

# PRIPOROČILNI SISTEMI

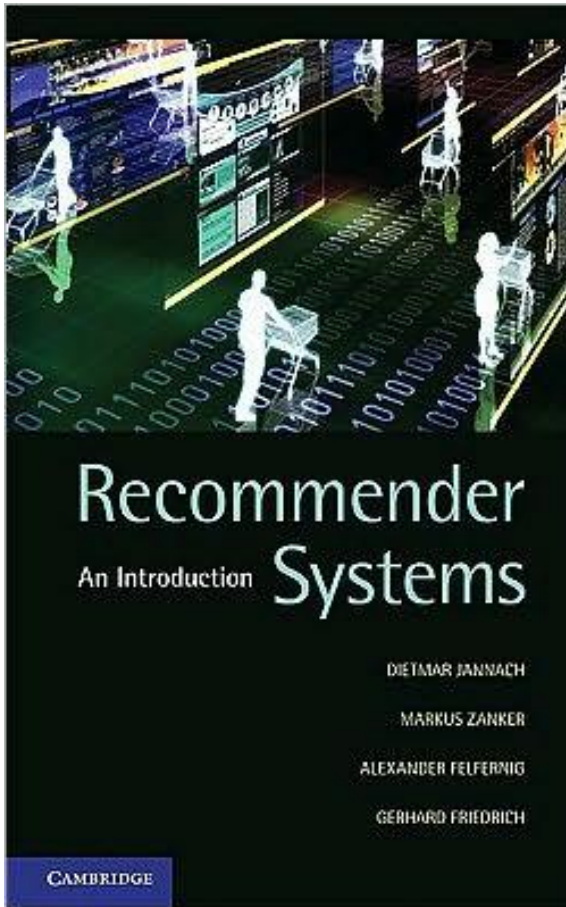
*recommender systems*

doc. dr. Matej Guid

Fakulteta za računalništvo in informatiko  
Univerza v Ljubljani

februar 2025

Jannach D, Zanker M, Felfernig A in Friedrich G. (2010). Recommender systems: an introduction. Cambridge University Press.



*„RS have the potential to support and improve the quality of the decisions consumers make while searching for and selecting products online.“*

(Xiao & Benbasat 2007)

Customers Who Bought This Item Also Bought



Recommender Systems Handbook  
Francesco Ricci  
Hardcover  
\$167.73



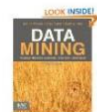
Algorithms of the Intelligent Web  
Haralambos Marmanis  
★★★★★ (14)  
Paperback  
\$26.76



Programming Collective Intelligence: ...  
> Toby Segaran  
★★★★★ (91)  
Paperback  
\$25.20



Machine Learning: A Probabilistic ...  
> Kevin P. Murphy  
★★★★★ (15)  
Hardcover  
\$81.00

