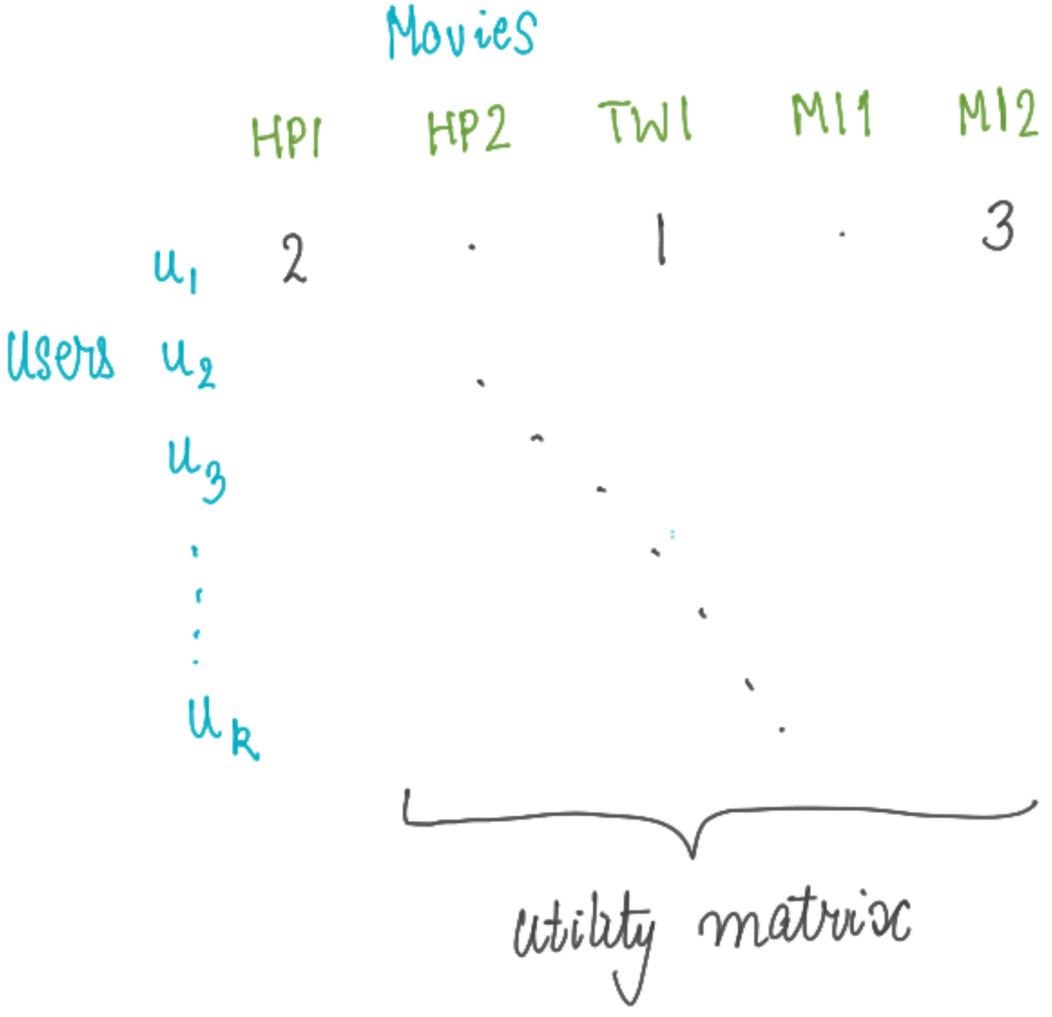
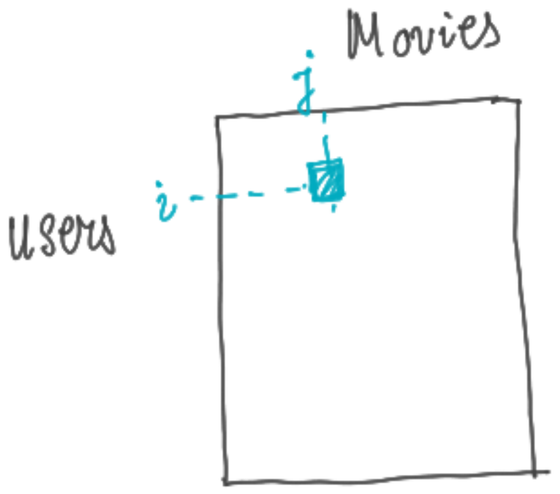


