Oscar Predictor

Ricky Truong

June 2025

1: Background

1.1: Introduction

Like many, I was fascinated by the 2025 Oscars race. In particular, one video[1] sparked my interest for this project. It claimed Ariana Grande was unlikely to win the Oscar for Best Supporting Actress since she had not won any of the four awards preceding the Oscars, which are usually good indicators for the final winners. Curious, I wanted to investigate this by looking at previous award seasons. That is, I wanted to know: What can the data tell us?

1.2: Methodology

This project used data on the Academy Awards (Oscars) and the four award ceremonies preceding it: the British Academy of Film and Television Arts (BAFTA) Awards, Critics' Choice Movie Awards, Golden Globe Awards, and Screen Actors Guild (SAG) Awards. Specifically, I was interested in the *Academy Award for Best Supporting Actress*, whose "equivalents" are the *BAFTA Award for Best Actress in a Supporting Role*, *Critics' Choice Movie Award for Best Supporting Actress*, *Golden Globe Award for Best Supporting Actress – Motion Picture*, and *SAG Award for Outstanding Performance by a Female Actor in a Supporting Role*.

To get the data, I started with an available, fully-updated data set for the Oscars found on Kaggle[2]. I also used an available data set for the Golden Globes found on Kaggle[3]; however, this one was missing data after 2020. Unfortunately, I could not find the data readily available for the BAFTA, Critics' Choice, and SAG Awards (as well as for the Golden Globes 2021-2025), so I scraped Wikipedia pages for the rest of the data[4].

For consistency, I denote the "year" as the year when the award ceremony takes place. The first year when data are available for all five award ceremonies (including nominees and winners) is 2002, so that's when my analysis starts. My data set also only contains actresses who were nominated for all five awards that year.

I was originally motivated to use a parametric model. Specifically, logistic regression seemed the sensible choice for a win/lose outcome. However, I soon realized there were some problems; I was interested in using solely binary data to predict the Oscar result, and with 4 awards, there are only $2^4 = 16$ possible variations for the predictors (where some variations, like a complete sweep, perfectly

[1] https://www.youtube.com/watch?v=3Ep7ftG4MfQ
[2] https://www.kaggle.com/datasets/unanimad/the-oscar-award
[3] https://www.kaggle.com/datasets/unanimad/golden-globe-awards
[4] https://github.com/rickystruong/oscar-predictor

predicted an Oscar win). With this came the issue of separation, resulting in a poor-performing logistic regression model. I thus tried the Firth logistic regression model instead, which performed better. However, I realized this situation would be more appropriate with a nonparametric approach, so I also used a decision tree and calculated the "sample" win rates as predictors.

1.3: Load libraries and data sets

```
# Load libraries
library(tidyverse)
library(logistf)
library(rpart)
library(rattle)
library(stringi)

# Load existing data sets from Kaggle
oscars <- read.csv("the_oscar_award.csv")
globes <- read.csv("golden_globe_awards.csv")

# Load data sets scraped online via Python
bafta_best_supp_actress <- read.csv("bafta_best_supp_actress.csv")
critics_best_supp_actress <- read.csv("critics_best_supp_actress.csv")
globes_best_supp_actress_2021_2025 <- read.csv("globes_best_supp_actress_2021_2025.csv")
sag_best_supp_actress <- read.csv("sag_best_supp_actress.csv")
```

1.4: Wrangle data

```
# Create new data set for best supporting actress in Oscars
oscars_best_supp_actress <- oscars %>%
  filter(canon_category == "ACTRESS IN A SUPPORTING ROLE") %>%
  rename(Year = year_ceremony) %>%
  rename(Name = name) %>%
  mutate(Oscar = ifelse(winner == "True", "Won", "Lost")) %>%
  select(Year, Name, Oscar)

# Create new data set for best supporting actress in Golden Globes
globes_best_supp_actress <- globes %>%
  filter(category == "Best Performance by an Actress in a Supporting Role in any Motion Picture
  rename(Year = year_award) %>%
  rename(Name = nominee) %>%
  mutate(Golden_Globe = ifelse(win == "True", "Won", "Lost")) %>%
  select(Year, Name, Golden_Globe)

globes_best_supp_actress <- bind_rows(globes_best_supp_actress, globes_best_supp_actress_2021_
```

