Recommender Systems

Chapter

proschool
An ims Initiative

What are recommender systems?

- A recommender system or a recommendation system is a subclass of information filtering system that seeks to predict the “rating” or “preference” a user would give to an item. – Wikipedia

- Put in simple words, they are computer systems which “recommend” stuff based on specific patterns/trends in the users/customers.

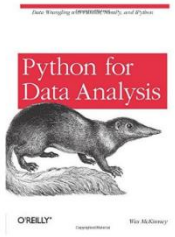
- Popular companies which use recommender systems – Amazon, Flipkart, Netflix, Facebook, Amazon Prime

Example of recommender systems

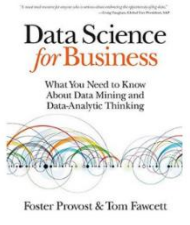
Customers Who Bought This Item Also Bought

Page 1 of 15

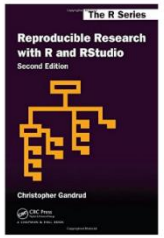
Data Science from Scratch: First Principles with Python
› Joel Grus
4.4 out of 5 stars 54
#1 Best Seller in Data Mining
Paperback
$33.99 Prime



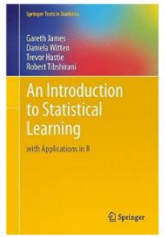
Python for Data Analysis: Data Wrangling with Pandas, NumPy, and…
› Wes McKinney
4.4 out of 5 stars 118
Paperback
$27.68 Prime



Data Science for Business: What You Need to Know about Data Mining and…
› Foster Provost
4.5 out of 5 stars 135
Paperback
$37.99 Prime

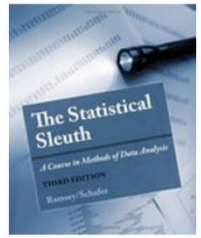


Reproducible Research with R and R Studio, Second Edition…
Christopher Gandrud
4.5 out of 5 stars 3
Paperback
$51.97 Prime



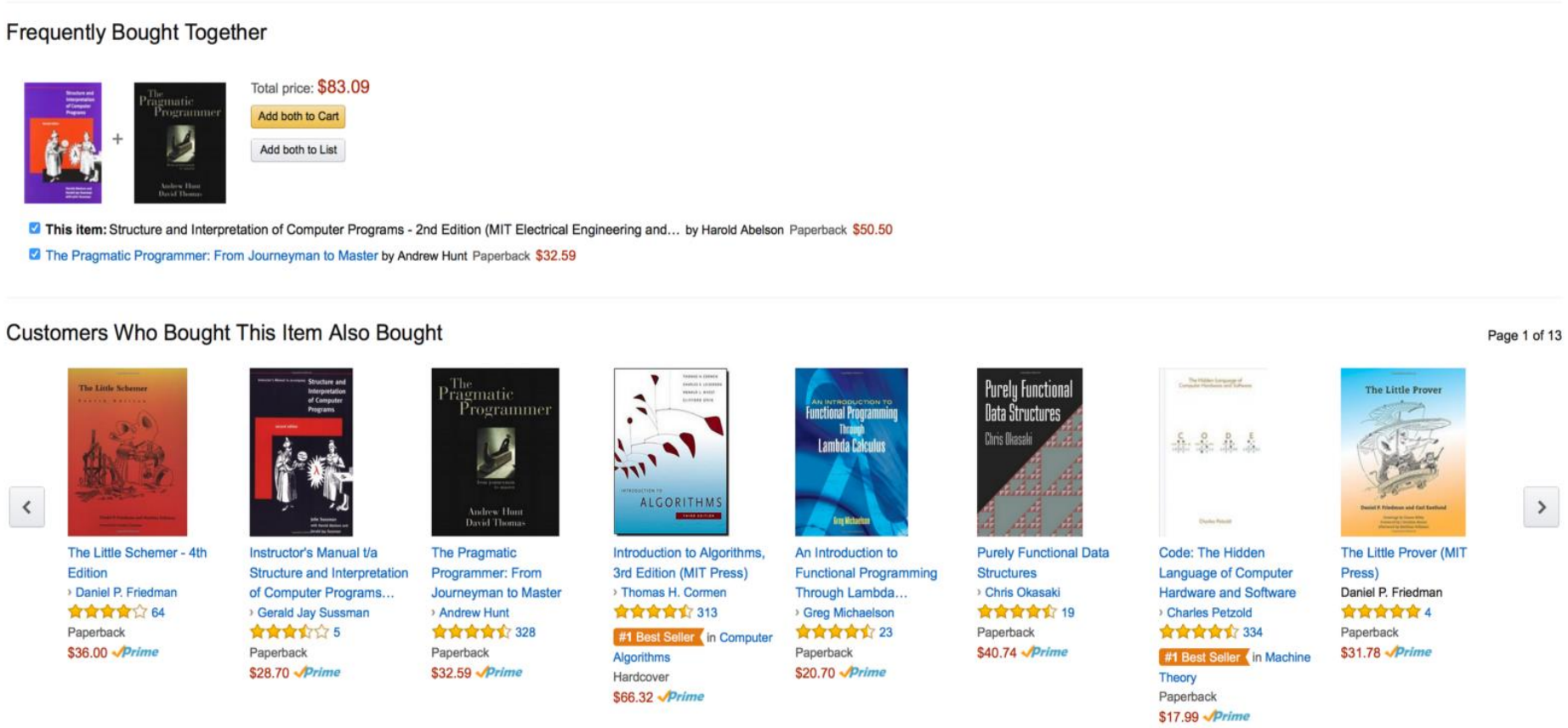
An Introduction to Statistical Learning: with Applications in R…
› Gareth James
4.9 out of 5 stars 105
Hardcover
$68.35 Prime