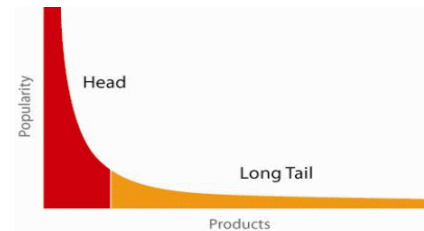
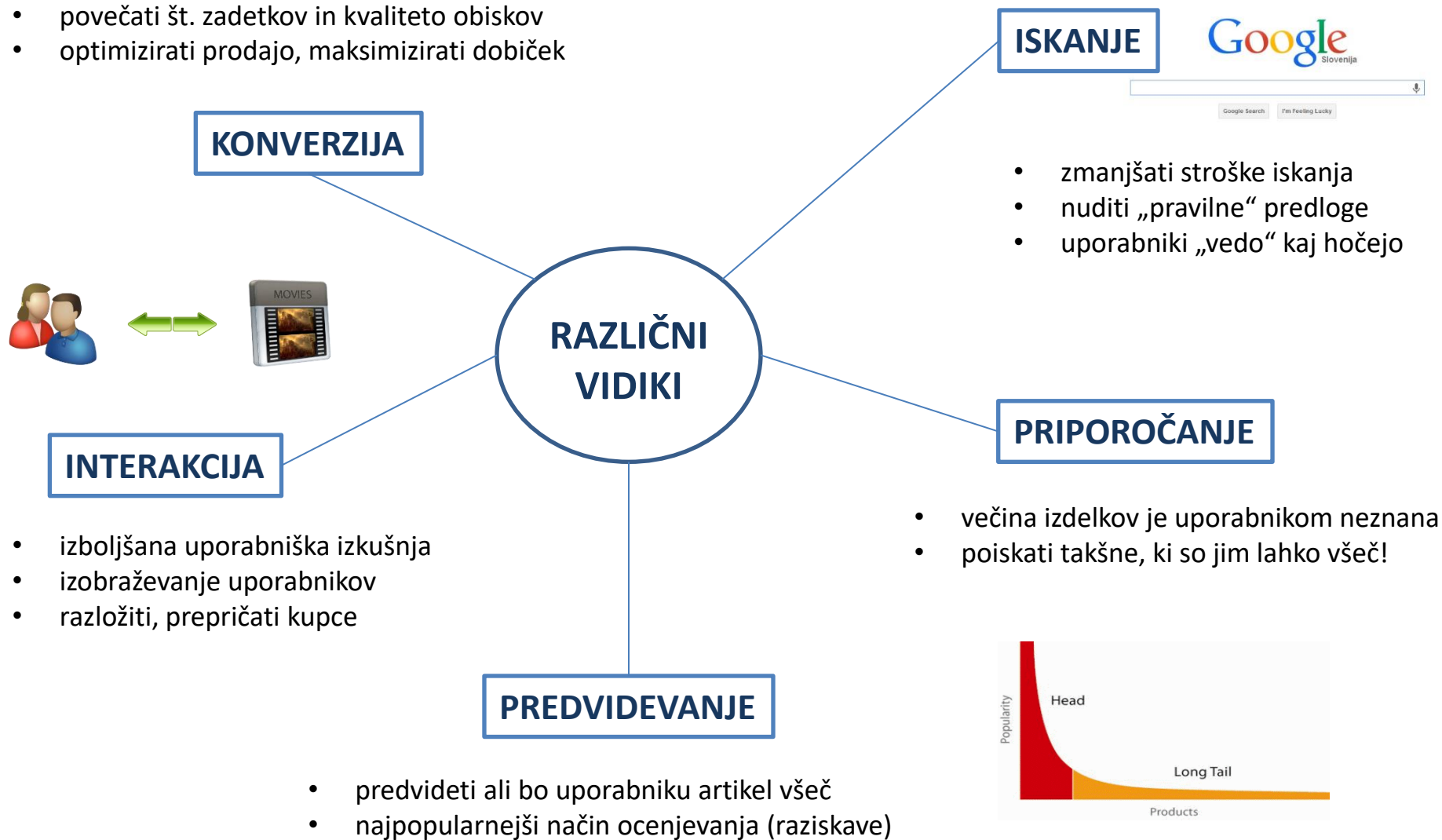


Data Mining: Practical Machine Learning ...  
> Ian H. Witten  
★★★★☆ (29)  
Paperback  
\$42.61

*„One of the most famous examples of collaborative filtering is item-to-item collaborative filtering, an algorithm popularized by Amazon.com's recommender system.“ (Wikipedia)*

# NAMEN PRIPOROČILNIH SISTEMOV

- povečati št. zadetkov in kvaliteto obiskov
- optimizirati prodajo, maksimizirati dobiček



## imamo podano

**U** = množica uporabnikov  $u \in U, u = 1 \dots N$

**I** = skupina izdelkov  $i \in I, i = 1 \dots M$

**R** = ratingi oz. ocene  $r \in R \quad r = \{0,1\}$  ali  $r = \{1,2,3,4,5\}$  ali  $r = [-10,10]$

**(u,i,r)** = preference uporabnikov: uporabnik  $u$  je ocenil izdelek  $i$  z oceno  $r$

Na voljo imamo lahko tudi dodatne opisne podatke o uporabnikih in izdelkih.

