CS276B

Text Information Retrieval, Mining, and Exploitation

Lecture 11

Feb 20, 2003

From the last lecture

- Recommendation systems
- What they are and what they do
- A couple of algorithms
- Going beyond simple behavior: context
- How do you measure them?
- Begin: how do you design them "optimally"?
  - Introduced utility formulation

Today's topics

- "Clean-up" details from last time
  - Implementation
  - Extensions
  - Privacy
  - Network formulations
- Recap utility formulation
- Matrix reconstruction for low-rank matrices
- Compensation for recommendation

Implementation details

- Don't really want to maintain this gigantic (and sparse) vector space
- Dimension reduction
- Fast near neighbors
- Incremental versions
  - update as new transactions arrive
  - typically done in batch mode
  - incremental dimension reduction etc.

Extensions

- Amazon - "why was I recommended this"
  - see where the "evidence" came from
- Clickstreams - do sequences matter?
- HMM's to infer user type from browse sequence
  - e.g., how likely is the user to make a purchase?
  - Meager improvement in using sequence

Privacy

- What info does a recommendation leak?
  - E.g., you're looking for illicit content and it shows me as an expert
- What about compositions of recommendations?
- "These films are popular among your colleagues"
- "People who bought this book in your dept also bought … "
  - "Aggregates" are not good enough
- Poorly understood

Network formulations

- Social network theory
  - Graph of acquaintanceship between people
  - Six degrees of separation, etc.
- Consider broader social network of people, documents, terms, etc.
  - Links between docs a special case

Network formulations

- Instead of viewing users/items in a vector space
- Use a graph for capturing their interactions
- Users with similar ratings on many products are joined by a "strong" edge
  - Similarly for items, etc.

Recommendation from networks

- Look for docs near a user in the graph
  - "horting"
- What good does this do us?
- (In fact, we've already invoked such ideas in the previous lecture, connecting it to Hubs/Auths)

Network formulations

- Advantages
  - Can use graph-theoretic ideas
    - E.g., similarity of two users based on proximity in graph
    - Even if they've rated no items in common
    - Good for intuition
- Disadvantages
  - With many rating transactions, edges build up
  - Graph becomes unwieldy representation
  - E.g., triangle inequality doesn't hold
    - No implicit connections between entities
  - should two items become "closer" simply because one user rates them both similarly?

Vector vs. network formulations

- Some advantages – e.g., proximity between users with no common ratings – can be engineered in a vector space
  - Use SVD's, vector space clustering
- Network formulations are good for intuition
  - Questionable for implementation
  - Good starting point then implement with linear algebra – as we did in link analysis

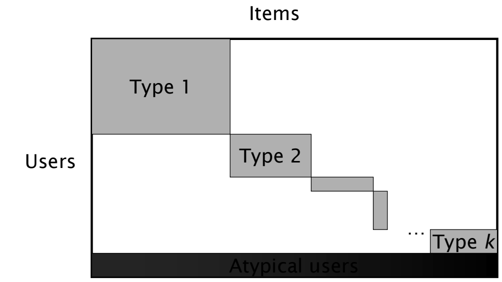
Measuring recommendations: Recall utility formulation

- $m \times n$ matrix U of utilities for each of m users for each of n items: U_{ij}
  - not all utilities known in advance
  - (which ones *do* we know?)
- Predict which (unseen) utilities are highest for each user i

User types

- If users are arbitrary, all bets are off
  - Assume matrix U is of low rank
  - a constant k independent of m,n
- I.e., users belong to k well-separated types
  - (almost)
  - Most users' utility vectors are close to one of k well-separated vectors

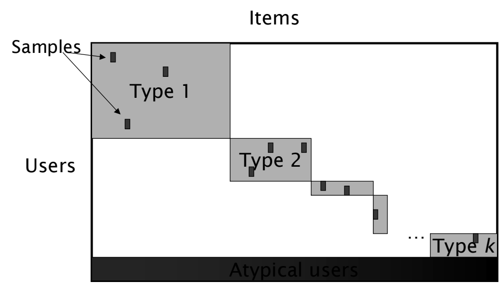
Intuitive picture (exaggerated)



Matrix reconstruction

- Given some utilities from the matrix
- Reconstruct missing entries
  - Suffices to predict biggest missing entries for each user
  - Suffices to predict (close to) the biggest
  - For most users
    - Not the atypical ones

