Intercept)            0.43716372  0.00429724 101.7312 < 2.2e-16 ***
## white_protagonistYes  -0.01708138  0.00453879  -3.7634 0.0001764 ***
## lgbtq_protagonistYes   0.02472108  0.00592205   4.1744 3.222e-05 ***
## female_protagonistYes -0.00037449  0.00422285  -0.0887 0.9293515
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(BC_combined)$r.squared
```

```
## [1] 0.02092886
```

Descriptives

Marginalized characters hypothesis

```
library(tidyverse)
library(psych)
```

```
##
## Attaching package: 'psych'

## The following object is masked from 'package:car':
##
##     logit

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
cat("Descriptive Statistics for Marginalized Characters Dataset\n")
```

```
## Descriptive Statistics for Marginalized Characters Dataset
```

```
numeric_vars <- data_marginalized |> select(where(is.numeric))
print(round(describe(numeric_vars), 2))
```

```
##                 vars    n     mean        sd  median trimmed     mad     min
## release_year       1 1109  1990.88     26.61 1999.00 1994.00   25.20 1914.00
## runtime            2 1109    98.69     28.32   96.00   98.25   19.27   20.00
## watched            3 1109 44901.20 253792.99 1026.00 3889.63 1292.83   71.00
## avg_rating         4  810     3.25      0.44    3.30    3.26    0.30    1.40
## X0.5               5 1109   267.40   1447.33    6.00   25.72    8.90    0.00
## X1                 6 1109   520.63   2544.82   14.00   54.88   19.27    0.00
## X1.5               7 1109   462.14   2161.72   15.00   55.29   20.76    0.00
```

```
## X2                8 1109  1610.47   7848.06   51.00  173.98   65.23    0.00
## X2.5              9 1109  1763.59   8249.88   70.00  209.52   87.47    1.00
## X3               10 1109  4881.06  24023.78  156.00  498.42  197.19    2.00
## X3.5             11 1109  4249.92  20351.78   98.00  398.26  128.99    0.00
## X4               12 1109  6845.92  44855.44   83.00  425.66  111.19    0.00
## X4.5             13 1109  2760.25  23810.85   16.00  102.86   22.24    0.00
## X5               14 1109  5157.28  55173.39   29.00  155.99   40.03    0.00
## fans             15  634   904.61   8277.99    5.00   40.28    5.93    1.00
## liked            16 1109 11518.19  92908.59  155.00  681.06  197.19    2.00
## total_ratings    17 1109 28518.66 169127.03  626.00 2412.92  791.71   50.00
## polarization     18 1109     1.84      0.15    1.84    1.84    0.17    1.23
## BC               19 1109     0.43      0.07    0.42    0.42    0.06    0.25
##                        max      range  skew kurtosis      se
## release_year       2025.00     111.00 -0.85    -0.24    0.80
## runtime             220.00     200.00  0.18     1.13    0.85
## watched         5483399.00 5483328.00 12.97   223.36 7621.03
## avg_rating            4.40       3.00 -0.56     1.27    0.02
## X0.5              28899.00   28899.00 11.34   173.15   43.46
## X1                41818.00   41818.00  9.06   104.51   76.42
## X1.5              27665.00   27665.00  8.42    85.08   64.91
## X2               128547.00  128547.00  9.32   110.52  235.67
## X2.5             130683.00  130682.00  9.29   111.46  247.73
## X3               397434.00  397432.00  9.96   125.50  721.40
## X3.5             290787.00  290787.00  8.88    95.63  611.13
## X4              1094011.00 1094011.00 15.78   330.68 1346.94
## X4.5             628608.00  628608.00 19.36   457.44  715.00
## X5              1517890.00 1517890.00 21.52   538.79 1656.78
## fans             171000.00  170999.00 16.03   296.58  328.76
## liked           2512925.00 2512923.00 19.93   490.46 2789.91
## total_ratings   3915331.00 3915281.00 14.32   275.51 5078.64
## polarization          2.23       0.99 -0.08    -0.37    0.00
## BC                    0.85       0.60  1.22     3.69    0.00
```

```
cat("\n")
```

```
cat("Categorical Frequencies for Marginalized Characters Dataset\n")
```

```
## Categorical Frequencies for Marginalized Characters Dataset
```

```
categorical_vars <- data_marginalized |> select(where(is.factor))
for (var in names(categorical_vars)) {
  cat("Variable:", var, "\n")
  print(table(categorical_vars[[var]]))
  cat("\n")
}
```

