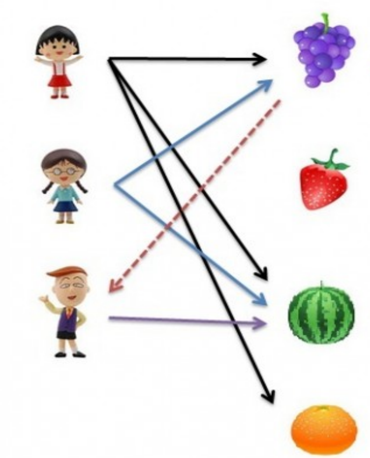
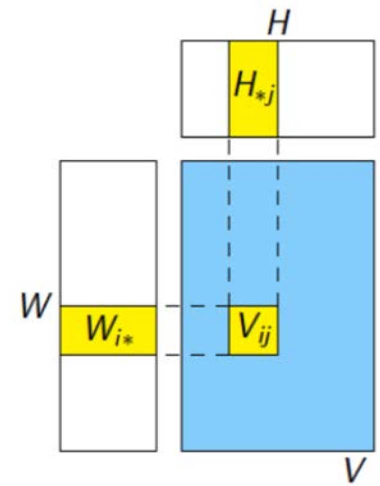
Recommendation Systems



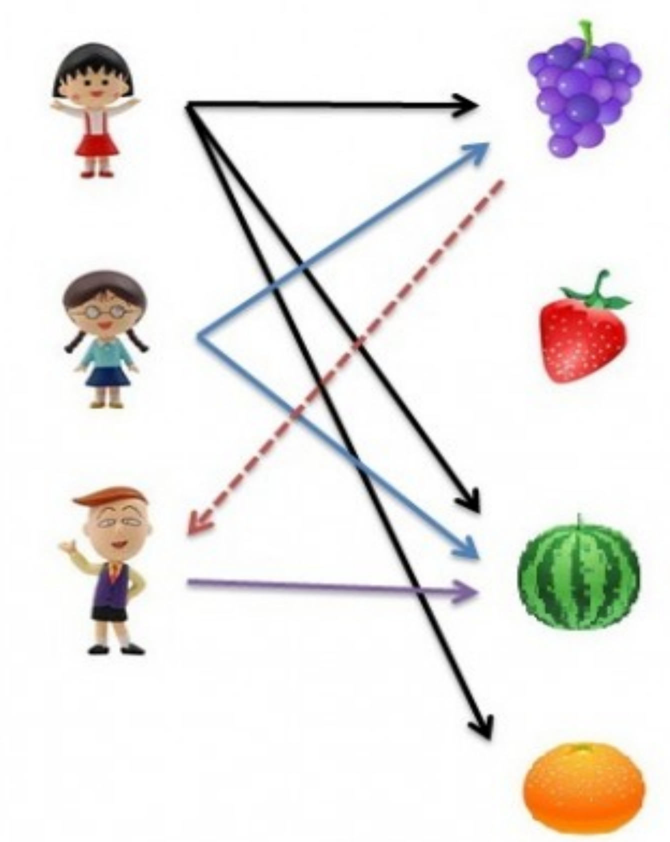
2		4	1
5		3	
2	4	5	



Prof. Mike Hughes

Many ideas/slides attributable to:
Liping Liu (Tufts), Emily Fox (UW)
Matt Gormley (CMU)

Recommendation Task:
Which users will like which items?



- Need recommendation everywhere

Utility matrix

- The “value” or “utility” of items to users
  - Only known when ratings happen
  - In practice, very sparse, many entries unknown

2		4	1
5		3	
2	4	5	

Rec Sys Unit Objectives

- Explain Recommendation Task
  - Predict which users will like which items
- Explain two major types of recommendation
  - Content-based (have features for items/users)
  - Collaborative filtering (only have scores for item+user pairs)
    - Detailed Approach: Matrix Factorization + SGD
- Evaluation:
  - Precision/recall for binary recs