Intuitive picture



Matrix reconstruction:
Achlioptas/McSherry

- Let $\hat{U}$ be obtained from U by the following sampling: for each i,j
  - $\hat{U}_{ij} = U_{ij}$, with probability $1/s$,
  - $\hat{U}_{ij} = 0$ with probability $1-1/s$.
- The sampling parameter s has some technical conditions, but think of it as a constant like 100.
- Interpretation: $\hat{U}$ is the sample of user utilities that we've managed to get our hands on
  - From past transactions
  - (that's a lot of samples)

How do we reconstruct U from $\hat{U}$?

- First the "succinct" way
  - then the (equivalent) intuition
- Find the best rank k approximation to $s\hat{U}$
  - Use SVD (best by what measure?)
  - Call this $\hat{U}_k$
- Output $\hat{U}_k$ as the reconstruction of U
  - Pick off top elements of each row as recommendations, etc

Achlioptas/McSherry theorem

- With <u>high probability</u>, reconstruction <u>error</u> is small
  - see paper for detailed statement
- What's high probability?
  - Over the *samples*
  - <u>not</u> the matrix entries
- What's error – how do you measure it?

Norms of matrices

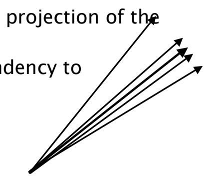
- Frobenius norm of a matrix M:
  - $|M|_F^2$ = sum of the square of the entries of M
- Let M_k be the rank k approximation computed by the SVD
- Then for any other rank k matrix X, we know
  - $|M- M_k|_F \leq |M-X|_F$
- Thus, the SVD gives the best rank k approximation for each k

Norms of matrices

- The L_2 norm is defined as
  - $|M|_2 = \max |Mx|$, taken over all unit vectors x
- Then for any other rank k matrix X, we know
  - $|M- M_k|_2 \leq |M-X|_2$
- Thus, the SVD also gives the best rank k approximation by the L_2 norm
- What is it doing in the process?
  - Will avoid using the language of eigenvectors and eigenvalues

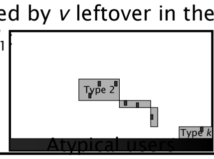
What is the SVD doing?

- Consider the vector v defining the L_2 norm of U:
  - $|U|_2 = |Uv|$
- Then v measures the "dominant vector direction" amongst the rows of U (i.e., users)
  - ith coordinate of Uv is the projection of the ith user onto v
  - $|U|_2 = |Uv|$ captures the tendency to align with v



What is the SVD doing, contd.

- U_1 (the rank 1 approximation to U) is given by Uvv^T
- If all rows of U are collinear, i.e., rank(U)=1, then U= U_1 ;
  - the error of approximating U by U_1 is zero
- In general of course there are still user types not captured by v leftover in the residual matrix U-U_1



Iterating to get other user types

- Now repeat the above process with the residual matrix U-U_1
- Find the dominant user type in U-U_1 etc.
  - Gives us a second user type etc.
- Iterating, get successive approximations U_2, U_3, … U_k

Achlioptas/McSherry again

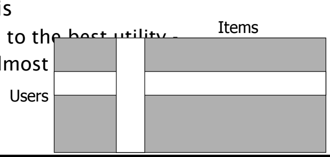
- SVD of $\hat{U}$: the uniformly sampled version of U
- Find the rank k SVD of $\hat{U}$
- The result $\hat{U}_k$ is close to the best rank k approximation to U
- Is it reasonable to sample uniformly?
  - Probably not
  - E.g., unlikely to know much about your fragrance preferences if you're a sports fan

Variants – Drineas et al.

- Good Frobenius norm approximations give nearly-highest utility recommendations
  - Net utility to user base close to optimal
- Provided most users near k well-separated prototypes, simple sampling algorithm
- Sample an element of U in proportion to its value
  - i.e., system more likely to know my opinions about my high-utility items

Drineas et al.

- Pick O(k) items and get all m users' opinions
  - → marketing survey
- Get opinions of ~k ln k random users on all n items
  - → guinea pigs
- Give a recommendation to each user that w.h.p. is
  - close to the best utility
  - for almost



Compensation

- How do we motivate individuals to participate in a recommendation system?
- Who benefits, anyway?
- E.g., eCommerce: should the system work for the benefit of
  - (a) the end-user, or
  - (b) the website?