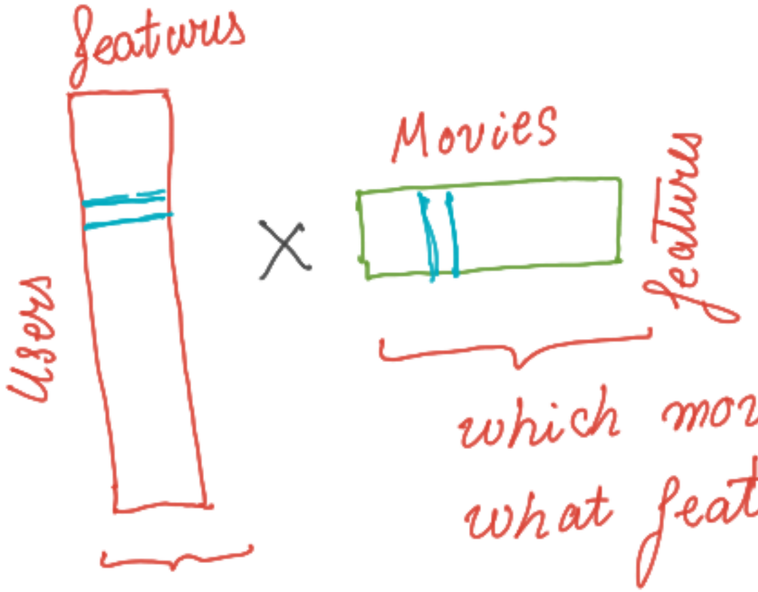
Movies	Fantasy	Action
HP1	0.9	0.1
HP2	0.87	0.03
TW	0.5	0.4
MI1	0.01	0.9
MI2	0.01	0.92

features

Users	Fantasy	Action
u_1	1.4	2
u_2	0.3	3
u_3	⋮	⋮
⋮	⋮	⋮
⋮	⋮	⋮



$\approx$



which user likes which feature & by how much

which movie has what features

	fantasy	action
i	0.5	1.7

	M14
fantasy	0.1
action	2

$$\begin{bmatrix} 0.5 & 1.7 \end{bmatrix} \begin{bmatrix} 0.1 \\ 2 \end{bmatrix} = 3.45$$

$U \approx P Q^T$

We have to find P & Q^T such that the product PQ^T is very close to U.

loss function (reconstruction loss)

$U = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ $\quad Q^T = [-1 \quad 2]$ $\quad P = \begin{bmatrix} 2 \\ -1 \end{bmatrix}$

$$PQ^T = \begin{bmatrix} 2 \\ -1 \end{bmatrix} [-1 \quad 2] = \begin{bmatrix} -2 & 4 \\ 1 & -2 \end{bmatrix}$$

How close is U to PQ^T?

↳ 55

→ distance → L2 norm of $|U - PQ^T|$

$$|U - PQ^T| = \begin{bmatrix} 3 & -2 \\ 2 & 6 \end{bmatrix}$$

$$\|U - PQ^T\|_2^2 = 3^2 + (-2)^2 + (2)^2 + (6)^2 = 55.$$

I want to find P & Q

→ that reproduces U as $PQ^T \approx U$ } same thing.

→ that minimizes $||U - PQ^T||_2^2$

l ; Recons. error

$\frac{\partial l}{\partial P} = 0 \quad ; \frac{\partial l}{\partial Q} = 0$ } gradient descent to find P & Q

↳ local minima

↳ uneven surface, long time to converge.

$P \rightarrow P + \gamma(\cdot\cdot)$ ✓
$Q \rightarrow Q + \gamma(\cdot\cdot)$ ✓ ALS algo.

→ P & Q can be found by decomposing U using SVD

$U \approx S \Sigma V^T$ →

$U \approx S \Sigma V^T$ → using SVD

$U \approx P Q^T$ → what we want.