Task: Recommendation

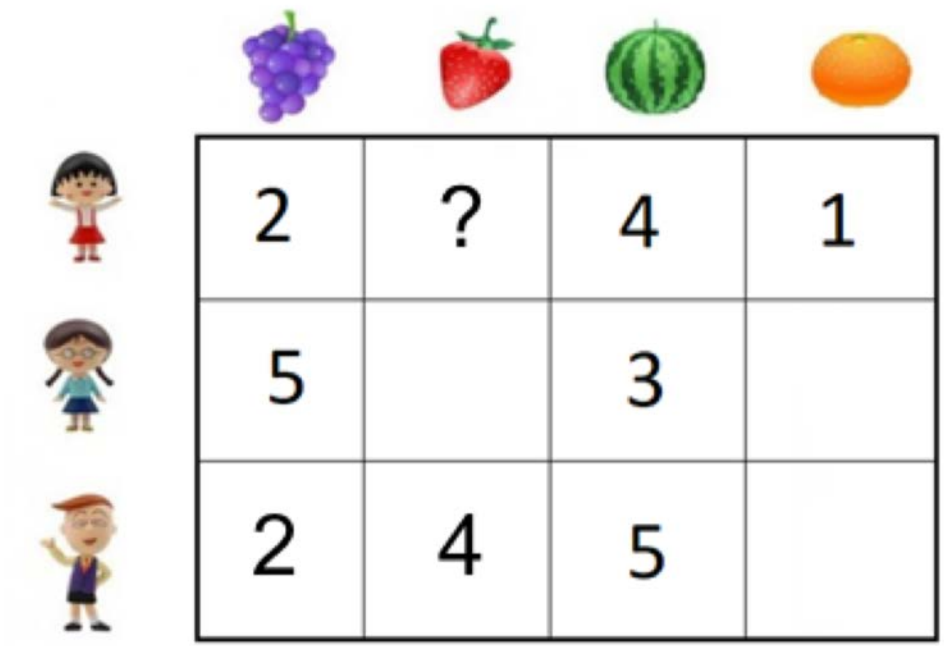
Supervised Learning

Content filtering

Unsupervised Learning

Collaborative filtering

Reinforcement Learning

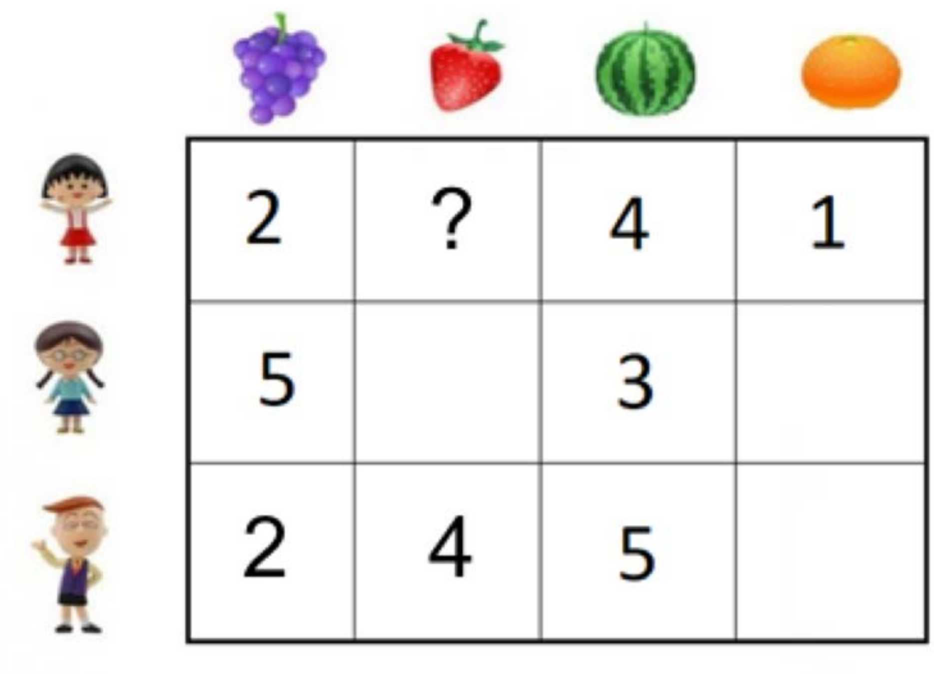


Content-based recommendation

Content-based

Key aspect:
Have common features for each item

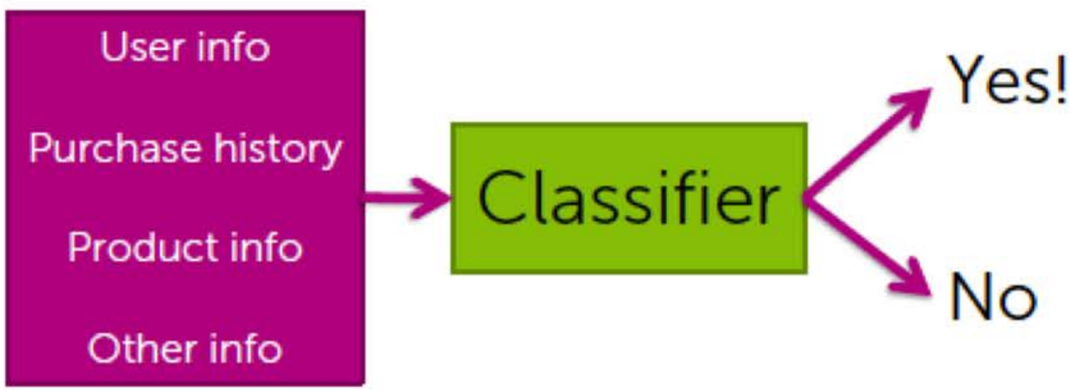
FEATURE	VALUE
is_round	1
is_juicy	1
average_price	$1.99/lb



2	?	4	1
5		3	
2	4	5	

Content-Based Recommendation

- Reduce per-user prediction to supervised prediction problem



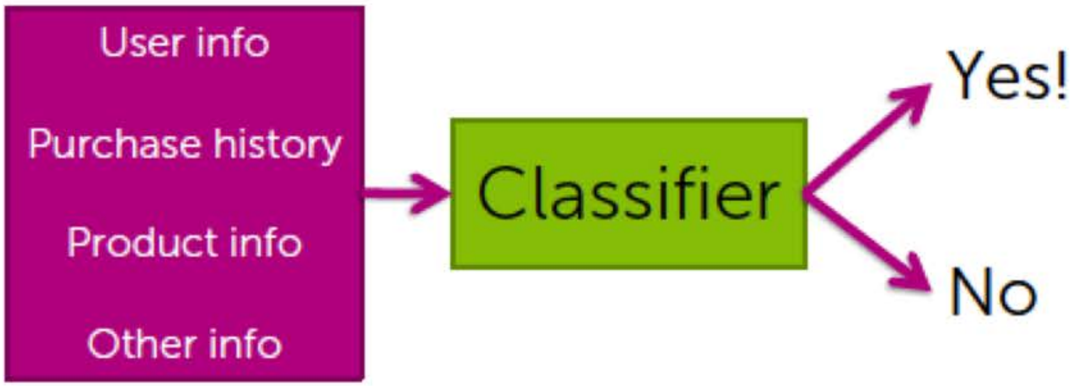
What features are necessary? What are pitfalls?

Fig. Credit: Emily Fox (UW)

Possible Per-Item Features

- Movie
  - Set of actors, director, genre, year
- Document
  - Bag of words, author, genre, citations
- Product
  - Tags, reviews

Content-Based Recommender



Pros:

- Personalized:
  Considers user info & purchase history
- Features can capture context:
  Time of the day, what I just saw,…
- Even handles limited user history:
  Age of user, …

Cons:

- Features may not be available
- Often doesn't perform as well as collaborative filtering methods (next)

Fig. Credit: Emily Fox (UW)

Collaborative filtering

