NYU | Center for Data Science

Week 10.3: Recommender evaluation

DS-GA 1004: Big Data

Some questions

- What is the workflow for recommender systems modeling?
- How should we evaluate a model?

Modeling workflow in machine learning

1. Obtain **training** and **testing** data
   a. *Test data should be independent of training data!*

2. Fit model to **training** data
   a. *Minimize training error, or whatever other objective you have…*

3. Evaluate model on **testing** data
   a. *Test data should be independent of each other as well…*
   b. $E[\text{risk}] \approx 1/N \cdot \sum_{x \in \text{Testing}} \text{loss}(x \mid \text{model})$

Working with recommender data

- It’s tempting to think of observations (u, v) as independent, but they aren’t!

- What are we actually predicting, and what do we value?
    - Typically we care about satisfying a **user**
    - This should influence our evaluation criteria

- Most interfaces provide several recommendations at once
    - Evaluate the **collection of recommendations** per user
    - **Average across users** to estimate system performance

Partitioning data

- We often need to partition training data for validation / parameter tuning
    - This is just a proxy for our eventual evaluation!
- It’s tempting to randomly split interactions, but this can fail badly
- The model needs some history for each user that it will predict upon
- ⇒ Partition each user’s observations into train / val separately

Evaluating recommender systems

- Modeling objectives are usually just a proxy for our main goal
- Early RecSys work focused on **mean squared error** (MSE) of star ratings
  - (Thanks, NetFlix)

Evaluating recommender systems

- Modeling objectives are usually just a proxy for our main goal
- Early RecSys work focused on **mean squared error** (MSE) of star ratings
  - (Thanks, NetFlix)
- Instead, think about how recommendations are ***delivered***
  - Ranked list? (NetFlix, Google, Amazon)
  - One at a time? (Pandora, streaming radio, YouTube autoplay)
- ***Evaluation should reflect user behavior!***

Ranking evaluations

Prediction: ordered list of items, **reference data**: held-out interactions

- **AUC** (area under ROC curve)
  - How often does a **+ interaction** rank ahead of a **- interaction**?
  - **-+-++--** ⇒ (3 + 2 + 2) / (3 * 4) = 7/12 = 0.583

Ranking evaluations

Prediction: ordered list of items, **reference data**: held-out interactions

- **AUC** (area under ROC curve)
  - How often does a **+ interaction** rank ahead of a **- interaction**?
  - **-+-++--** ⇒ (3 + 2 + 2) / (3 * 4) = 7/12 = 0.583
- **Average Precision** (AP)
  - For each **+ interaction**, what fraction of higher-ranked items were also **positive**?
  - **-+-++--** ⇒ ⅓ * (½ + ½ + ⅗) = 0.533

Ranking evaluations

Prediction: ordered list of items, **reference data**: held-out interactions

- **AUC** (area under ROC curve)
  - How often does a **+ interaction** rank ahead of a **- interaction**?
  - -+-++-- ⇒ (3 + 2 + 2) / (3 * 4) = 7/12 = 0.583
- **Average Precision** (AP)
  - For each **+ interaction**, what fraction of higher-ranked items were also **positive**?
  - -+-++-- ⇒ $\frac{1}{3} * (\frac{1}{2} + \frac{1}{2} + \frac{3}{5}) = 0.533$
- **Reciprocal rank** (MRR)
  - The inverse rank position of first **+ interaction**
  - -+-++-- ⇒ $\frac{1}{2}$

Ranking evaluations

Prediction: ordered list of items, **reference data**: held-out interactions

- **AUC** (area under ROC curve)
  - How often does a **+ interaction** rank ahead of a **- interaction**?
  - **-+-++--** ⇒ (3 + 2 + 2) / (3 * 4) = 7/12 = 0.583
- **Average Precision** (AP)
  - For each **+ interaction**, what fraction of higher-ranked items were also **positive**?
  - **-+-++--** ⇒ ⅓ * (½ + ½ + ⅗) = 0.533
- **Reciprocal rank** (MRR)
  - The inverse rank position of first **+ interaction**
  - **-+-++--** ⇒ ½

Users are assumed to be independent, and scores are averaged across users at evaluation time.

In real life...

- In practice, recommender systems exist in a **feedback loop**
- These are extremely difficult to measure offline from **observational data**
  - Interpret ranking metrics with healthy skepticism!
- Competing models are often evaluated by A/B testing, and measuring some dependent variable
  - Engagement, sales, etc.
- Recently, focus is shifting to reinforcement learning and causal models

Summary

part 3

- Properly evaluating a recommender system is not easy!

- Ranking metrics are a start, but there's much more to it than "accuracy"
    - Diversity? Novelty? Serendipity?
    - Efficiency? Ease of use?
    - Explainability / transparency?
    - Adverse effects on users?