## cilj

- pridobiti ocene relevantnosti (in nato razvrščanje...)
- z vidika predvidevanja: oceniti **R** iz **(U,I)** tako, da bo napaka čim manjša

MAE, RMSE, SSE...

**Priporočilni sistemi  
z ocenjevanjem relevantnosti  
zmanjšujejo  
obremenjenost z informacijami**



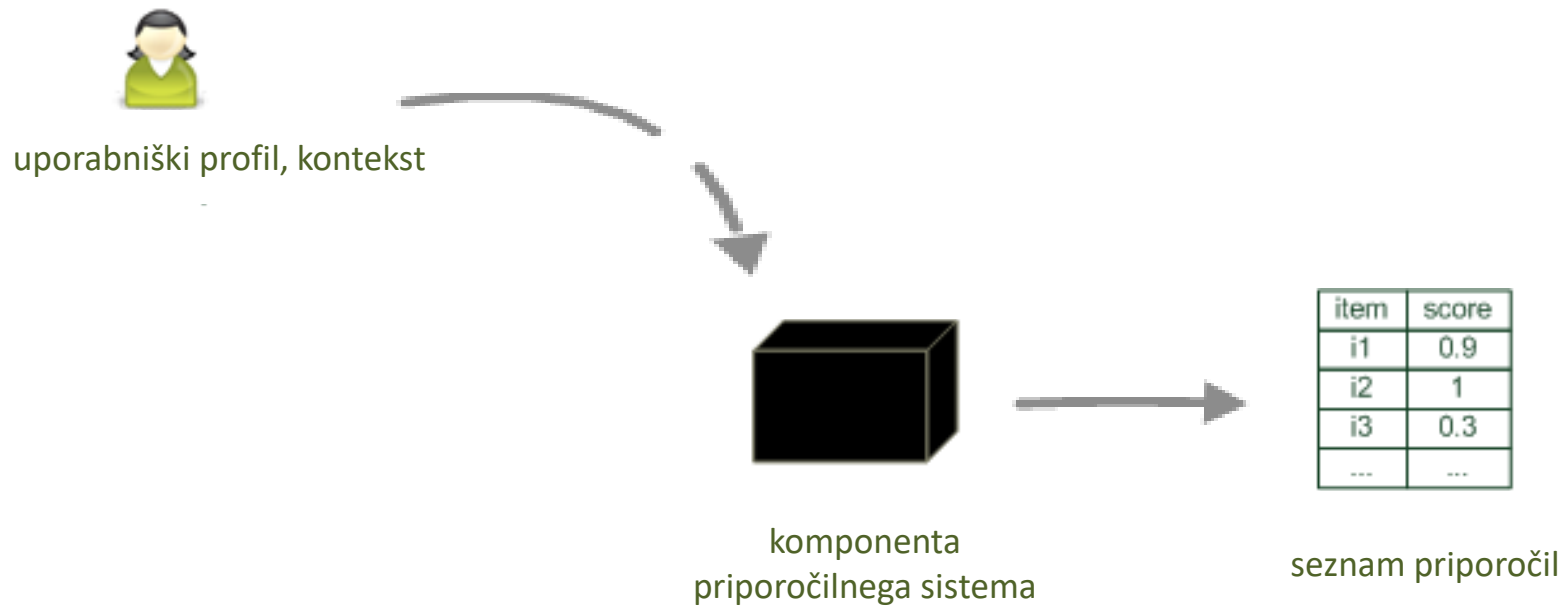
komponenta  
priporočilnega sistema



item	score
i1	0.9
i2	1
i3	0.3
...	...

