To do:

* Redo the synthetic data
  + Look into synthetic data libraries
  + SDV github: <https://github.com/sdv-dev> for creating synthetic data
* Permutation for feature importance

15 slide ppt (pdf or ppt)

Github link

User shows up to the bar:

Orders first drink, then an item based recommender system

After 5 drinks then collaborative recommender system

Hybrid based rec system

Recommender systems in python 101: <https://www.kaggle.com/code/gspmoreira/recommender-systems-in-python-101>

<https://medium.com/aimonks/evaluating-the-performance-of-recommender-systems-an-insight-into-metrics-and-methodologies-2e5b81a2c958>

There is a class project that is done by each student. The purpose of the project is to test the students’ ability to apply the machine learning techniques and tools presented in the class to diverse practical problems and interpret & apply the results.

The class project covers:

Choose a real-world machine learning problem with an objective

• Procure and analyze the data

Source: <https://dachang.github.io/CocktailViz/>

Columns:

A screenshot of a computer

Description automatically generated

Count: 48 rows

First 5 Rows:

A screenshot of a phone

Description automatically generated

• Develop at a deep learning or recommender system model

* Content-based filtering: suggesting cocktails based on other cocktails
  + Cosine similarity
* Collaborative filtering: with user ratings you can recommend cocktails based on user similarity
  + SVD
* Deep learning model: if you have a large amount of data
  + Neural Networks
    - SVD
    - kNN

• Outline data / model assumptions & limitations / hypotheses

• Evaluate model metrics such as overfitting / underfitting

• Select final model based on selection criteria

• Determine how the model can be improved

• Effectively communicate analytical work

For the class project, students are graded on 8 criteria outlined below. Each criterion is 5 points for a total of 40 points. The criteria are:

1. Problem Statement

2. Assumptions/Hypotheses about data and model

3. Exploratory Data Analysis

4. Feature Engineering & Transformations

5. Proposed Approaches (Model) with checks for overfitting/underfitting

6. Proposed Solution (Model Selection) with regularization, if needed

7. Results (Accuracy) and Learnings from the methodology

8. Future Work

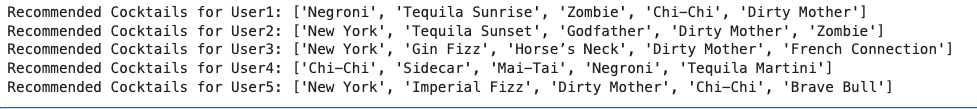
Feedback Loop- retraining based on new data (either new cocktails recipes added or user preferences change)

Synthetic dataset: training 90k, validation 10k

Testing: Survey data 100 records, use up-sampling to create total of 10k records

To do:

1. Create a survey and send out to friends and family
2. De-identify and use user\_id



PPT

Slide 1: Title Slide

- Cocktail Recommender System

- Dharti Seagraves

- May 22, 2023

Slide 2: Problem Statement

- Description of the problem

An owner of a new restaurant in town has just hired an inexperienced bartender to keep the costs down. The bartender is not great at recommending drinks to customers so the owner wants to use a machine learning approach to solve this problem.

- Objective of the project

The objective of this project is to use a machine learning algorithm to create an app the bartender can use under the counter to recommend to his/her customers based on the drinks the customer has already purchased.

Slide 3: Assumptions and Hypotheses

- Assumptions about the data

- Hypotheses for the model

Slide 4: Exploratory Data Analysis (EDA)

- Key insights from EDA

- Visualizations

Slide 5: Feature Engineering and Transformations

- Description of features used

- Feature selection process

Train a RandomForest model to determine feature importances and use RFE for feature selection.

Slide 6: Model Approaches

- Description of different models used

- Overfitting/Underfitting checks

* **Model-based**: This approach, models are developed using different machine learning algorithms to recommend items to users. There are many model-based CF algorithms, like neural networks, bayesian networks, clustering models, and latent factor models such as Singular Value Decomposition (SVD) and, probabilistic latent semantic analysis.
* **SDV example**: <https://colab.research.google.com/drive/1MCTkTj9-93Ei-cLDQoj9AXaqPhpue7a3?usp=sharing#scrollTo=oG2kyPzsFwEG>

Slide 7: Model Selection and Regularization

- Criteria for selecting the final model

- Regularization techniques used

Slide 8: Results and Learnings

- Accuracy and other metrics for train/val/test sets

- Key learnings from the methodology

Slide 9: Future Work

- Suggestions for future improvements

- Potential next steps

- Challenges for collaborative filtering

Sparsity – most users don’t provide ratings so matrix will be sparse, lots of empty cells

With a lot of users or a lot of show the model would slow down considerably

What happens when there’s a new show and no one has watched it. Do we wait for some people to watch it and then rate it and then it gets added into the user ratings table?

Grey sheep: falling on the border of different groups

Black sheet: not really fitting near any of the groups (just on an island by themselves)