```
## Variable: female_protagonist
##
##  No Yes
## 666 443
##
## Variable: lgbtq_protagonist
```

```
##
##   No  Yes
## 1010   99
##
## Variable: white_protagonist
##
##  No Yes
## 338 771
##
## Variable: experimental
##
##   No  Yes
## 1016   93
##
## Variable: moral_themes
##
## FALSE  TRUE
##   390   718
```

Movie age hypothesis

```r
library(psych)

cat("Descriptive Statistics for Full Dataset\n")
```

```
## Descriptive Statistics for Full Dataset
```

```r
numeric_vars <- data |> select(where(is.numeric))
print(round(describe(numeric_vars), 2))
```

```
##                vars    n     mean        sd  median trimmed    mad     min
## release_year      1 2974  1995.95     24.08 2004.00 1999.38  20.76 1914.00
## runtime           2 2974    95.02     31.12   93.00   94.36  16.31   20.00
## watched           3 2974 27615.67 189665.15  624.50 1858.69 740.56   67.00
## avg_rating        4 1912     3.21      0.44    3.30    3.23   0.44    1.40
## X0.5              5 2974   191.10   1362.62    5.00   16.29   7.41    0.00
## X1                6 2974   397.34   2664.10   12.00   32.67  16.31    0.00
## X1.5              7 2974   368.57   2395.08   12.00   33.04  16.31    0.00
## X2                8 2974  1191.14   7482.98   36.00   94.23  43.00    0.00
## X2.5              9 2974  1260.79   7488.12   41.00  111.84  50.41    0.00
## X3               10 2974  3266.20  21237.90   86.50  253.08 104.52    0.00
## X3.5             11 2974  2710.33  17283.09   56.00  191.34  72.65    0.00
## X4               12 2974  4028.79  32717.84   46.00  181.38  60.79    0.00
## X4.5             13 2974  1495.46  15802.81    9.00   40.26  13.34    0.00
## X5               14 2974  2695.99  35299.01   16.00   63.89  20.76    0.00
## fans             15 1459   511.28   5572.80    4.00   18.82   4.45    1.00
## liked            16 2974  6607.11  62809.28   90.00  307.29 111.19    0.00
## total_ratings    17 2974 17605.70 124815.42  388.50 1175.03 464.05   50.00
## polarization     18 2974     1.85      0.16    1.85    1.85   0.17    0.93
## BC               19 2974     0.44      0.08    0.43    0.43   0.06    0.22
##                      max      range   skew kurtosis      se
```

```
## release_year     2025.00      111.00 -1.09     0.47    0.44
## runtime           625.00      605.00  2.75    35.95    0.57
## watched       5483399.00 5483332.00 15.36   320.10 3477.90
## avg_rating          4.40        3.00 -0.48     0.71    0.01
## X0.5            42633.00   42633.00 17.58   424.46   24.99
## X1              76627.00   76627.00 14.90   307.19   48.85
## X1.5            71292.00   71292.00 15.30   333.73   43.92
## X2             161369.00  161369.00 11.95   175.79  137.22
## X2.5           130683.00  130683.00 11.33   153.73  137.31
## X3             452955.00  452955.00 13.19   212.58  389.44
## X3.5           352378.00  352378.00 12.41   187.98  316.92
## X4            1094011.00 1094011.00 18.89   492.79  599.95
## X4.5           628608.00  628608.00 26.14   901.23  289.78
## X5            1517890.00 1517890.00 31.39  1215.49  647.28
## fans           171000.00  170999.00 23.13   637.53  145.90
## liked         2512925.00 2512925.00 25.88   903.49 1151.74
## total_ratings 3915331.00 3915281.00 16.71   393.44 2288.75
## polarization        2.25        1.32 -0.42     0.53    0.00
## BC                  0.86        0.65  1.08     2.31    0.00
```

```r
cat("\n")
```

Moral themes

```r
cat("Descriptive Statistics for Moral Themes Dataset\n")
```

```
## Descriptive Statistics for Moral Themes Dataset
```

```r
numeric_vars <- data_moral |> select(where(is.numeric))
print(round(describe(numeric_vars), 2))
```