$P \leftarrow S$

$Q^T \leftarrow \Sigma V^T$

Result:

SVD minimizes reconstruction error!

↳ $S \rightarrow P$ & $\Sigma V^T \rightarrow Q^T$ are the best values for P & Q^T

choice of # of features to be done wisely

↳ domain knowledge

↳ Recons. error

$$l = \sum_i (y_i - w^T x_i)^2$$

Find w^T that minimizes l

$$\frac{\partial l}{\partial w^T} = 0 \rightarrow \text{X closed form soln}$$

↓

gradient descent.

Gradient Descent

choose random w

$$w_{t+1} \rightarrow w_t - \gamma \left(\frac{\partial l}{\partial w^T} \cdot .\right)$$ ↓ till convergence

w^* ↙

$l = \sum (y_i - w^T x_i)^2$

Find w that minimizes l

$$\boxed{w = (X^T X)^{-1} X^T y}$$ ✓✓

[ ] coeffs or parameters

$$X = \begin{bmatrix} x_1 \dashrightarrow \\ x_2 \longrightarrow \\ \vdots \end{bmatrix}$$ ✓

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \end{bmatrix}$$ ✓

Normal Equations

$$y = ax^2 + bx + c$$

$$\max / \min (y) \left\{ x = \left(\frac{-b}{2a} \right) \equiv x \right\}$$ ✓✓

$U_{1000 \times 10000}$ $\approx$ $P_{1000 \times 10}$ $\times$ $Q^T_{10 \times 10000}$

$U_{1000 \times 10000}$ (users × items)

$10^3 \times 10^4$

$= 10^7$ nos to store.

$P_{1000 \times 10}$ (users × features)

$10^3 \times 10$

$= 10^4$ numbers

$Q^T_{10 \times 10000}$ (features × items)

10×10^4

$= 10^5$ numbers.

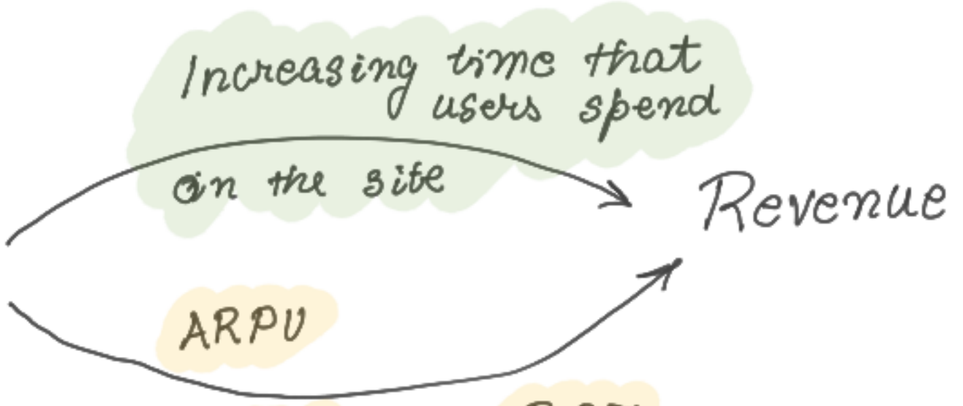
$+$

$\sim 10^5$ numbers.

100 x reduction in storage if we use LF models.

Evaluating RecSys

- Users
- Platform



How much value does the addition of recommender systems add to the platform?

Users' way of evaluating RecSys

- accuracy metrics
  - Top N acc / precision / recall → $\frac{TP}{TP+FN}$
    - precision → $TP/TP+FP$ → Amazon (items >> users)
    - Netflix (users >> items)