seznam priporočil

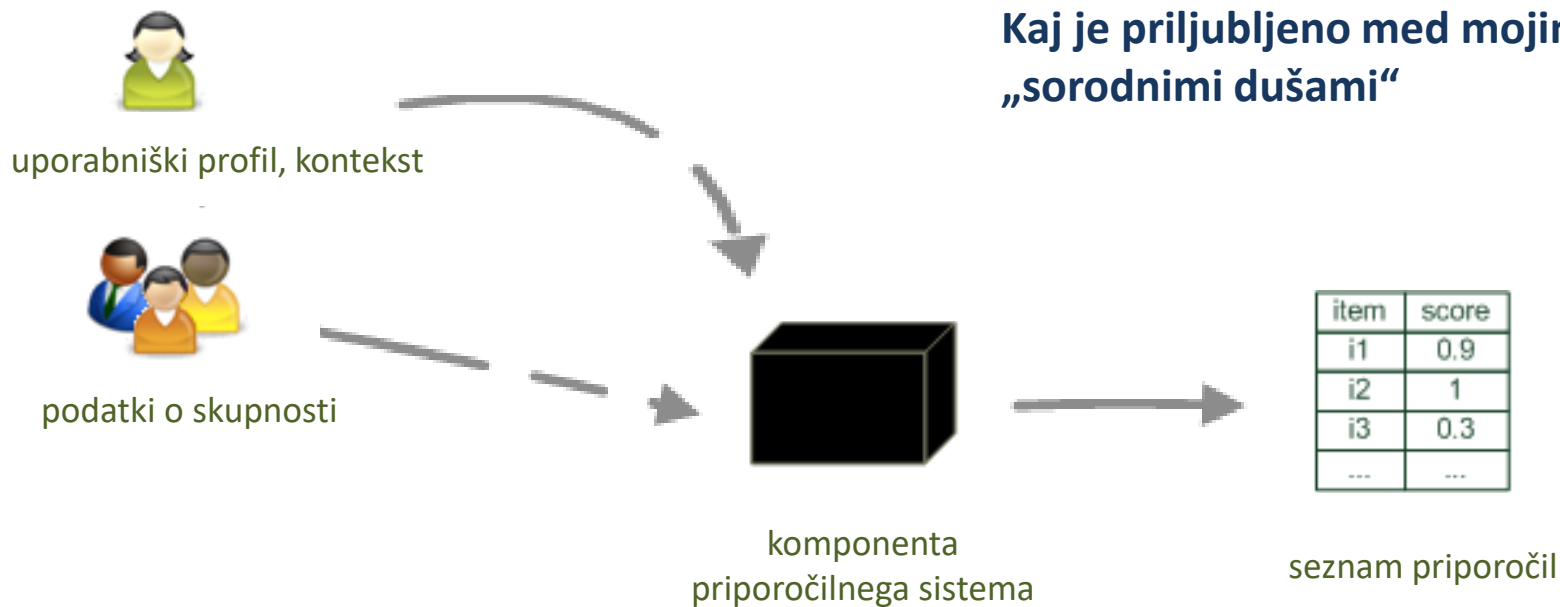
## Personalizirana priporočila



*collaborative filtering*

