Assignment 2

Source of this assignment.

The purpose of Assignment 2 is to practice data processing, visualization, and inference using the HETREC Movie data we have been using for examples in class.

Please check the due day on Syllabus and Canvas. Submit your `.ipynb` and PDF files to Canvas.

Data and Setup (25%)

For this assignment, you will work with the HETREC Movie data.

Consult the work from class and the tutorial notebooks for code to load the data, and many hints!

Pay attention to the Missing Data notebook for handling missing data. **Use the strategies in that notebook to replace the '0's used to encode unknown RottenTomatoes ratings with missing (NA) values.**

{{mtodo}} Set up your notebook to load the data, convert erroneous 0s to NAs, and show the size & columns of your data set.

Comparing Ratings (25%)

{{mtodo}} **Describe the distributions** of the RottenTomatoes critic ratings (All Critics and Top Critics), the Audience Rating, and the mean rating given to a movie by MovieLens users, both numerically and graphically.

{{mtodo}} **Describe the distribution** of the *difference* between the All Critics and Top Critics ratings for movies where both are defined, both numerically and graphically.

{{mtodo}} Answer the following questions using **paired T-tests** (the SciPy `ttest_rel` function computes this):

- Do the data indicate a difference between the ratings given to movies by all critics and those given by top critics?
- Do the data indicate a difference between the average audience rating RottenTomatoes users give to a movie and the mean rating MovieLens users give to it?

Consider: why is the paired t-test the appropriate test here?

:::{admonition} Missing Data :class: tip

The SciPy test functions have an `nan_policy` function, and if you pass `nan_policy='omit'` they will ignore missing values instead of propagating them into NaN results. I recommend doing this, and also dropping NAs in your bootstraps. :::

Confidence Intervals (25%)

We now want to see if some genres of movies fare better with critics than others.

{{mtodo}} For each of the 20 genres, compute the mean and a 95% confidence interval for the all-critic ratings using the standard error method. Show the results as a data frame sorted by decreasing mean (look up the `sort_values` method in Pandas). Does it look like the top two genres have different mean critic ratings? Does it look like the top and bottom genres have different mean critic ratings? Defend your answers using the confidence intervals.

:::{admonition} Vectorization :class: tip

You can do all of this with vectorized operations. Start with a frame whose rows are genres, and whose columns are the mean, count, and standard deviation (and/or standard error of the mean) of the all-critic ratings for movies in that genre. That will let you compute all the confidence intervals in just a handful of Python operations. :::

:::{admonition} Joining :class: tip

If you join or merge the movie genre table with your movie info or stats table on the movie ID, it will**duplicate** each movie for each genre it has. Grouping by genres and aggregating will then compute your aggregate statistics, such as the mean, correctly. :::

{{mtodo}} For each of the 20 genres, compute the mean and a 95% bootstrapped confidence interval for the mean all-critic rating. Show the result in a table. Does this look the same as the standard error CIs?

:::{admonition} Group-Apply :class: tip

Remember the group-apply we saw in Penguin Inference? That will help here too! You can bootstrap inside the function instead of computing an error-based confidence interval. :::

:::{admonition} Independence :class: note

These groups are *not* indepednent. We can compute confidence intervals, but making group comparisons require more care. :::

Popularity and Bootstraps (20%)

Action movies are most likely more popular than documentaries. By this I mean that more people are likely to watch an action movie than a documentary.

Compute the number of MovieLens users who have rated each movie. This will yield observations of *movies* and their *number of ratings*.

{{mtodo}} Test the null hypothesis that action movies and documentaries have the same *median number of ratings* using a bootstrapped *p*-value. Does your test accept or reject the null? What *are* the median number of ratings for movies in each of these genres?

What if you use the # of audience ratings from RottenTomatoes instead of the # of MovieLens ratings?

:::{admonition} Bootstrapping the Test :class: tip

This will use the same technique as we used in Penguins to bootstrap a test for different means. :::

{{mtodo}} Compare the mean of the critic ratings (using the All Critics ratings from Rotten Tomatoes) between action and documentary movies. Is there a difference? Test the difference with both the bootstrap and an approprate *t*-test.

Reflection (5%)

{{mtodo}} Write 2 paragraphs about what you have learned through this assignment.

If you have comments on the accuracy of time estimates, I would also appreciate those.

Time Estimates

This is my estimated times, similar to A1:

- Data and Setup: 30 min
- Comparing Ratings: 1 hour
- Confidence Intervals: 90 minutes
- Bootstrap: 2 hours
- Reflection: 30 minutes