External Slides

- Matt Gormley at CMU

- https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture25-mf.pdf

- We'll use page 4 – 34
  - Start: "Recommender Systems" slide
  - Stop at: comparison of optimization algorithms

Matrix Factorization (MF)

- User i represented by vector $\boldsymbol{u}_i \in R^k$
- Item j represented by vector $\boldsymbol{v}_j \in R^k$
- Inner product $\boldsymbol{u}_i^\top \boldsymbol{v}_j$ approximates the utility M_{ij}
- Intuition:
  - Two items with similar vectors get similar utility scores from the same user;
  - Two users with similar vectors give similar utility scores to the same item

Training an MF model

- Variables to optimize
  - $\mathbf{U} = (\boldsymbol{u}_i : i = 1, \ldots, N), \mathbf{V} = (\boldsymbol{v}_i : j = 1, \ldots, M)$
- Training objective

$$\min_{\mathbf{U},\mathbf{V}} \sum_{ij} \left(M_{ij} - \boldsymbol{u}_i^\top \boldsymbol{v}_j\right)^2 + \lambda \sum_i \|\boldsymbol{u}_i\|_2^2 + \lambda \sum_j \|\boldsymbol{v}_j\|_2^2$$

- How to optimize?
  - Stochastic gradient descent, visit each user-item entry at random!

- Key practical aspects
  - Regularization terms to prevent overfitting

Include intercept/bias terms!