**Skupinsko filtriranje:**

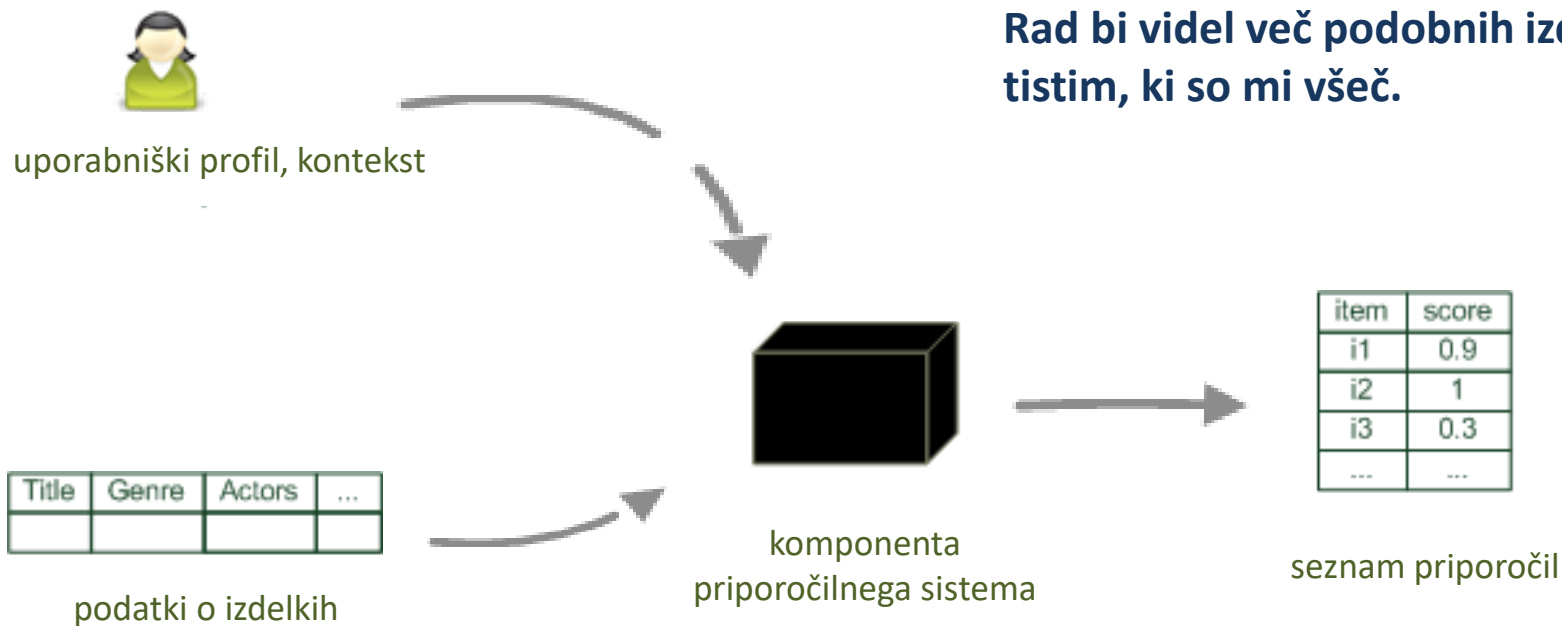
**Kaj je priljubljeno med mojimi „sorodnimi dušami“**