End-user vs. website

- End-user measures recommendation system by utility of recommendations
  - Our formulation for this lecture so far
  - Applicable even in non-commerce settings
- But for a commerce website, different motivations
  - Utility measured by purchases that result
  - What fraction of recommendations lead to purchases?
  - What is the average "upsell" amount?

End-user vs. website

- Why should an end-user offer opinions to help a commerce site?
- Is there a way to compensate the end-user for the net contribution from their opinions?
- How much?

Coalitional games

Game with players in [*n*].

$v(S)$ = the maximum total payoff of all players in S, *under worst case play by* $[n] - S$.

How do we split $v([n])$*?*

For example …

- How should A, B, C split the loot (=20)?
- We are given what each subset can achieve by itself as a function *v* from the powerset of {A,B,C} to the reals.
- $v(\{\}) = 0$.

Values of *v*

- A: 10
- B: 0
- C: 6
- AB: 14
- BC: 9
- AC: 16
- ABC: 20

First notion of "fairness": Core

A vector $(x_1, x_2, \ldots, x_n)$ with $\Sigma_i\, x_i = v([n])$ (= 20) is in the core if for all S, we have $x[S] \geq v(S)$.

In our example: A gets 11, B gets 3, C gets 6.

Problem: Core is often empty (e.g., if v[AB]=15).

Second idea: Shapley value

$$x_i = E_\pi(v[\{j: \pi(j) \leq \pi(i)\}] - v[\{j: \pi(j) < \pi(i)\}])$$

(Meaning: Assume that the players arrive at random. Pay each one his/her incremental contribution at the moment of arrival. Average over all possible orders of arrival.)

Theorem [Shapley]: The Shapley value is the only allocation that satisfies Shapley's axioms.

In our example…

- A gets:

10/3 + 14/6 + 10/6 + 11/3 = 11

- B gets:

0/3 + 4/6+ 3/6 +4/3 = 2.5

- C gets the rest = 6.5

Values of *v*

- A: 10
- B: 0
- C: 6
- AB: 14
- BC: 9
- AC: 16
- ABC: 20

e.g., the UN security council

- 5 permanent, 10 non-permanent members
- A resolution passes if voted by a majority of the 15, including all 5 P
- $v[S] = 1$ if $|S| > 7$ *and* S contains 1,2,3,4,5; otherwise 0
- What is the Shapley value (~power) of each P member? Of each NP member?

e.g., the UN security council

- What is the probability, when you are the 8^{th} arrival, that all of 1,…,5 have arrived?
- Calculation:

  Non-Permanent members ~ .7%

  Permanent members: ~ 18.5%

Notions of fairness

third idea: bargaining set
fourth idea: nucleolus
.
.
.
seventeenth idea: the von Neumann-Morgenstern solution

Privacy and recommendation systems

- View privacy as an economic commodity.
  - Surrendering private information is measurably good or bad for you
  - Private information is intellectual property controlled by others, often bearing negative royalty
- Proposal: *evaluate/compensate the individual's contribution when using personal data for decision-making.*

Compensating recommendations

- Each user likes/dislikes a set of items (user is a vector of 0, ±1)
- The "similarity" of two users is the inner product of their vectors
- We have k "well separated types": ±1 vectors
  - each user is a *random perturbation* of a particular type
- Past purchases a random sample for each user

Compensating recommendations

- A user gets advice on an item from the k nearest neighbors
- Value of this advice is ±1
  - +1 if the advice agrees with actual preference, else -1
- How should agents be compensated (or charged) for their participation?

Compensating recommendations

Theorem: A user's compensation (= value to the community) *is an increasing function of how typical* (close to his/her type) *the user is.*

In other words, the closer we are to our (stereo)type, the more valuable we are and the more we get compensated.

Resources

- Achlioptas McSherry
  - http://citeseer.nj.nec.com/462560.html
- Azar et al
  - http://citeseer.nj.nec.com/azar00spectral.html
- Aggarwal et al - Horting
  - http://citeseer.nj.nec.com/aggarwal99horting.html
- Drineas et al
  - http://portal.acm.org/citation.cfm?doid=509907.509922
- Coalitional games