NYU | Center for Data Science

Final project

Recommender system

- Develop and evaluate a collaborative filter recommender system using the MovieLens dataset(s)

- Explicit feedback collaborative filter
    - Small version: 9000 movies × 600 users
    - Full version: 58000 movies × 280000 users

Project structure

- You can work in **groups of up to 3**
    - Groups are self-organizing on Brightspace and GitHub Classroom
    - You need to join a group on Brightspace before accessing the assignment
    - You'll need to manage group membership yourselves on both platforms
- Grading:
    - 80% for basic recommender implementation
    - 20% extension(s) of your choice
- Groups of 1 or 2 must do 1 extension for full credit
  Groups of 3 must do 2 extensions

80%: basic recommender requirements

- Data partitioning (train, validation, test)
- Baseline popularity model
- Latent factor model
- Ranking evaluation

20%: Extensions

Pick from these, or propose your own idea:

- Benchmark Spark ALS against single-machine implementations
- Accelerated / approximate search
- Cold-start model using tag genome data
- Qualitative error analysis and visualization

Data partitioning

- **Do this first!**

  Write it as a script and save the results out to separate files to save time.

- Partition each user’s history into **training**, **validation**, and **test**

- Use **validation** data to estimate generalization performance

- Only use test at the very end when you’re finished.
  Do not let test performance influence your modeling - that’s cheating!

Evaluation

- MovieLens has explicit feedback, but we'll evaluate with ranking metrics
  - meanAP @ 100 (top 100 predictions)
  - Include others as needed: NDCG@100 would be a good choice

- You can also use RMSE both for optimization and as an evaluation criterion, but it shouldn't be your only metric

Hyper-parameters to tune

- Bias model:
  - Damping factors β
- Latent factor model:
  - Rank (dimensionality)
  - Regularization penalty
- The range of these parameters is up to you to explore
  - General tip: start with small values and increase until validation performance degrades
  - Which metric you use for validation is up to you, but document and justify your choice

Deadlines and submission

- **2022-04-29**: Checkpoint submission
  - You should have the popularity baseline working on small and large datasets
  - Preliminary results of latent factor model on small dataset
- **2022-05-17**: Final submission (no extensions)
  - Submit full report as PDF via brightspace
  - Include link to your group's github repository
  - Full list of requirements is in the project README

General tips

- When consulting spark documentation, make sure to use version 3.0.1
    - Bookmark this to avoid being mislead by googling!
- Start small and start early!
    - Fitting on the large dataset can take quite a while
    - Shake out the bugs on the small set first
    - Develop locally on your own machines to start out
- Explore the data and experiment
    - Don't leap into modeling before getting familiar with the data
    - **Read the documentation (README.txt) for each dataset!**
    - Actually look at your popularity model's predictions using titles, not identifiers!
- Use other software as necessary
    - Don't rely *only* on spark - it's one tool among many.