*content-based*

**Temelječi na vsebini:**

**Rad bi videl več podobnih izdelkov  
tistim, ki so mi všeč.**

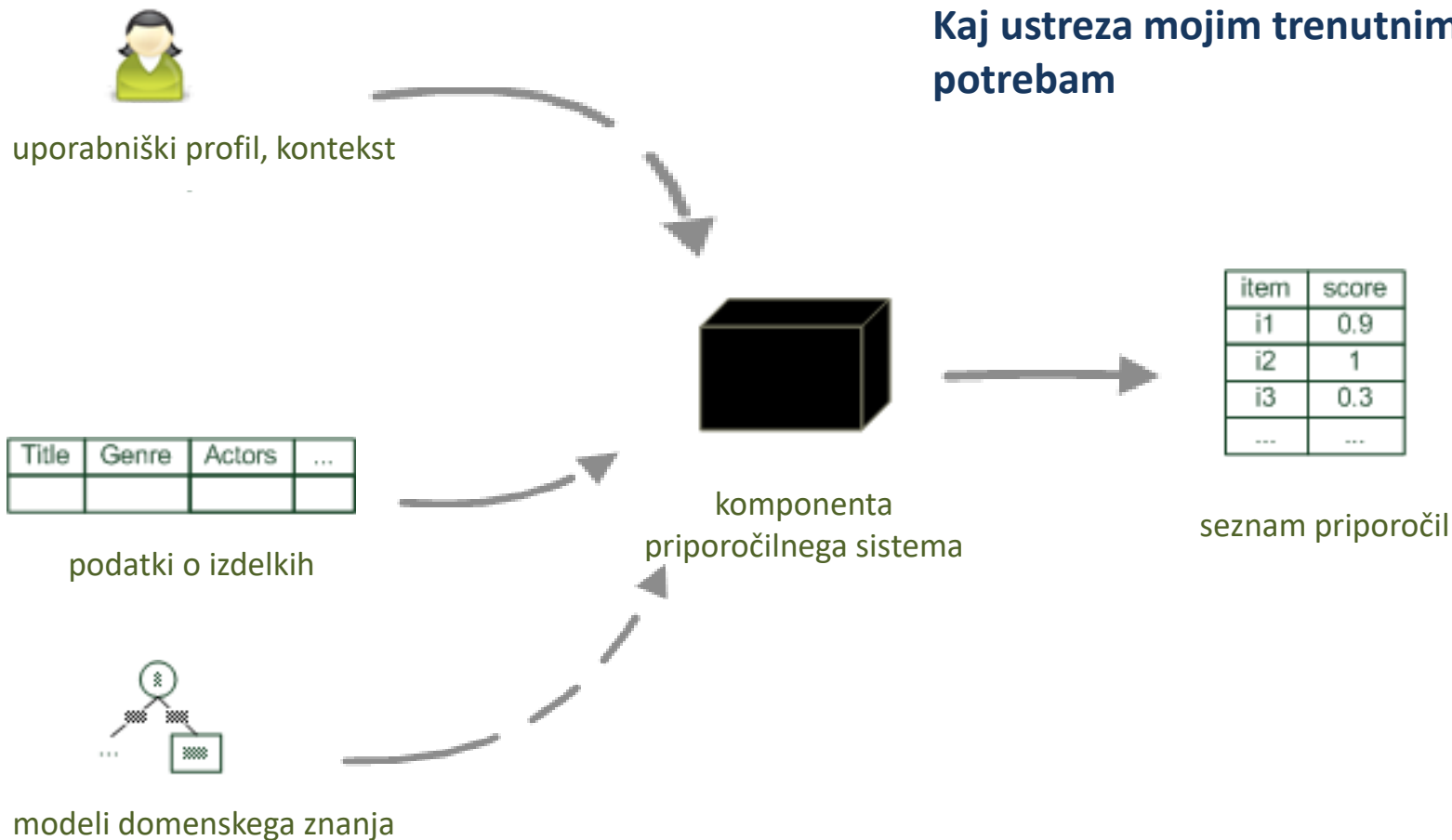




*knowledge-based*

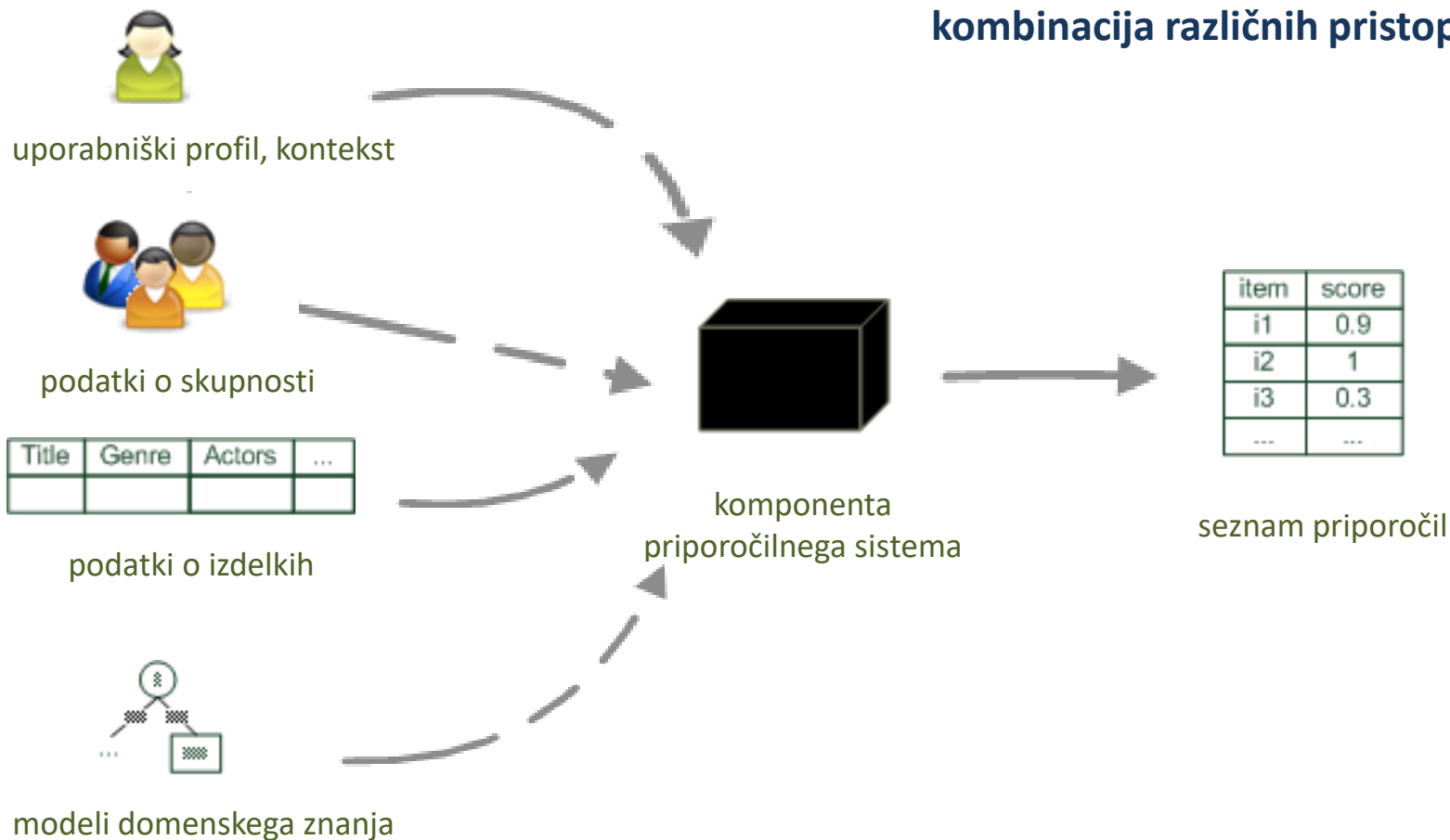
**Temelječi na znanju:**

**Kaj ustreza mojim trenutnim potrebam**



*hybridization*

**Kombinirani:**  
**kombinacija različnih pristopov**





## **skupinsko filtriranje** *collaborative filtering*

- ni ročnega vnašanja znanja
  - učenje zanimivih segmentov
  - omogoča priporočanje „z repa“
  - skalabilnost
- zahteva podane ocene
  - novi uporabniki in novi izdelki: „hladen zagon“

## **temelječi na vsebini** *content-based*

- brez množice uporabnikov
  - možnost primerjave izdelkov
  - možnost razlag
- zahteva opise izdelkov
  - novi uporabniki: „hladen zagon“
  - ni presenečenj

## **temelječi na znanju** *knowledge-based*

- deterministično priporočanje
  - tipično kvalitetna priporočila
  - brez „hladnega“ začetka
- težavno zajemanje znanje (npr. iz domenskih ekspertov)
  - ignoriranje kratkoročnih trendov

- najbolj priljubljena tehnika v praksi
- številni algoritmi, številne različice
- različne domene (knjige, filmi, DVD-ji, ...)