Data Smart: Using Data Science to Transform Information into Insight
› John W. Foreman
4.5 out of 5 stars 99
#1 Best Seller in Computer Simulation
Paperback
$28.16 Prime



The Statistical Sleuth: A Course in Methods of Data Analysis
Fred Ramsey
4.4 out of 5 stars 6
Hardcover
$284.42 Prime

Example of recommender systems



Types of recommendation systems

- Collaborative filtering
- Content based filtering

Collaborative Filtering

- This type of filter is based on users' ratings on products (movies, items, etc)
- Taking example of movies, we compare two users with the movies they rated.
- If the two users rated movies in similar ways, i.e. they both like thriller and adventure, and if one user has watched 3 movies and the other has watched 2 movies, we will recommend the 3rd movie to the user.

Collaborative Filtering - Example

- Following people like or dislike the movies

	Incredibles	Despicable me	Avengers	Iron Man
Kumar	Yes	Yes	Yes	Yes
Kishan	No	Yes	No	Yes
Rohan	Yes	No	Yes	No
Rahul	No	No	Yes	Yes

Collaborative Filtering - Example

- There is a new user Ashok, who likes **Incredibles**.

	Incredibles	Despicable me	Avengers	Iron Man
Kumar	Yes	Yes	Yes	Yes
Kishan	No	Yes	No	Yes
Rohan	Yes	No	Yes	No
Rahul	No	No	Yes	Yes
Ashok	Yes	?	?	?

Collaborative Filtering - Example

- Kumar and Rohan also like **Incredibles**.
- Among Kumar and Rohan, they **both like Avengers**.
- Thus, Ashok will be recommended **Avengers**.

	Incredibles	Despicable me	Avengers	Iron Man
Kumar	Yes	Yes	Yes	Yes
Kishan	No	Yes	No	Yes
Rohan	Yes	No	Yes	No
Rahul	No	No	Yes	Yes
Ashok	Yes	?	?	?
	Matching Yes	1 Yes	2 Yes	1 Yes

Content based Filtering

- This type of filter does not involve multiple users, only a single user.
- Based on the user's rating/liking, we will recommend similar products.
- Example
  - If user likes Levi's jeans blue colour, we will recommend other brands which will have blue colour jeans.
  - If user likes Amitabh Bachchan movies, we will recommend movies of Amitabh Bachchan which the user has not watched yet.

Similarity

- Techniques used to find similarity between vectors (users, ratings, items, etc)
- This is the actual technique being used. This can be used in both content based and collaborative filtering.
- Types
  - Cosine similarity
  - Correlation similarity
  - Matrix factorization

Cosine Similarity

- Cosine Similarity is the measure used to find out if two vectors are similar or not.
- It is defined by the following formula

$$\text{similarity} = \cos\theta = \frac{A \cdot B}{|A| * |B|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

- Range of similarity
  - -1 means exactly opposite
  - 1 means exactly same
  - 0 means orthogonal

Cosine Similarity

- Example
  - Ram loves Sita more than Shyam loves Sita
  - Rahul likes Sita more than Ram loves Sita

- List of words from both texts
  - Ram loves Sita more than Shyam Rahul likes

Ram	Loves	Sita	More	Than	Shyam	Rahul	Likes
1	2	2	1	1	1	0	0
1	1	2	1	1	0	1	1

Cosine Similarity

- We will decide how close the above example texts are similar to each other by finding the cosine angle between them

- Rewriting the above example as vectors
  - A: [1, 2, 2, 1, 1, 1, 0, 0]
  - B: [1, 1, 2, 1, 1, 0, 1, 1]

- Using the formula

$$\frac{(1*1+2*1+2*2+1*1+1*1+1*0+0*1+0*1)}{\sqrt{(1^2+2^2+2^2+1^2+1^2+1^2+0^2+0^2)}*\sqrt{(1^2+1^2+2^2+1^2+1^2+0^2+1^2+1^2)}}$$

$$= 0.82158$$

- Which means that the two sentences are very similar.

- Similar process is followed for computing cosine similarity for a matrix containing vectors, eg. User vs movies with ratings.