Pred. \ Actual	Present	Absent
Present	TP	FP
Absent	FN	TN

- novelty
- diversity

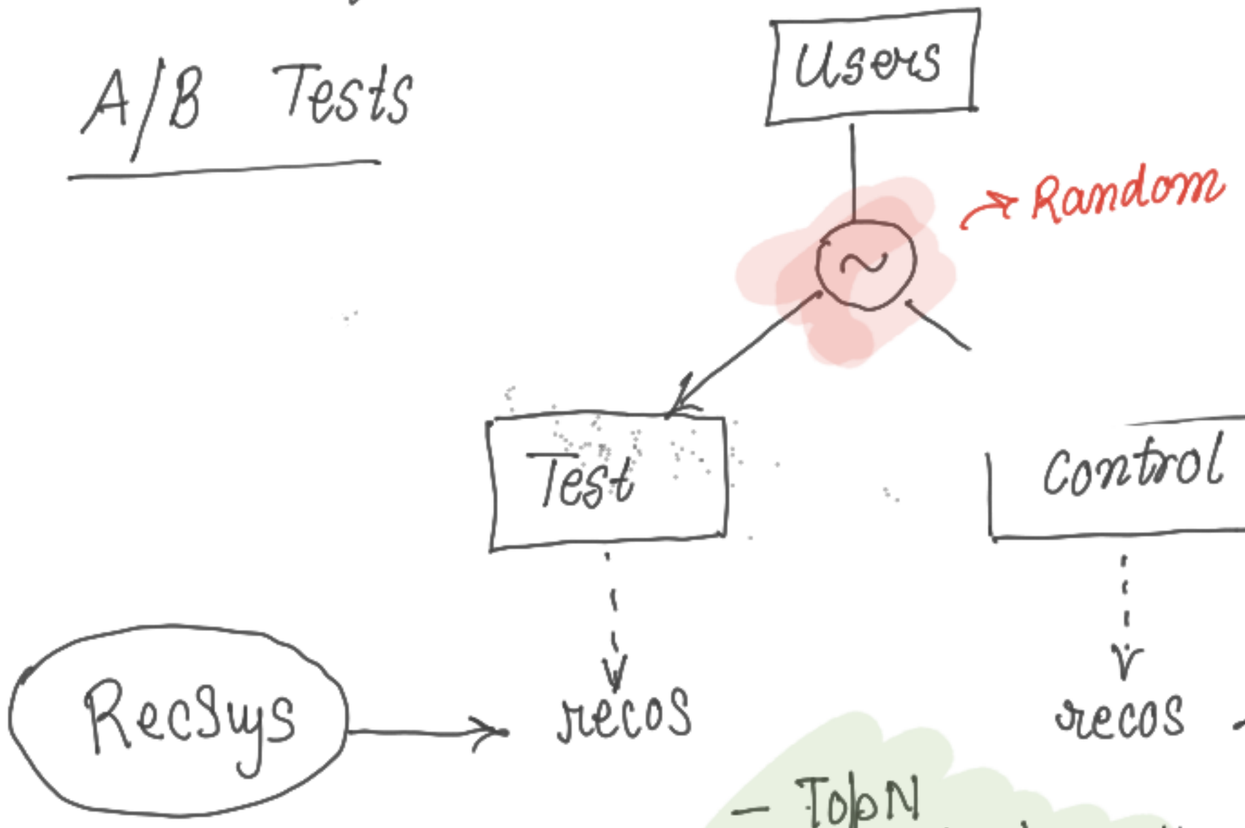
- ranking metrics
  - Reciprocal Rank

Pred: [a, b, c, d, e]

Actl: [c] → $RR = \frac{1}{3}$

Online Eval. of RecSys

A/B Tests



- TopN acc | pr | recall
- RR | CDG | nCDG
- Time spent | ARPU | CLTV

~ same usr numbers (N)

	T	C	t-test
acc	a_1	a_2	$(a_1 - a_2)$
pr	p_1	p_2	$p_1 - p_2$
recall	r_1	r_2	$r_1 - r_2$
rr	rr_1	rr_2	$rr_1 - rr_2$

pairwise t-test

TF - IDF

Term frequency (TF) : # of times a term w_j occurs in all docs $\underbrace{D_1 \cdots D_N}_{N\text{-docs}}$

Inverse doc frequency (IDF) : $\log\left(\frac{N}{\#\text{ of docs the term } w_j \text{ is present}}\right)$

$$TF\text{-}IDF = TF \times IDF$$

for unique words in docs ; TF - IDF ↑

for common words in all docs ; TF - IDF ↓