```
##               vars    n     mean        sd  median trimmed    mad     min
## release_year     1 2966  1995.93     24.09 2004.00 1999.35  20.76 1914.00
## runtime          2 2966    95.12     31.07   93.50   94.43  17.05   20.00
## watched          3 2966 27689.67 189915.49  626.00 1870.56 742.78   67.00
## avg_rating       4 1911     3.21      0.44    3.30    3.23   0.44    1.40
## X0.5             5 2966   191.61   1364.42    5.00   16.39   7.41    0.00
## X1               6 2966   398.40   2667.62   12.00   32.87  16.31    0.00
## X1.5             7 2966   369.56   2398.23   12.00   33.24  16.31    0.00
## X2               8 2966  1194.33   7492.81   36.00   94.78  43.00    0.00
## X2.5             9 2966  1264.17   7497.94   41.00  112.48  50.41    0.00
## X3              10 2966  3274.94  21265.86   87.00  254.57 105.26    0.00
## X3.5            11 2966  2717.59  17305.82   56.00  192.59  72.65    0.00
## X4              12 2966  4039.59  32761.29   46.00  182.66  60.79    0.00
## X4.5            13 2966  1499.48  15823.92    9.00   40.55  13.34    0.00
## X5              14 2966  2703.22  35346.33   16.00   64.33  20.76    0.00
## fans            15 1457   511.98   5576.59    4.00   18.85   4.45    1.00
## liked           16 2966  6624.82  62893.03   91.00  309.31 112.68    0.00
## total_ratings   17 2966 17652.88 124980.38  391.50 1182.68 467.76   50.00
## polarization    18 2966     1.85      0.16    1.85    1.85   0.17    0.93
```

```
## BC              19 2966     0.44      0.08    0.43    0.43   0.06    0.22
##                       max       range  skew kurtosis      se
## release_year      2025.00      111.00 -1.09     0.47    0.44
## runtime            625.00      605.00  2.78    36.25    0.57
## watched       5483399.00 5483332.00 15.34   319.24 3487.18
## avg_rating           4.40        3.00 -0.48     0.71    0.01
## X0.5             42633.00   42633.00 17.55   423.32   25.05
## X1               76627.00   76627.00 14.88   306.37   48.98
## X1.5             71292.00   71292.00 15.28   332.84   44.04
## X2              161369.00  161369.00 11.94   175.31  137.58
## X2.5            130683.00  130683.00 11.31   153.31  137.68
## X3              452955.00  452955.00 13.17   212.00  390.48
## X3.5            352378.00  352378.00 12.40   187.47  317.77
## X4             1094011.00 1094011.00 18.87   491.47  601.56
## X4.5            628608.00  628608.00 26.11   898.81  290.56
## X5             1517890.00 1517890.00 31.35  1212.22  649.02
## fans            171000.00  170999.00 23.11   636.66  146.10
## liked          2512925.00 2512925.00 25.85   901.07 1154.83
## total_ratings 3915331.00 3915281.00 16.69   392.39 2294.86
## polarization         2.25        1.32 -0.42     0.54    0.00
## BC                   0.86        0.65  1.09     2.32    0.00
```

```r
cat("\n")
```

```r
cat("Categorical Frequencies for Moral Themes Dataset\n")
```

```
## Categorical Frequencies for Moral Themes Dataset
```

```r
categorical_vars <- data_moral |> select(where(is.factor))
for (var in names(categorical_vars)) {
  cat("Variable:", var, "\n")
  print(table(categorical_vars[[var]]))
  cat("\n")
}
```

```
## Variable: female_protagonist
##
##   No  Yes
## 1731 1153
##
## Variable: lgbtq_protagonist
##
##   No  Yes
## 1129  132
##
## Variable: white_protagonist
##
##   No  Yes
##  744 1825
##
## Variable: experimental
##
```

```
##   No  Yes
## 2621  326
##
## Variable: moral_themes
##
## FALSE  TRUE
##  1205  1761
```

Experimental movies

```r
cat("Descriptive Statistics for Experimental Dataset\n")
```

```
## Descriptive Statistics for Experimental Dataset
```

```r
numeric_vars <- data_experimental |> select(where(is.numeric))
print(round(describe(numeric_vars), 2))
```