- Per-user scalar s_i
- Per-item scalar t_j

$$\min_{\mathbf{U},\mathbf{V}} \sum_{ij} \left(M_{ij} - \boldsymbol{u}_i^\top \boldsymbol{v}_j - s_i - t_j\right)^2 + \lambda \sum_i \|\boldsymbol{u}_i\|_2^2 + \lambda \sum_j \|\boldsymbol{v}_j\|_2^2$$

Why include these?

Include intercept/bias terms!

- Per-user scalar s_i
- Per-item scalar t_j

$$\min_{\mathbf{U},\mathbf{V}} \sum_{ij} \left(M_{ij} - \boldsymbol{u}_i^\top \boldsymbol{v}_j - s_i - t_j\right)^2 + \lambda \sum_i \|\boldsymbol{u}_i\|_2^2 + \lambda \sum_j \|\boldsymbol{v}_j\|_2^2$$

Why include these? **Improve accuracy**

Some items just more popular

Some users just more positive

Summary of Methods

Task: Recommendation

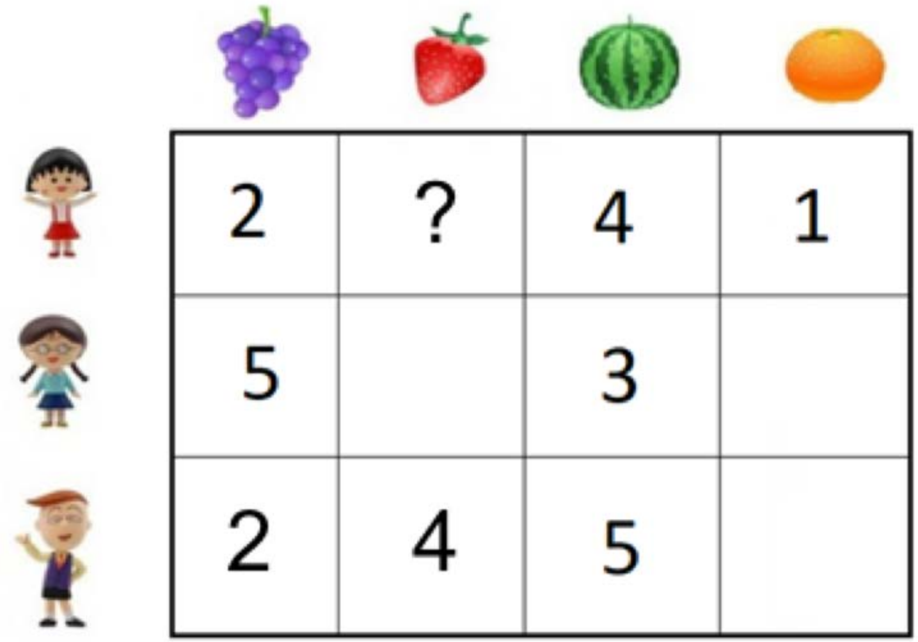
Supervised Learning

Content-based filtering

Unsupervised Learning

Collaborative filtering

Reinforcement Learning

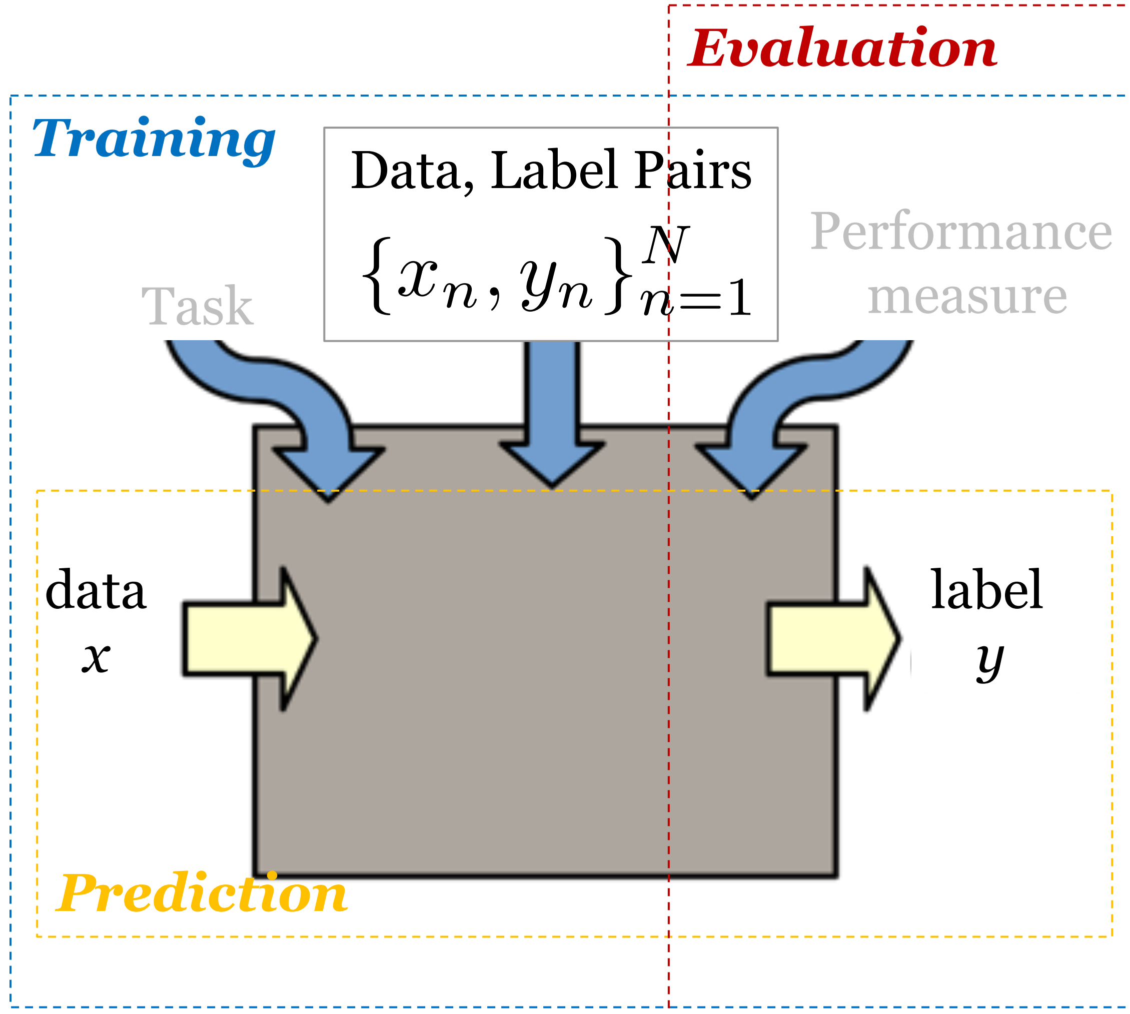


Recall: Supervised Method

Supervised Learning

Unsupervised Learning

Reinforcement Learning