1.5: Create final data set

```r
# Standardize names (i.e, remove special characters for inner_join())
oscars_best_supp_actress$Name_Std <- stri_trans_general(oscars_best_supp_actress$Name, "Latin-
bafta_best_supp_actress$Name_Std <- stri_trans_general(bafta_best_supp_actress$Name, "Latin-AS
critics_best_supp_actress$Name_Std <- stri_trans_general(critics_best_supp_actress$Name, "Lati
globes_best_supp_actress$Name_Std <- stri_trans_general(globes_best_supp_actress$Name, "Latin-
sag_best_supp_actress$Name_Std <- stri_trans_general(sag_best_supp_actress$Name, "Latin-ASCII"

# Perform successive inner joins to combine data sets
best_supp_actress <- oscars_best_supp_actress %>%
  inner_join(bafta_best_supp_actress, by = c("Year", "Name_Std")) %>%
  inner_join(critics_best_supp_actress, by = c("Year", "Name_Std")) %>%
  inner_join(globes_best_supp_actress, by = c("Year", "Name_Std")) %>%
  inner_join(sag_best_supp_actress, by = c("Year", "Name_Std"))

# Filter for years 2002-2025 and select specific variables
best_supp_actress <- best_supp_actress %>%
  filter(Year >= 2002, Year <= 2025) %>%
  select(Year, Name, Oscar, BAFTA, Critics_Choice, Golden_Globe, SAG)

# Rewrite "Won" as 1 and "Lost" as 0
best_supp_actress <- best_supp_actress %>%
  mutate(Oscar = ifelse(Oscar == "Won", 1, 0)) %>%
  mutate(BAFTA = ifelse(BAFTA == "Won", 1, 0)) %>%
  mutate(Critics_Choice = ifelse(Critics_Choice == "Won", 1, 0)) %>%
  mutate(Golden_Globe = ifelse(Golden_Globe == "Won", 1, 0)) %>%
  mutate(SAG = ifelse(SAG == "Won", 1, 0))

# Remove incorrect duplicates (an actress is nominated for 2 films the same year)
best_supp_actress <- best_supp_actress %>%
  group_by(Year, Name) %>%
  arrange(desc(Oscar + BAFTA + Critics_Choice + Golden_Globe + SAG)) %>%
  slice(1) %>%
  ungroup()
```

2: Data visualization

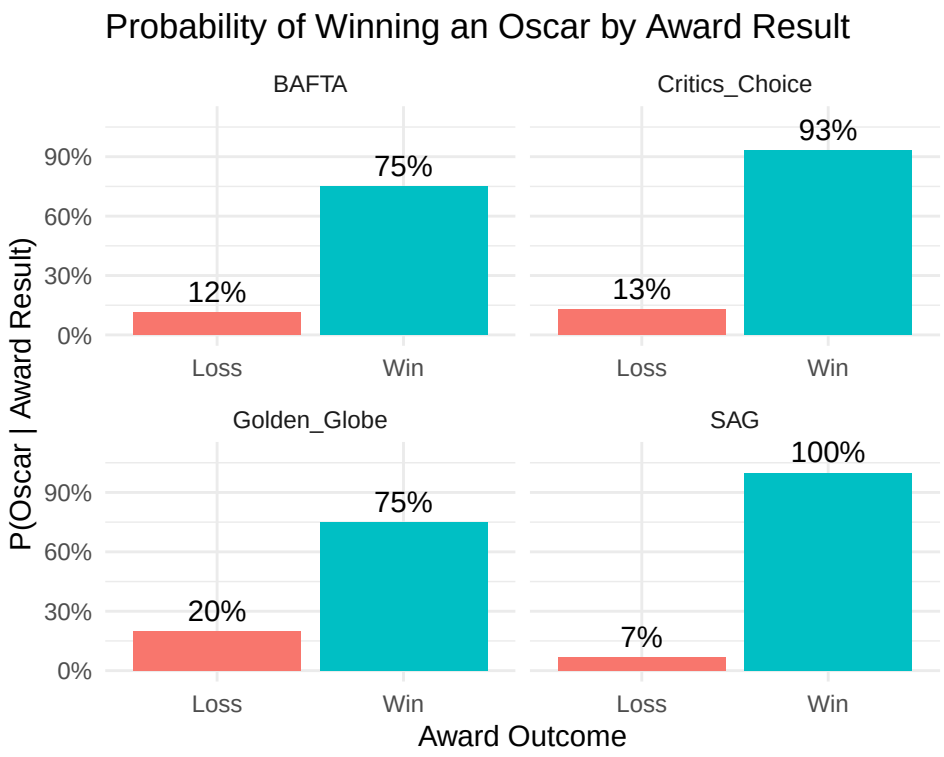
2.1: Visualize probability of winning Oscar based on each award

```r
# Create new data set for each nominee's non-Oscar result
best_supp_actress_long <- best_supp_actress %>%
  mutate(across(c(BAFTA, Critics_Choice, Golden_Globe, SAG), ~ ifelse(. == 1, "Win", "Loss")))
  pivot_longer(cols = c(BAFTA, Critics_Choice, Golden_Globe, SAG),
               names_to = "Award",
               values_to = "Result")

# Summarize counts and proportions
```

```r
award_summary <- best_supp_actress_long %>%
  group_by(Award, Result) %>%
  summarize(Oscar_Wins = sum(Oscar),
            Total = n(),
            Proportion = Oscar_Wins / Total,
            .groups = "drop")

# Plot probability of Oscar win, conditioning on each non-Oscar result
ggplot(award_summary, aes(x = Result,
                          y = Proportion,
                          fill = Result)) +
  geom_col() +
  geom_text(aes(label = scales::percent(Proportion, accuracy = 1)),
            vjust = -0.5, size = 4) +
  facet_wrap(~ Award, scales = "free_x") +
  scale_y_continuous(labels = scales::percent_format(accuracy = 1), limits = c(0, 1.1)) +
  labs(x = "Award Outcome",
       y = "P(Oscar | Award Result)",
       title = "Probability of Winning an Oscar by Award Result") +
  theme_minimal() +
  theme(legend.position = "none")
```