```
##               vars    n     mean        sd  median trimmed    mad     min
## release_year     1 2948  1995.85     24.12 2004.00 1999.26  20.76 1914.00
## runtime          2 2948    95.24     31.04   94.00   94.52  16.31   20.00
## watched          3 2948 27856.80 190482.51  632.50 1894.98 750.94   67.00
## avg_rating       4 1907     3.21      0.44    3.30    3.23   0.44    1.40
## X0.5             5 2948   192.69   1368.51    5.00   16.49   7.41    0.00
## X1               6 2948   400.73   2675.58   12.00   33.17  16.31    0.00
## X1.5             7 2948   371.75   2405.38   12.50   33.59  16.31    0.00
## X2               8 2948  1201.50   7515.10   36.00   95.92  43.00    0.00
## X2.5             9 2948  1271.78   7520.16   41.50  113.81  51.15    0.00
## X3              10 2948  3294.73  21329.19   88.00  257.68 106.75    0.00
## X3.5            11 2948  2734.04  17357.30   57.50  195.24  73.39    0.00
## X4              12 2948  4064.10  32859.68   46.00  185.39  60.79    0.00
## X4.5            13 2948  1508.60  15871.74    9.00   41.18  13.34    0.00
## X5              14 2948  2719.67  35453.48   16.00   65.34  20.76    0.00
## fans            15 1453   513.38   5584.20    4.00   18.91   4.45    1.00
## liked           16 2948  6665.01  63082.70   92.00  313.61 114.16    0.00
## total_ratings   17 2948 17759.59 125354.00  394.00 1198.65 472.21   50.00
## polarization    18 2948     1.85      0.16    1.85    1.85   0.17    0.93
## BC              19 2948     0.44      0.08    0.43    0.43   0.06    0.22
##                      max      range  skew kurtosis      se
## release_year     2025.00     111.00 -1.09     0.45    0.44
## runtime           625.00     605.00  2.80    36.55    0.57
## watched       5483399.00 5483332.00 15.29   317.31 3508.26
## avg_rating          4.40       3.00 -0.48     0.71    0.01
## X0.5            42633.00   42633.00 17.50   420.77   25.20
## X1              76627.00   76627.00 14.84   304.52   49.28
## X1.5            71292.00   71292.00 15.24   330.84   44.30
## X2             161369.00  161369.00 11.90   174.24  138.41
## X2.5           130683.00  130683.00 11.28   152.37  138.50
## X3             452955.00  452955.00 13.13   210.71  392.84
## X3.5           352378.00  352378.00 12.36   186.32  319.68
## X4            1094011.00 1094011.00 18.81   488.50  605.20
## X4.5           628608.00  628608.00 26.03   893.37  292.32
```

```
## X5             1517890.00 1517890.00 31.25  1204.87   652.97
## fans            171000.00  170999.00 23.08   634.91   146.50
## liked          2512925.00 2512925.00 25.77   895.64 1161.84
## total_ratings 3915331.00 3915281.00 16.64   390.03 2308.74
## polarization        2.25       1.32 -0.39     0.42     0.00
## BC                  0.85       0.63  1.06     2.16     0.00
```

```
cat("\n")
```

```
cat("Categorical Frequencies for Experimental Dataset\n")
```

```
## Categorical Frequencies for Experimental Dataset
```

```
categorical_vars <- data_experimental |> select(where(is.factor))
for (var in names(categorical_vars)) {
  cat("Variable:", var, "\n")
  print(table(categorical_vars[[var]]))
  cat("\n")
}
```

```
## Variable: female_protagonist
##
##   No  Yes
## 1724 1145
##
## Variable: lgbtq_protagonist
##
##   No  Yes
## 1129  132
##
## Variable: white_protagonist
##
##   No  Yes
##  739 1821
##
## Variable: experimental
##
##   No  Yes
## 2622  326
##
## Variable: moral_themes
##
## FALSE  TRUE
##  1199  1748
```

```
library(tidyverse)
library(patchwork)