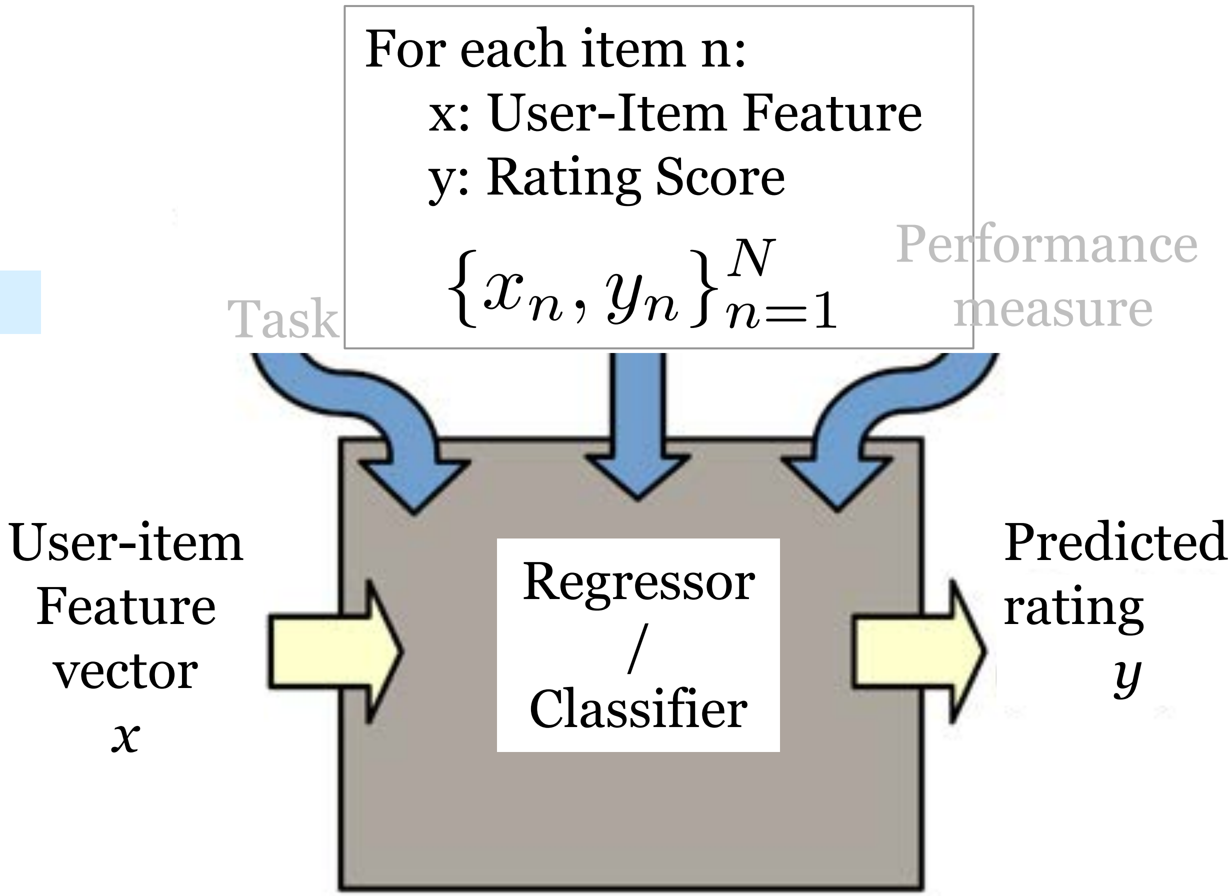
Example: Per-User Predictor

Supervised Learning

Content-based filtering

Unsupervised Learning

Reinforcement Learning

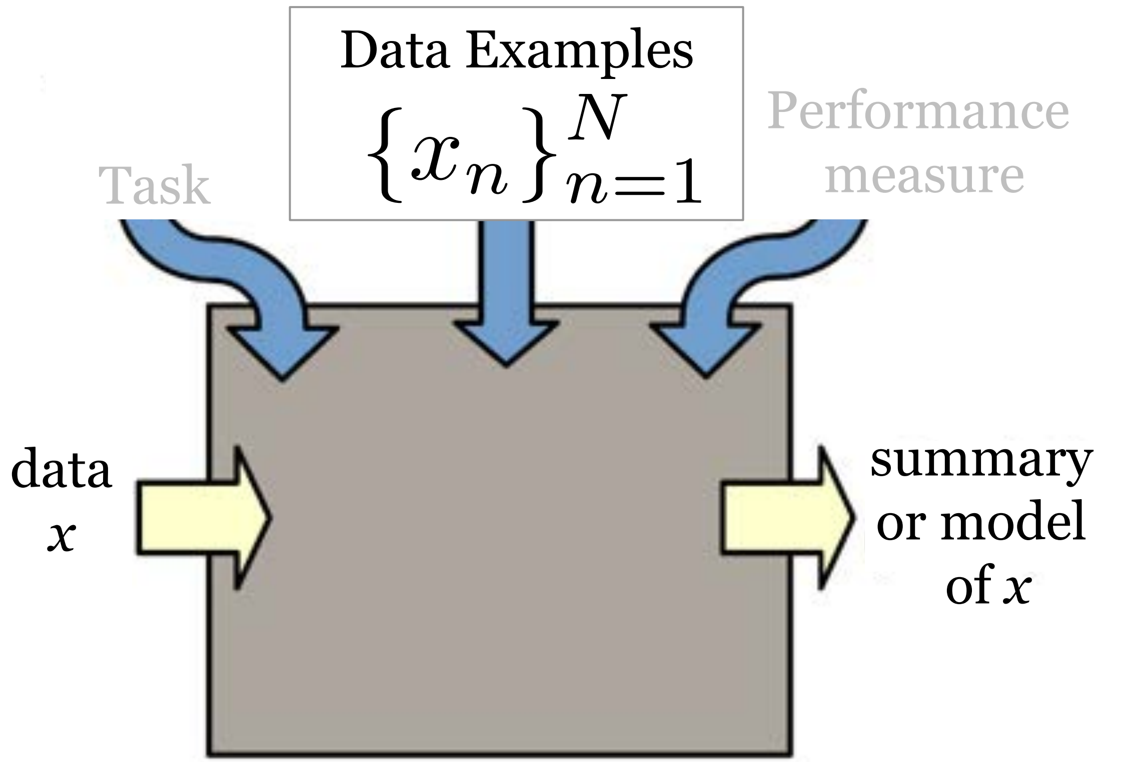


Recall: Unsupervised Method

Supervised Learning

Unsupervised Learning

Reinforcement Learning



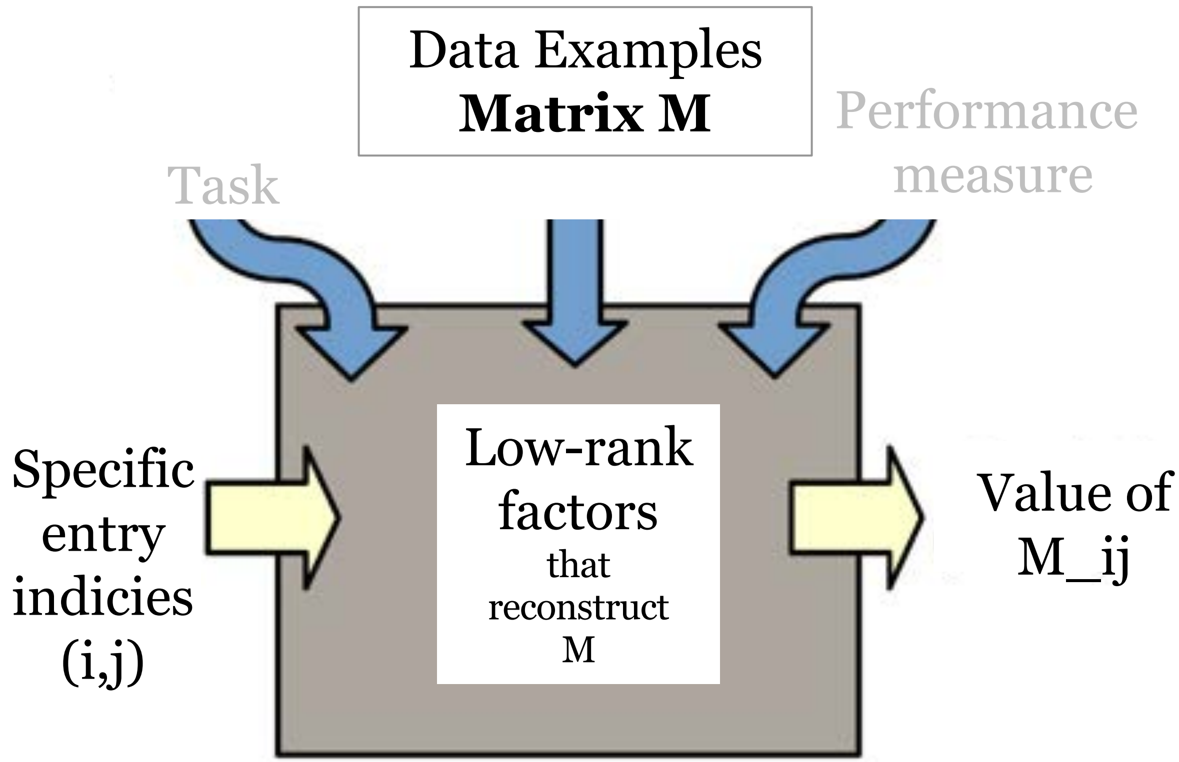
Example: Matrix Factorization

Supervised Learning

Unsupervised Learning

Collaborative filtering

Reinforcement Learning



Evaluation

Evaluation

Assumptions

- For given user, we can rate each item with score
- We care most about our top-score predictions
- Setup:
  - Algorithm rates each item with score
  - Sort items from high to low score
  - Have "true" relevant/not usage labels (unused by algo.)

Item ranking	1	2	3	4	5	6	7	8
Actual usage	1	0	1	0	0	0	1	1

How many liked items were recommended?

Recall

$$\text{Recall} = \frac{\#\ \text{liked \& shown}}{\#\ \text{liked}}$$

How many recommended items were liked?