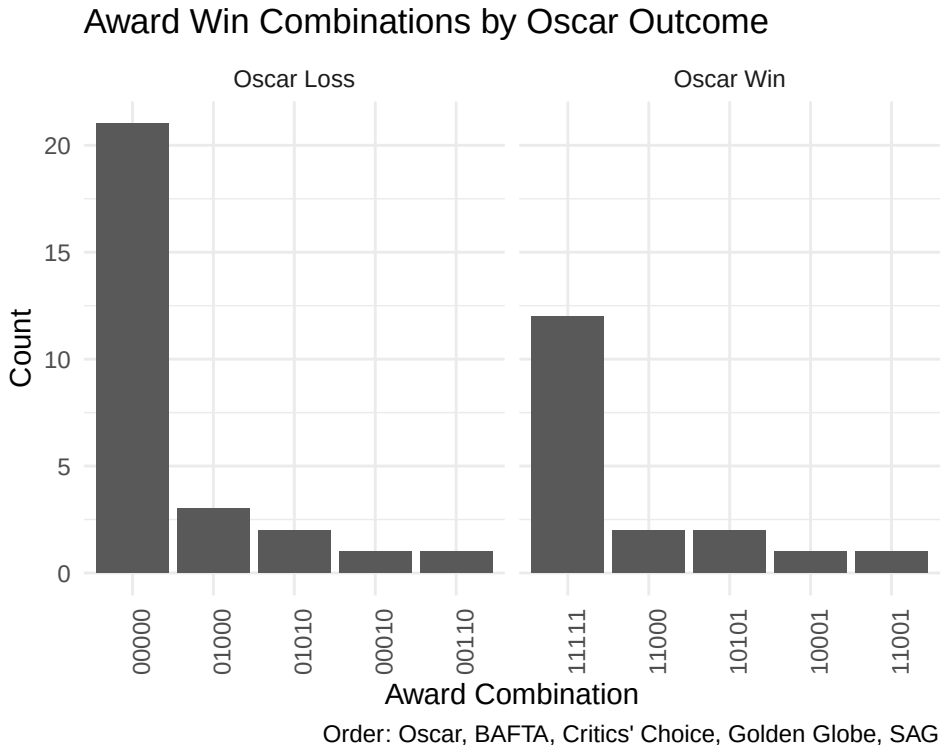


From the plot above, we see a SAG win is incredibly influential. Given an actress wins the SAG, she has 100% probability of winning the Oscar! On the other hand, the Golden Globe Awards are the least informative in that conditioning on a Globe win, there is only a 75% probability of winning the Oscar (while a Globe loss is not the end of the world since there's still a 19% probability of

winning the Oscar after).