moral <- tibble(
  predictor = "Moral theme",
  outcome   = c("Entropy","BC"),
  b         = c(-0.029, -0.006),
```

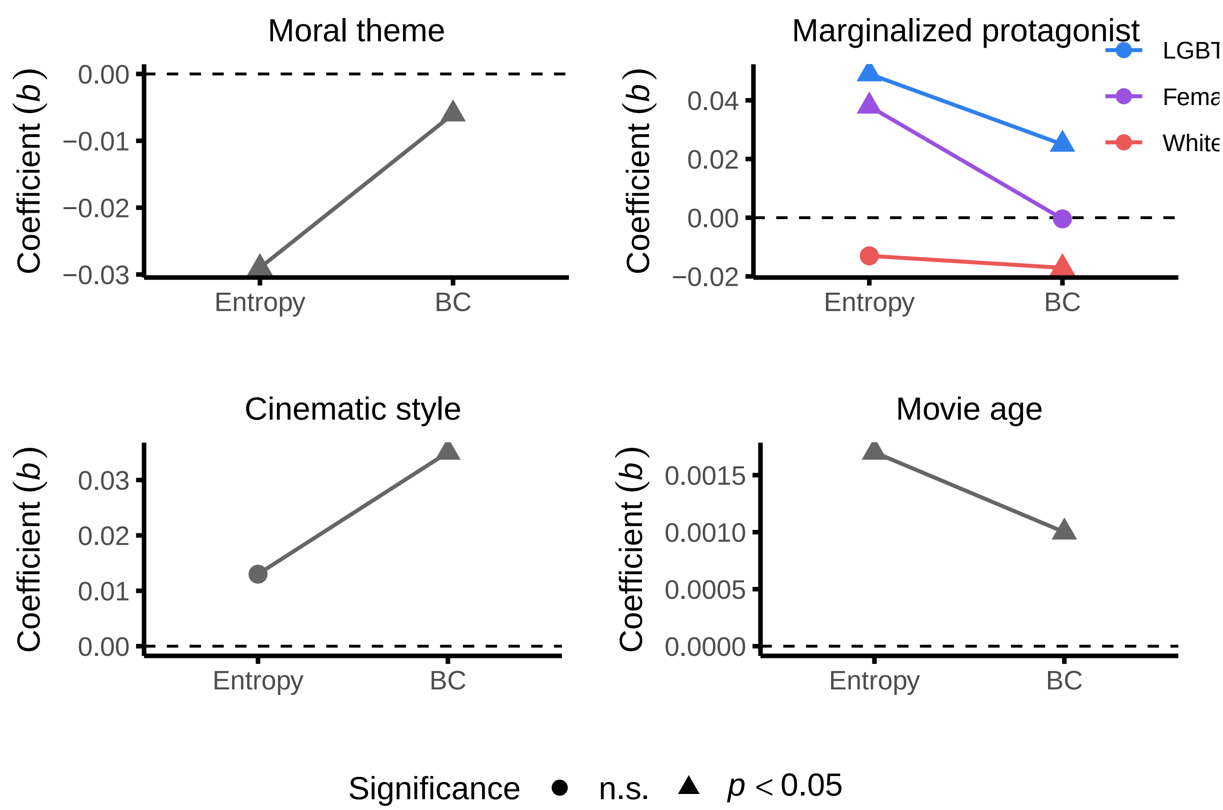
```
  p         = c(0.0001, 0.046)
)

style <- tibble(
  predictor = "Cinematic style",
  outcome   = c("Entropy","BC"),
  b         = c(0.013, 0.035),
  p         = c(0.15,  0.0009)
)

age <- tibble(
  predictor = "Movie age",
  outcome   = c("Entropy","BC"),
  b         = c(0.0017, 0.0010),
  p         = c(0.0001, 0.0001)
)

WHITE_BC_B <- -0.017
marginalized <- tibble(
  predictor = rep(c("LGBTQ+","Female","White"), each = 2),
  outcome   = rep(c("Entropy","BC"), times = 3),
  b         = c(0.049, 0.025,   0.038, -0.0004,   -0.013, -0.01708),
  p         = c(0.003, 0.0009,  0.0009, 0.929,     0.188, 0.0009)
)

# Colors
marg_levels <- c("LGBTQ+","Female","White")
cols <- c(
  "LGBTQ+" = "#2F80ED",
  "Female" = "#9B51E0",
  "White"  = "#EB5757",
  "Moral theme"    = "grey40",
  "Cinematic style" = "grey40",
  "Movie age"      = "grey40"
)

hypo_plot <- function(data, title, yzero = 0) {
  data <- data %>%
    mutate(
      outcome = factor(outcome, levels = c("Entropy","BC")),
      sig = factor(ifelse(p < 0.05, "p < .05", "n.s."),
                   levels = c("n.s.","p < .05"))
    )

  ggplot(data, aes(outcome, b, group = predictor)) +
    geom_hline(yintercept = yzero, linetype = "dashed") +
    geom_line(aes(color = predictor), linewidth = 0.7) +
    geom_point(aes(color = predictor, shape = sig), size = 3) +
    scale_color_manual(values = cols, breaks = marg_levels, drop = FALSE) +
    scale_shape_manual(values = c("n.s." = 16, "p < .05" = 17), drop = FALSE) +
    labs(title = title, x = NULL, y = expression(Coefficient~(italic(b)))) +
    theme_classic(base_size = 12) +
    theme(
```

```
      plot.title = element_text(face = "plain", hjust = .5, size = 12), # not bold
      panel.grid = element_blank(),
      axis.line = element_line(linewidth = 0.8, color = "black"),
      axis.ticks = element_line(linewidth = 0.8, color = "black"),
      legend.position = "none"
    )
}

p_moral <- hypo_plot(moral, "Moral theme")
p_style <- hypo_plot(style, "Cinematic style")
p_age   <- hypo_plot(age,   "Movie age")

p_marg <- hypo_plot(marginalized, "Marginalized protagonist") +
  scale_color_manual(
    values = cols, breaks = marg_levels, name = NULL, drop = FALSE
  ) +
  guides(
    shape = "none",
    color = guide_legend(
      title = NULL,
      ncol = 1,
      override.aes = list(linetype = 1, shape = 16, size = 2.5)
    )
  ) +
  theme(
    legend.position = c(0.78, 0.85),
    legend.justification = c(0, 0.5),
    legend.background = element_rect(fill = "white", color = NA),
    legend.text = element_text(size = 8.5)
  )
```

```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.