Precision

$$\frac{\text{\# liked \& shown}}{\text{\# shown}}$$

External Slides

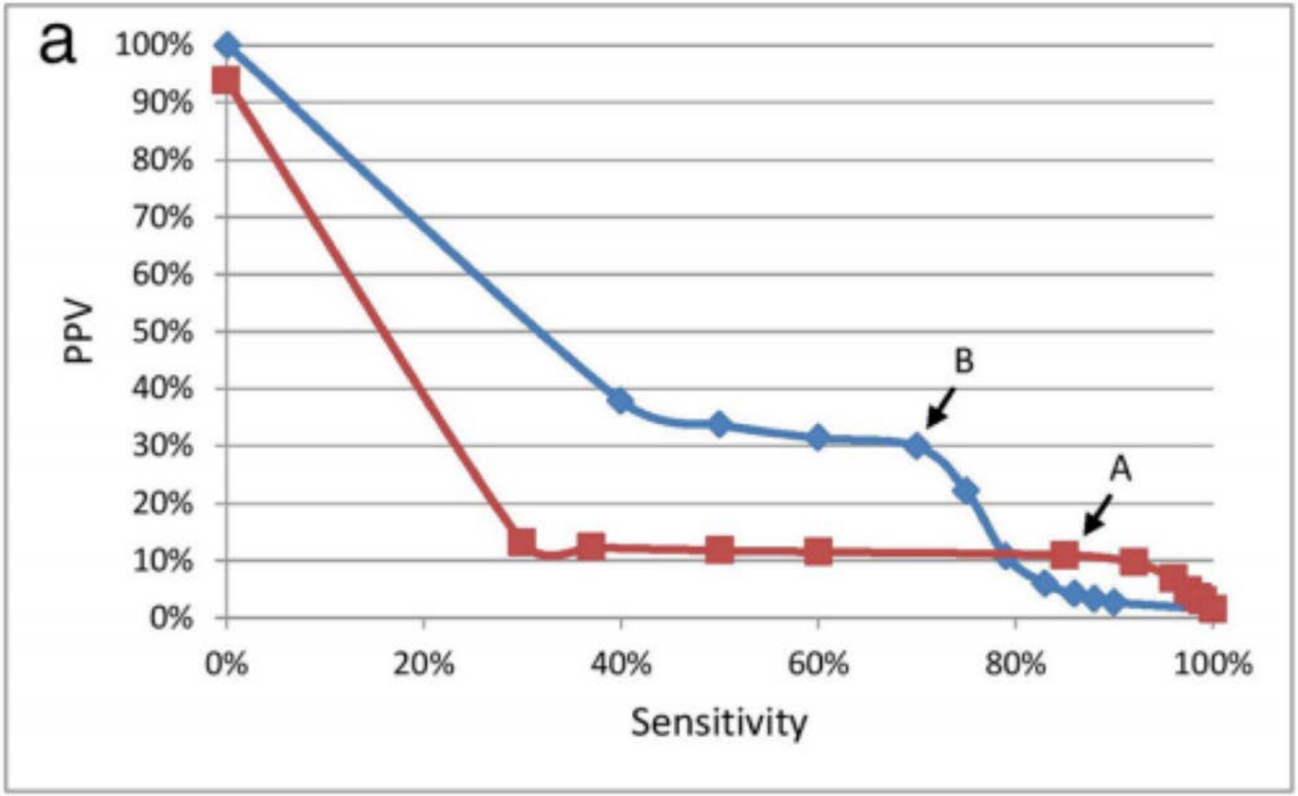
- Emily Fox's slides

- https://courses.cs.washington.edu/courses/cse416/18sp/slides/L13_matrix-factorization.pdf#page=19

- Start: Slide 19 (world of all baby products)
- Stop: End of that section

Precision-Recall Curve

precision



recall (= TPR)

Precision @ k

- Assume only top k results are predicted positive
  - E.g. Netflix can only show you ~8 results on screen at a time, we want most of these to be relevant
- Example:

Item ranking	1	2	3	4	5	6	7	8
Actual usage	1	0	1	0	0	0	1	1

- Prec @ 1 : 100%
- Prec @ 2: 50%
- Prec @ 8: 50%

Cold Start Issue

- New user entering the system
  - Hard for both content-based and matrix factors
  - Matching similar users
  - Trial-and-error
- New item entering the system
  - Easy with per-user content-based recommendation
    - IF easy to get the item's feature vector
  - Hard with matrix factorization
    - Trial-and-error

Practical Issues in Real Systems

- Recommendation system and users form a loopy system
  - RS changes user's behavior
  - User generate data for RS
- User groups becoming more homogeneous
  - Youtube recommendation of politic videos: recommend videos from the same camp

Rec Sys Unit Objectives

- Explain Recommendation Task
    - Predict which users will like which items
- Explain two major types of recommendation
    - Content-based (have features for items/users)
    - Collaborative filtering (only have scores for item+user pairs)
        - Detailed Approach: Matrix Factorization + SGD
- Evaluation:
    - Precision/recall for binary recs