2.2: Visualize most common award combinations that lead to Oscar wins/losses

```
best_supp_actress %>%
  # Count award combinations
  count(Oscar, BAFTA, Critics_Choice, Golden_Globe, SAG) %>%
  mutate(combo = interaction(Oscar, BAFTA, Critics_Choice, Golden_Globe, SAG, sep = ""),
         OscarWin = ifelse(Oscar == 1, "Oscar Win", "Oscar Loss"),
         combo = fct_reorder(combo, n, .desc = TRUE)) %>%
  # Plot frequencies
  ggplot(aes(x = combo,
             y = n)) +
  geom_col() +
  facet_wrap(~ OscarWin, scales = "free_x") +
  labs(x = "Award Combination",
       y = "Count",
       title = "Award Win Combinations by Oscar Outcome",
       caption = "Order: Oscar, BAFTA, Critics' Choice, Golden Globe, SAG") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



From the plot above, we see the most common outcome is losing every award while the second most is winning everything.

3: Parametric approach

3.1: Model data using standard logistic regression model

```
# Create standard logistic regression model and print the numbers
model <- glm(Oscar ~ BAFTA + Critics_Choice + Golden_Globe + SAG,
             data = best_supp_actress,
             family = binomial)
summary(model)
```

```
##
## Call:
## glm(formula = Oscar ~ BAFTA + Critics_Choice + Golden_Globe +
##     SAG, family = binomial, data = best_supp_actress)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -22.305   9032.408  -0.002    0.998
## BAFTA             21.900   9032.408   0.002    0.998
## Critics_Choice     1.348  32069.173   0.000    1.000
## Golden_Globe     -21.621  21168.964  -0.001    0.999
## SAG               43.390  24688.494   0.002    0.999
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 61.5782  on 45  degrees of freedom
## Residual deviance:  6.7301  on 41  degrees of freedom
## AIC: 16.73
##
## Number of Fisher Scoring iterations: 21
```

This model does not perform well as indicated by the large and highly variable estimates.

3.2: Model data using Firth logistic regression model

```
# Create Firth logistic regression model and print the numbers
firth_model <- logistf(Oscar ~ BAFTA + Critics_Choice + Golden_Globe + SAG,
                       data = best_supp_actress)
summary(firth_model)
```

```
## logistf(formula = Oscar ~ BAFTA + Critics_Choice + Golden_Globe +
##     SAG, data = best_supp_actress)
##
## Model fitted by Penalized ML
## Coefficients:
##                      coef se(coef) lower 0.95 upper 0.95      Chisq            p
## (Intercept)    -3.649896 1.191375 -8.6232829  -1.750828 26.0952537 3.249819e-07
## BAFTA           3.243972 1.416958  0.5421718   8.449539  5.7172015 1.679949e-02
## Critics_Choice  1.005245 2.151535 -4.6751143   9.140380  0.1554906 6.933426e-01
```

```
## Golden_Globe   -1.509594 1.709647 -7.4904992   1.839908  0.6244163 4.294109e-01
## SAG             4.590115 1.887010  1.1858621  10.700767  7.5772797 5.910858e-03
##                method
## (Intercept)         2
## BAFTA               2
## Critics_Choice      2
## Golden_Globe        2
## SAG                 2
##
## Method: 1-Wald, 2-Profile penalized log-likelihood, 3-None
##
## Likelihood ratio test=44.74737 on 4 df, p=4.487035e-09, n=46
## Wald test = 14.82054 on 4 df, p = 0.005088273
```

Because separation occurs in the original model with our data, I used a Firth logistic regression model instead of a standard one, which performed better as seen by the drastically lower numbers for the estimates and their standard errors.

Our logistic regression model is given by $\log(\frac{P(Y=1|\vec{X})}{P(Y=0|\vec{X})}) = \beta_0 + \beta_1 X_1 + \ldots + \beta_4 X_4$. Interestingly, $\hat{\beta}_4 = 4.6514755$ implies a SAG win multiplies the odds of winning the Oscar by $e^{4.6514755} = 104.739$.

3.3: Predict probability for new observation using the Firth model

```r
# Create person who loses every preceding award
person <- data.frame(BAFTA = 0, Critics_Choice = 0, Golden_Globe = 0, SAG = 0)