## Warning: A numeric `legend.position` argument in `theme()` was deprecated in ggplot2
## 3.5.0.
## i Please use the `legend.position.inside` argument of `theme()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
legend_sig <- cowplot::get_legend(
  ggplot(tibble(x=1, y=1, sig=c("n.s.","p < .05")),
         aes(x, y, shape = sig)) +
    geom_point(size = 2.5) +
    scale_shape_manual(
      name = "Significance",
      values = c("n.s." = 16, "p < .05" = 17),
      labels = c("n.s.", expression(italic(p) < .05))
    ) +
    theme_void(base_size = 12) +
    theme(
      legend.position = "bottom",
```

```
        legend.direction = "horizontal",
        legend.title = element_text(face = "plain", size = 12),
        legend.text  = element_text(size = 12)
    )
)

# Combine plots
final_plot <- (p_moral | p_marg) / (p_style | p_age) +
  patchwork::plot_annotation() &
  theme(plot.margin = margin(10, 10, 20, 10))

final_with_legend <- patchwork::wrap_plots(
  final_plot,
  legend_sig,
  ncol = 1,
  heights = c(12, 1)
)

final_with_legend
```




```
# Save
ggsave(
  "panel_plot_APA.png",
  final_with_legend,
  width = 10,
```

```r
  height = 7,
  dpi = 300
)
```

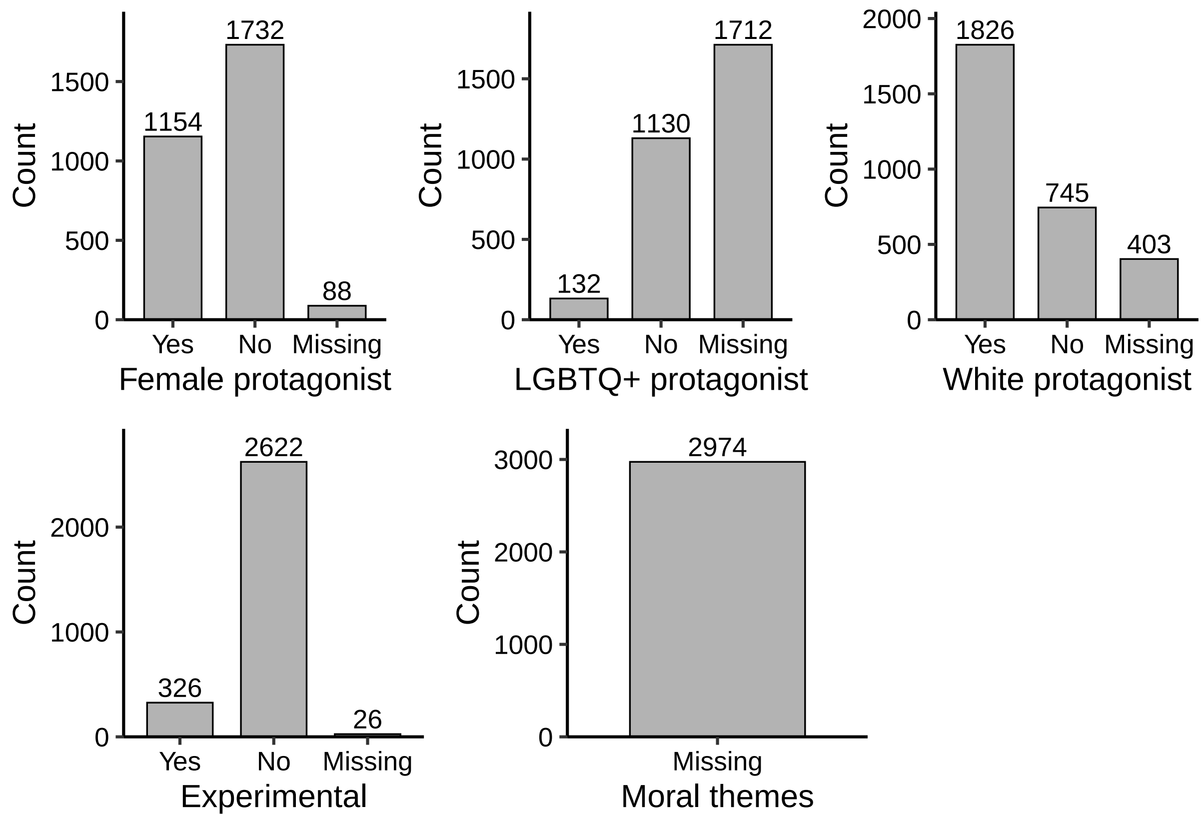
```r
library(ggplot2)
library(patchwork)
library(dplyr)

