Mid-Project Checkpoint

Molly Srour

MSIA 423

5/15/19

Highlights

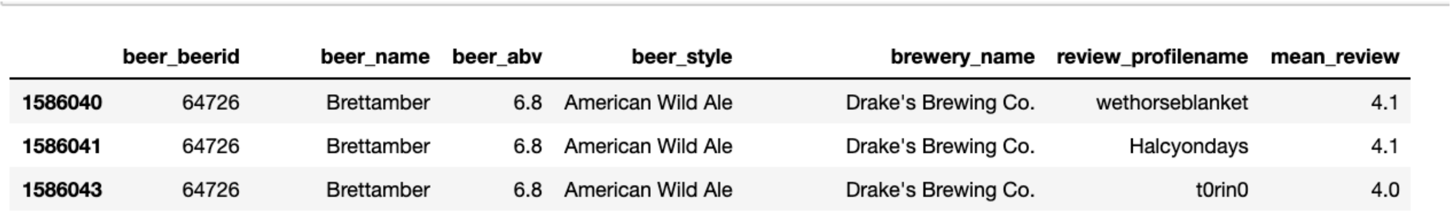
- Dataset broken down into 10 datasets by beer category; rec system predicts based on user choice of beer category
- Due to data constraints, predictions must be pre-entered prior to app deployment – therefore a popularity-based rec system will be used
- Data organized and ready to deploy full dataset into RDS to be queried in flask app

Review Progress

- ***Theme 1: Targeted Beer Recommendation System***
  - **Epic 1: Data Organization and Cleaning**
    - Story 1: Research and Data Cleaning - COMPLETED
    - Story 2: Beer Categorization - COMPLETED
    - Story 3: Sparse Matrices - COMPLETED
  - **Epic 2: Python-Based Model**:
    - Story 2: Introductory Model Building - COMPLETED
- ***Theme 2: Model Communication with App and User Interface***
  - **Epic 1: S3 and RDS - Linking Model with App**
    - Story 1: S3 - Put Model and Sparse Matrices in S3 - Raw Data is in S3
    - Story 2: RDS - Put Database in RDS - Created introductory Table

Demo / Analysis

- Data Wrangling – 10 data frames based on broad categories of beer that I created through aggregation of beer_style
- Data sample below shows prediction by popularity for sour beers, still have not come up with accuracy measure for popularity-based recommendation system



	beer_beerid	beer_name	beer_abv	beer_style	brewery_name	review_profilename	mean_review
1586040	64726	Brettamber	6.8	American Wild Ale	Drake's Brewing Co.	wethorseblanket	4.1
1586041	64726	Brettamber	6.8	American Wild Ale	Drake's Brewing Co.	Halcyondays	4.1
1586043	64726	Brettamber	6.8	American Wild Ale	Drake's Brewing Co.	t0rin0	4.0

Lessons Learned

- I originally wanted to have a complicated recommendation system that would give different predictions based on expert or novice level of beer knowledge − but app constraints forced me to narrow my vision
- RDS can store the dataset for querying based on popularity predictions

Recommendations

Next Steps:

- User interface creation
- HTML encoding
- Flask app deployment
- Modularizing model code