# Predict probability that person wins an Oscar
predict(firth_model, person, type = "response")
```

```
## [1] 0.02533526
```

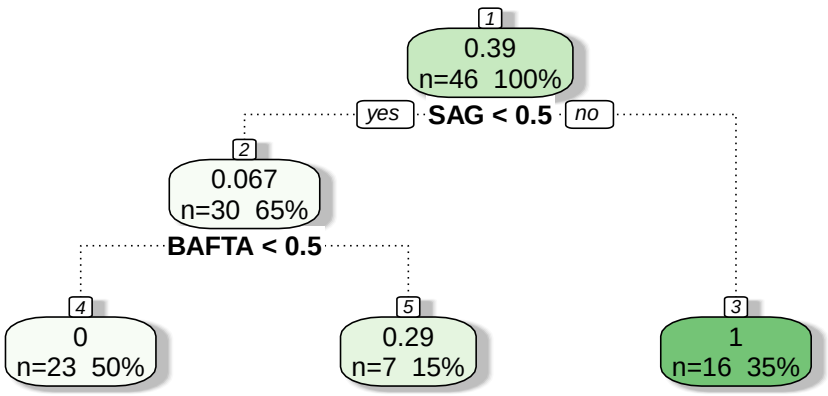
The Firth model predicts, for a person who loses the BAFTA, Critics' Choice, Golden Globe, and SAG Awards, a 2% probability of winning the Oscar (which is also given by $\hat{\beta}_0 = -3.7013533$).

4: Nonparametric approach

4.1: Construct decision tree and visualize it

```r
# Create decision tree
best_supp_actress_tree <- rpart(Oscar ~ BAFTA + Critics_Choice + Golden_Globe + SAG,
                                data = best_supp_actress)

# Visualize it
fancyRpartPlot(best_supp_actress_tree,
               cex = 0.8,
               caption = "",
               type = 2)
```



With these unique data, a nonparametric approach seems more appropriate, so I first constructed a decision tree. This model predicts a probability of 0 for someone who loses the SAG and BAFTA Awards.

Additionally, the decision tree shows the SAG is the most important predictor for the Oscars, which is corroborated by Figure 2.1. We see, conditioning on a SAG loss, the BAFTA becomes the next most important result, which we can verify by visualizing the conditioned data.

4.2: Visualize (conditional) probability of winning Oscar based on each award

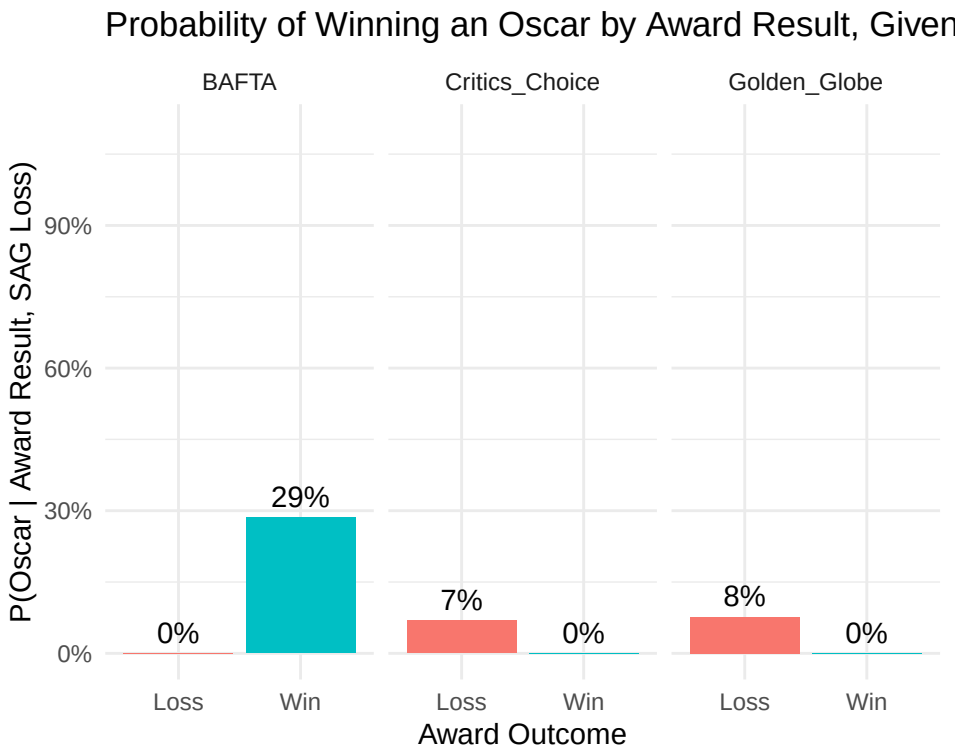
```
# Condition on SAG loss
best_supp_actress_sag_loss <- best_supp_actress %>%
  filter(SAG == 0)