theme_apa7 <- theme_classic(base_size = 12) +
  theme(
    legend.position = "none",
    panel.grid = element_blank(),
    axis.text = element_text(color = "black"),
    axis.title = element_text(color = "black")
  )

bar_count_apa <- function(df, xvar, xlab) {
  ggplot(df %>%
           mutate(
             !!sym(xvar) := factor(
               case_when(
                 .data[[xvar]] == "Yes" ~ "Yes",
                 .data[[xvar]] == "No" ~ "No",
                 TRUE ~ "Missing"
               ),
               levels = c("Yes", "No", "Missing")
             )
           ),
         aes_string(x = xvar)) +
    geom_bar(fill = "grey70", color = "black", linewidth = 0.3, width = 0.7) +
    geom_text(stat = "count", aes(label = ..count..),
              vjust = -0.3, size = 3.5) +
    labs(x = xlab, y = "Count") +
    scale_y_continuous(expand = expansion(mult = c(0, 0.12))) +
    theme_apa7
}

# ---------- Build the five categorical plots ----------
p1 <- bar_count_apa(data, "female_protagonist", "Female protagonist")
```

```
## Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
## i Please use tidy evaluation idioms with `aes()`.
## i See also `vignette("ggplot2-in-packages")` for more information.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```r
p2 <- bar_count_apa(data, "lgbtq_protagonist",  "LGBTQ+ protagonist")
p3 <- bar_count_apa(data, "white_protagonist",  "White protagonist")
p4 <- bar_count_apa(data, "experimental",       "Experimental")
p5 <- bar_count_apa(data, "moral_themes",       "Moral themes")

# ---------- Combine into one APA-style panel ----------
```

```r
panel_counts <- (p1 | p2 | p3) / (p4 | p5 | plot_spacer())

# Show
panel_counts
```

```
## Warning: The dot-dot notation (`..count..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(count)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