Correlation based similarity

- Also called pearson correlation similarity
- It measures the correlation between two vectors (users/items/etc) to find how similar they are.
- It is defined by the following formula

$$\rho(X,Y) = corr(X,Y) = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{\sum_{i=1}^{n}(Xi - \bar{X})(Yi - \bar{Y})}{\sigma_X \sigma_Y}$$

- Range of similarity
  - -1 means exactly dissimilar
  - 1 means exactly similar
  - 0 means neither similar not dissimilar

Correlation based similarity

	Item1	Item2	Item3	Item4	Item5	Item6	Item7
Kumar	4	5	3	5	1		
Rahul	5	5	4		2		
Ram				1	5	3	5

- Given above is an example of users having bought items at an online store.
- Each of them have rated on a scale of 1-5 for the items they have bought.
- The items which are rated blank, are ones which have not been bought by the user.
- We will now try to recommend Rahul a suitable item, based on his previous purchases.

Correlation based similarity

	Item1	Item2	Item3	Item4	Item5	Item6	Item7
Kumar	4	5	3	5	1		
Rahul	5	5	4		2		
Ram				1	5	3	5

- We see that Kumar and Rahul might look similar. Let's calculate properly and find out.

- Correlation similarity calculation between Kumar and Rahul is as follows

- $\bar{X} = Mean(Kumar) = \frac{4+5+3+5+1+0+0}{7} = 2.57$
- $\bar{Y} = Mean(Rahul) = \frac{5+5+4+0+2+0+0}{7} = 2.29$

- $\sigma_X = stdDeviation(Kumar) = 5.45$
- $\sigma_Y = stdDeviation(Rahul) = 5.78$

Correlation based similarity

- Correlation(Kumar, Rahul) is given as follows

$$\frac{\begin{array}{c}(4-2.7)*(5-2.29)+(5-2.7)*(5-2.29)\\+(3-2.7)*(4-2.29)+(5-2.7)*(0-2.29)\\+(1-2.7)*(2-2.29)+(0-2.7)*(0-2.29)\\+(0-2.7)*(0-2.29)\end{array}}{5.45\ *5.78}=0.567$$

- Similarly we calculate Correlation(Kumar, Ram) = -0.843
- And Correlation(Ram, Rahul) = -0.673
- Thus we see that Kumar and Rahul are very similar. Thus from the given example, **Item4** will be recommended to Rahul.

Matrix Factorization

- Basic idea is to decompose a matrix into smaller matrices.
- These smaller matrices can be multiplied to get the bigger matrix.
- Simple example is
  - 36 = 4 x 9
  - We broke 36 into its factors 4 and 9.
- A dataset having movie ratings and users.
  - Dataset = movies (with features of movies) X users (with features of users)
  - Example (Harry potter) = Harry potter (features: fantasy, magic, Daniel Radcliffe) X user (features: likes fantasy, magic)
- A new user will be compared with the most similar user, and then be recommended the movie.
- This process is done via Singular Value Decomposition (SVD)
- SVD is an algorithm which decomposes the given matrix into the best smallest rank matrix possible.

Matrix Factorization - formula

$$A_{m\,x\,n} = Um_{\,x\,r} \sum_{r\,x\,r} (V_{n\,x\,r})^T$$

- A: Input data matrix
  - m x n matrix (eg. m documents, n terms)
- U: Left singular vectors
  - m x r matrix (m documents, r concepts)
- ∑: Singular values
  - r x r diagonal matrix (strength of each ‘concept’. r : rank of matrix A
- V: Right singular vectors
  - n x r matrix (n terms, r concepts)

Matrix Factorization - example

	MV1	MV2	MV3	MV4	MV5
Thriller	3	1	1	3	1
Comedy	1	2	4	1	3

	Thriller	Comedy
A	1	0
B	0	1
C	1	0
D	1	1

	MV1	MV2	MV3	MV4	MV5
A	3	1	1	3	1
B	1	2	4	1	3
C	3	1	1	3	1
D	4	3	5	4	4

- B likes comedy. Movie 3 has 4 points of comedy and 1 point for thriller. Thus, B's rating for the movie would be 0*1 + 1*4 = 4.

- This is how the factorization is done (using gradient descent) to find out the optimal weights for the features used for both U and V.

Matrix Factorization - example

	MV1	MV2	MV3	MV4	MV5
Thriller	3	1	1	3	1
Comedy	1	2	4	1	3

	Thriller	Comedy
A	1	0
B	0	1
C	1	0
D	1	1
E	1	1

	MV1	MV2	MV3	MV4	MV5
A	3	1	1	3	1
B	1	2	4	1	3
C	3	1	1	3	1
D	4	3	5	4	4
E	3	3			4

- E is the new user who likes both Thriller and Comedy. E has similar tastes like D. Thus, using multiplication, we predict that E would also give (1*1 + 1*4) = 5 for **MV3.** So we recommend **MV3** for E hoping it matches E's taste.

Evaluation metrics

- Offline evaluation – evaluate the system using low prediction errors like RMSE/MAE
- Precision/Recall scores are also used to evaluate the performance. This depends on the domain in which the system is being used.
- Online evaluation – evaluate the business success using A/B testing to make sure your recommender system works well. Other measures include Click Through Rate and Conversion Rate
- Personalization – using cosine similarity you can figure out if the system is giving proper personalized recommendations for the user.
- ROI – return of investment on the based on the system deployed.

Thank You.