# Create new data set for each nominee's non-Oscar result
best_supp_actress_long_sag_loss <- best_supp_actress_sag_loss %>%
  mutate(across(c(BAFTA, Critics_Choice, Golden_Globe), ~ ifelse(. == 1, "Win", "Loss"))) %>%
  pivot_longer(cols = c(BAFTA, Critics_Choice, Golden_Globe),
               names_to = "Award",
               values_to = "Result")

# Summarize counts and proportions
award_summary_sag_loss <- best_supp_actress_long_sag_loss %>%
  group_by(Award, Result) %>%
  summarize(Oscar_Wins = sum(Oscar),
            Total = n(),
            Proportion = Oscar_Wins / Total,
            .groups = "drop")

# Plot probability of Oscar win, conditioning on each non-Oscar result
ggplot(award_summary_sag_loss, aes(x = Result,
                                   y = Proportion,
```

```
                                    fill = Result)) +
  geom_col() +
  geom_text(aes(label = scales::percent(Proportion, accuracy = 1)),
            vjust = -0.5, size = 4) +
  facet_wrap(~ Award, scales = "free_x") +
  scale_y_continuous(labels = scales::percent_format(accuracy = 1), limits = c(0, 1.1)) +
  labs(x = "Award Outcome",
       y = "P(Oscar | Award Result, SAG Loss)",
       title = "Probability of Winning an Oscar by Award Result, Given Sag Loss") +
  theme_minimal() +
  theme(legend.position = "none")
```



The plot above reaffirms our decision tree in 4.1.

4.3: Compute sample win rates to predict probability for new observations

```
# Compute sample win rates
best_supp_actress %>%
  group_by(BAFTA, Critics_Choice, Golden_Globe, SAG) %>%
  summarize(
    count = n(),
    oscar_wins = sum(Oscar),
    win_rate = mean(Oscar)
  ) %>%
  arrange(desc(win_rate))
```

```
## # A tibble: 9 x 7
```

```
## # Groups:   BAFTA, Critics_Choice, Golden_Globe [7]
##   BAFTA Critics_Choice Golden_Globe   SAG count oscar_wins win_rate
##   <dbl>          <dbl>        <dbl> <dbl> <int>      <dbl>    <dbl>
## 1     0              0            0     1     1          1      1
## 2     0              1            0     1     2          2      1
## 3     1              0            0     1     1          1      1
## 4     1              1            1     1    12         12      1
## 5     1              0            0     0     5          2      0.4
## 6     0              0            0     0    21          0      0
## 7     0              0            1     0     1          0      0
## 8     0              1            1     0     1          0      0
## 9     1              0            1     0     2          0      0
```

Finally, as the most straight-forward approach, I directly computed the sample win rates for each observed combination. Based on our data, for the 22 actresses who lost the BAFTA, Critics' Choice, Golden Globe, and SAG awards, none of them won the Oscar, a win rate of 0/22. Thus, we estimate the probability of winning an Oscar given these four losses to be 0%, which pretty much agrees with our results from 3.3 and 4.1.

5: Conclusion

5.1: Summary

Regardless of whether we use a logistic regression model or a nonparametric approach, we see the probability an actor wins the Oscar for Best Supporting Actress given she lost the BAFTA, Critics' Choice, Golden Globe, and SAG Awards is estimated to be practically zero. This agrees with the original argument: Ariana Grande's chances diminished with every loss as awards season continued.

Also, within the four awards we use as predictors, we see the SAG Award is the most important. On the other hand, the Golden Globe Awards lend the least amount of information toward the Oscar win.

5.2: Future considerations

The logistic regression model most likely could have been improved by adding more (quantitative) predictors like box office gross and Rotten Tomatoes score, but my original question solely concerned the four other award ceremonies. Improving upon this model is something I'd be interested in doing later down the line. Additionally, performing this analysis on the other categories (like *Academy Award for Best Actor*) would be interesting.

I take responsibility for all mistakes; please contact me if you find any!

Email: rickytruong23@gmail.com