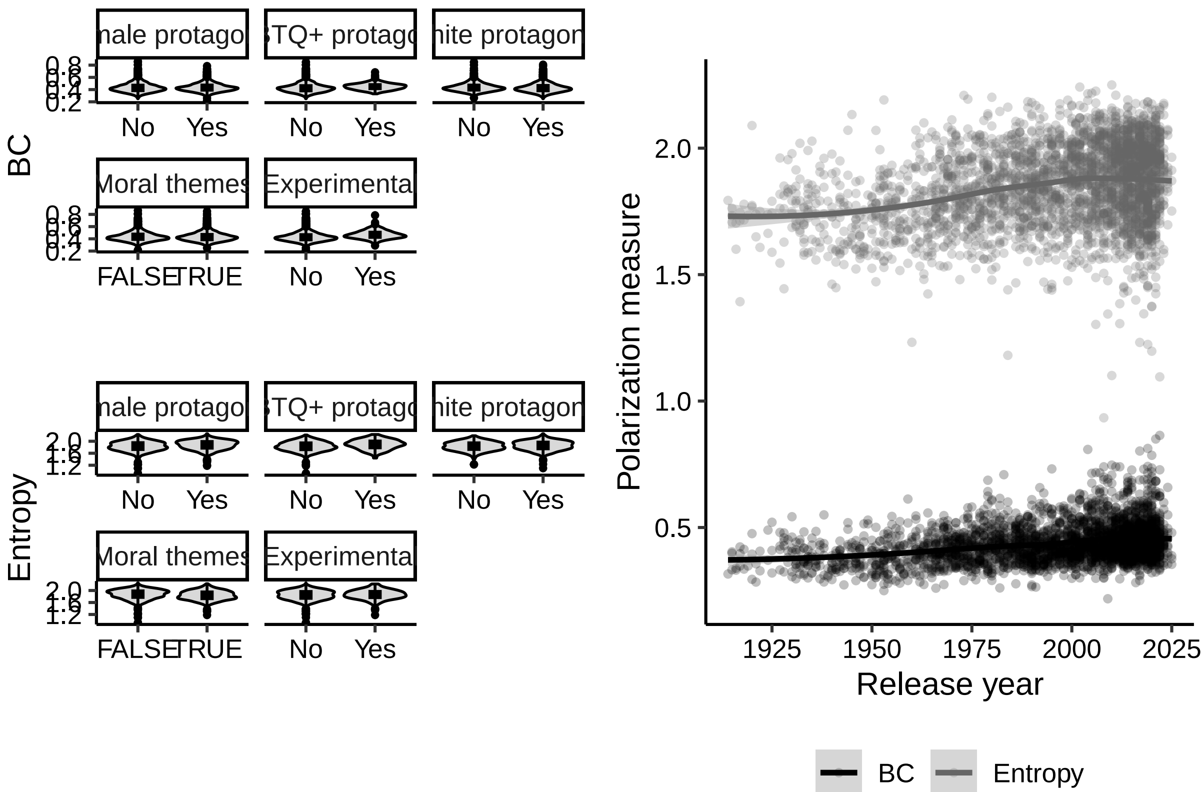

```r
# ---------- Save ----------
ggsave("panel_counts_APA.png", plot = panel_counts, width = 12, height = 8, dpi = 300)
```

```r
library(tidyverse)
library(patchwork)

data_long <- data %>%
  select(BC, polarization, female_protagonist, lgbtq_protagonist,
         white_protagonist, experimental, moral_themes) %>%
  pivot_longer(cols = c(female_protagonist, lgbtq_protagonist,
                        white_protagonist, experimental, moral_themes),
               names_to = "Predictor", values_to = "Level") %>%
  filter(!is.na(Level)) %>%
```

```
  mutate(
    Predictor = factor(
      Predictor,
      levels = c("female_protagonist",
                 "lgbtq_protagonist",
                 "white_protagonist",
                 "moral_themes",
                 "experimental")
    )
  )

# Labels
predictor_labels <- c(
  female_protagonist = "Female protagonist",
  lgbtq_protagonist  = "LGBTQ+ protagonist",
  white_protagonist  = "White protagonist",
  moral_themes       = "Moral themes",
  experimental       = "Experimental"
)

# APA theme
theme_apa7 <- theme_classic(base_size = 12) +
  theme(
    legend.position = "bottom",
    legend.title = element_blank(),
    panel.grid = element_blank(),
    axis.text = element_text(color = "black"),
    axis.title = element_text(color = "black")
  )

p_cats_bc <- ggplot(data_long, aes(Level, BC)) +
  geom_violin(fill = "grey85", color = "black", trim = TRUE) +
  geom_boxplot(width = 0.15, fill = "white", color = "black", outlier.size = 0.8) +
  facet_wrap(~Predictor, scales = "free_x", labeller = labeller(Predictor = predictor_labels)) +
  labs(x = NULL, y = "BC") +
  theme_apa7

p_cats_entropy <- ggplot(data_long, aes(Level, polarization)) +
  geom_violin(fill = "grey85", color = "black", trim = TRUE) +
  geom_boxplot(width = 0.15, fill = "white", color = "black", outlier.size = 0.8) +
  facet_wrap(~Predictor, scales = "free_x", labeller = labeller(Predictor = predictor_labels)) +
  labs(x = NULL, y = "Entropy") +
  theme_apa7

p_cats <- p_cats_bc / plot_spacer() / p_cats_entropy +
  plot_layout(heights = c(1, 0.15, 1))

data_long_year <- data %>%
  select(release_year, BC, polarization) %>%
  pivot_longer(cols = c(BC, polarization),
             names_to = "Measure", values_to = "Value") %>%
  mutate(Measure = dplyr::recode(Measure,
                                 "BC" = "BC",
```

```
                                    "polarization" = "Entropy"))

p_year <- ggplot(data_long_year, aes(x = release_year, y = Value, color = Measure)) +
  geom_point(alpha = 0.25, size = 1) +
  geom_smooth(method = "loess", se = TRUE, linewidth = 1) +
  labs(x = "Release year", y = "Polarization measure") +
  scale_color_manual(values = c("BC" = "black", "Entropy" = "grey40")) +
  theme_apa7

final_plot <- p_cats | p_year
final_plot
```

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




```
ggsave("descriptives_predictors_APA_clean.png", final_plot, width = 14, height = 8, dpi = 300)
```

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