Math 150 - Methods in Biostatistics - Homework 6

your name here

Due: Wednesday, March 1, 2023

Assignment Summary (Goals)

- considerations of working with variables
- fluent use of the multiple logistic model for prediction using the **tidymodels** framework

Note that if you don't know the R code either check the class notes or ask me!!! Happy to scaffold, debug, send resources, etc. Don't go down a rabbit hole trying to figure out an R function or syntax.

Also, note that you'll need to get the data from Canvas and use it for this analysis. Look back to your own HW1 file to see the line of code **you** used to import the `games1.csv` dataset. Ask me if it isn't obvious to you after you look at your own HW1.

Q1. Collaborative Learning Describe one thing you learned from someone (a fellow student or mentor) in our class this week (it could be: content, logistical help, background material, R information, etc.) 1-3 sentences.

Q2. And the Winner Is... The data this week come from kaggle as a compilation form Academy Award winners since the award began. https://www.kaggle.com/datasets/unanimad/the-oscar-award

The data wrangling below are an effort to consolidate the labels so that the variables measure the same over basic quality over the last 95 years. https://www.verdict.co.uk/oscars-90-whats-changed-years-oscars-records/

The dataset is in the files tab on Canvas (in a folder called "data").

```
# the only thing you should change in this R chunk is
# the path of the dataset
the_oscar <- readr::read_csv("~/Dropbox/teaching/MA150/the_oscar_award.csv") %>%
  rename(year = year_ceremony) %>%
  filter(year != 2022) %>%
  select(-year_film, -ceremony, -name) %>%
  drop_na() %>%
  mutate(winner = case_when(
    winner == TRUE ~ 1,
    winner == FALSE ~ 0
  )) %>%
  distinct(category, film, .keep_all = TRUE) %>%
  mutate(category = case_when(
    category == "OUTSTANDING PICTURE" ~ "BEST PICTURE",
    category == "OUTSTANDING MOTION PICTURE" ~ "BEST PICTURE",
    category == "OUTSTANDING PRODUCTION" ~ "BEST PICTURE",
    category == "BEST MOTION PICTURE" ~ "BEST PICTURE",
    category == "ACTOR" ~ "ACTOR IN A LEADING ROLE",
    category == "ACTRESS" ~ "ACTRESS IN A LEADING ROLE",
    category == "INTERNATIONAL FEATURE FILM" ~ "FOREIGN LANGUAGE FILM",
    str_detect(category, "DIRECTING") ~ "DIRECTING",
```

```
    str_detect(category, "WRITING") ~ "WRITING",
    str_detect(category, "VISUAL EFFECTS") ~ "VISUAL EFFECTS",
    str_detect(category, "SOUND") ~ "SOUND",
    str_detect(category, "SHORT SUBJECT") ~ "SHORT SUBJECT",
    str_detect(category, "SHORT FILM") ~ "SHORT FILM",
    str_detect(category, "MUSIC") ~ "MUSIC",
    str_detect(category, "MAKEUP") ~ "MAKEUP",
    str_detect(category, "DOCUMENTARY") ~ "DOCUMENTARY",
    str_detect(category, "COSTUME DESIGN") ~ "COSTUME DESIGN",
    str_detect(category, "CINEMATOGRAPHY") ~ "CINEMATOGRAPHY",
    str_detect(category, "ART DIRECTION") ~ "ART DIRECTION",
    str_detect(category, "PRODUCTION") ~ "PRODUCTION",
    TRUE ~ category)) %>%
  filter(!(category %in% c("OUTSTANDING PICUTRE",
                           "UNIQUE AND ARTISTIC PICTURE",
                           "SPECIAL ACHIEVEMENT AWARD (Sound Effects)",
                           "SPECIAL ACHIEVEMENT AWARD (Visual Effects)",
                           "SPECIAL ACHIEVEMENT AWARD (Sound Effects Editing)",
                           "SPECIAL ACHIEVEMENT AWARD (Sound Editing)",
                           "ENGINEERING EFFECTS",
                           "DANCE DIRECTION",
                           "ASSISTANT DIRECTOR"
                           ))) %>%
  group_by(film, category) %>%
  arrange(desc(winner)) %>%
  filter(row_number() == 1) %>%
  ungroup() %>%
  pivot_wider(id_cols = c("year", "film"),
              names_from = category, values_from = winner,
              values_fill = 0) %>%
  janitor::clean_names() %>%
  mutate(best_picture = as.factor(ifelse(best_picture == 1, "win", "not win")))
```

(a) Create a logistic regression model using all the explanatory variables except `film` (which is the title of the film).

Use the framework from **tidymodels** which consists of the following steps:

- split the data into test training. you likely want to stratify on `best_picture` (because the dataset is very imbalanced).
- build a recipe. to communicate that you don't consider `film` to be an identifier, add the following step to your recipe: `update_role(film, new_role = "ID")`
- set the model to be logistic regression
- fit the model on the training data
- tidy the model to see the coefficients / p-values

(b) The glm.fit likely said that the fitted probabilities were numerically 0 or 1. Check the following three probability of best picture win: Cinema Paradiso, Parasite, The Hurt Locker (you'll likely need to Google / look at wikipedia).

(c) Use the test data to predict whether or not each of the test movies will win an Academy Award for Best Picture. Summarize using accuracy, sensitivity, and specificity (here "successes" is "win"). Feel free to also make a plot to describe the table of predictions.

Not due, but maybe fun? (d) Is your model able to predict the winner for best picture from 2022? (The dataset only goes up until 2021, so you'll need to find the relevant information from the Google.)

Q3. And the Winner Is... Continue to use the data on the Academy Awards. Using 10-fold Cross Validation, compare the following two models:

1. A model based only on the four actor awards, makeup, and music
2. A model based only on directing and writing

(a) Build the models on the training data only using cross validation. (Note: in the formula add the variables and also add `+ film` before updating the film role to be only an ID.)

(b) Cross validate to assess which model is better. Choose a model based on overall accuracy. Commit to that model using only the training data. Which model did you choose?

(c) What do you think the accuracy of the model (from part (b)) will be when you apply your model in the real world? That is, use the variables from part (b), train the model using the training data, test the model using the test data.

Q4. Voting Another dataset which could have been used for this HW set was a dataset that comes from 538 on voting behavior. See the information about the dataset here: https://github.com/fivethirtyeight/data/tree/master/non-voters and the article about it here: https://projects.fivethirtyeight.com/non-voters-poll-2020-election/.

If you want, you can also import the data into R using the following code.

```
voters <- read_csv("https://raw.githubusercontent.com/fivethirtyeight/data/master/non-voters/nonvoters_
```

Consider the following variables:

a. Q21: do you plan to vote in Nov 2020?
b. Q30: which political party do you consider yourself aligned with?
c. INCOME_CAT: household income category

For each variable, explain (in words, no code here) how the variable would need to be transformed to be able to be used in the logistic regression model.

```
praise()
```

```
## [1] "You are cool!"
```