## pristop

„modrost množice“ ... temelji na preferenčnem znanju množic uporabnikov

## osnovna ideja in predpostavke

- uporabniki dajejo ocene izdelkom (implicitno ali eksplicitno)
- uporabniki, ki so imeli podoben okus v preteklosti, bodo imeli podoben okus še naprej

## vhodni podatki

(le) matrika kombinacij ocen: uporabnik-izdelek

	Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
Uporabnik1	3	1	2	3	3
Uporabnik2	4	3	4	3	5
Uporabnik3	3	3	1	5	4
Uporabnik4	1	5	5	2	1

## tipi izhodnih podatkov

- (numerična) napoved: kako vseč bo določen izdelek izbranemu uporabniku
- seznam („Top-N“) priporočenih predmetov

		izdelki				
		Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
uporabniki	Ana	5	3	4	4	
	Uporabnik1	3	1	2	3	3
	Uporabnik2	4	3	4	3	5
	Uporabnik3	3	3	1	5	4
	Uporabnik4	1	5	5	2	1

## osnovna tehnika

- za izbranega uporabnika nas zanima, kateri so njemu podobni uporabniki
- z uporabo učnih podatkov ocenimo njegove preferenčne ocene neocenjenih izdelkov

## osnovna ideja in predpostavke

- uporabniki, ki so imeli podoben okus v preteklosti, bodo imeli podoben okus še naprej
- uporabniške preference so stabilne in konsistentne skozi čas

		izdelki				
		Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
uporabniki	Ana	5	3	4	4	
	Uporabnik1	3	1	2	3	3
	Uporabnik2	4	3	4	3	5
	Uporabnik3	3	3	1	5	4
	Uporabnik4	1	5	5	2	1

- **Kako naj merimo podobnost med uporabniki**
- Kako generiramo napoved z uporabo ocen sosedov
- Koliko „sosedov“ naj upoštevamo pri tem

# NERA PODOBNOSTI: PEARSONOV KORELACIJSKI KOEFICIENT

		izdelki				
		Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
uporabniki	Ana	5	3	4	4	
	Uporabnik1	3	1	2	3	3
	Uporabnik2	4	3	4	3	5
	Uporabnik3	3	3	1	5	4
	Uporabnik4	1	5	5	2	1

*Pearson correlation coefficient*

$a, b$  : uporabnika

$r_{a,i}$  : ocena uporabnika  $a$  za izdelek  $i$

$I$  : množica izdelkov, ocenjenih s strani obeh uporabnikov ( $a$  in  $b$ )

možne vrednosti so lahko na intervalu  $[-1,1]$

$$\mathit{sim}(a, b) = \frac{\sum_{i \in I} (r_{a,i} - \bar{r}_a)(r_{b,i} - \bar{r}_b)}{\sqrt{\sum_{i \in I} (r_{a,i} - \bar{r}_a)^2} \sqrt{\sum_{i \in I} (r_{b,i} - \bar{r}_b)^2}}$$



# MERA PODOBNOSTI: PEARSONOV KORELACIJSKI KOEFICIENT

		izdelki				
		Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
uporabniki	Ana	5	3	4	4	3
	Uporabnik1	3	1	2	3	3
	Uporabnik2	4	3	4	3	5
	Uporabnik3	3	3	1	5	4
	Uporabnik4	1	5	5	2	1

sim = 0,85

$$\bar{r}_{Ana} = \bar{r}_a = 4$$

$$\bar{r}_{Uporabnik1} = \bar{r}_b = 2,4$$

$$\frac{(5 - \bar{r}_a) * (3 - \bar{r}_b) + (3 - \bar{r}_a) * (1 - \bar{r}_b) + \dots + (4 - \bar{r}_a) * (3 - \bar{r}_b)}{\sqrt{(5 - \bar{r}_a)^2 + (3 - \bar{r}_a)^2 + \dots} \sqrt{(3 - \bar{r}_b)^2 + (1 - \bar{r}_b)^2 + \dots}} = 0,85$$

$$sim(a, b) = \frac{\sum_{i \in I} (r_{a,i} - \bar{r}_a)(r_{b,i} - \bar{r}_b)}{\sqrt{\sum_{i \in I} (r_{a,i} - \bar{r}_a)^2} \sqrt{\sum_{i \in I} (r_{b,i} - \bar{r}_b)^2}}$$

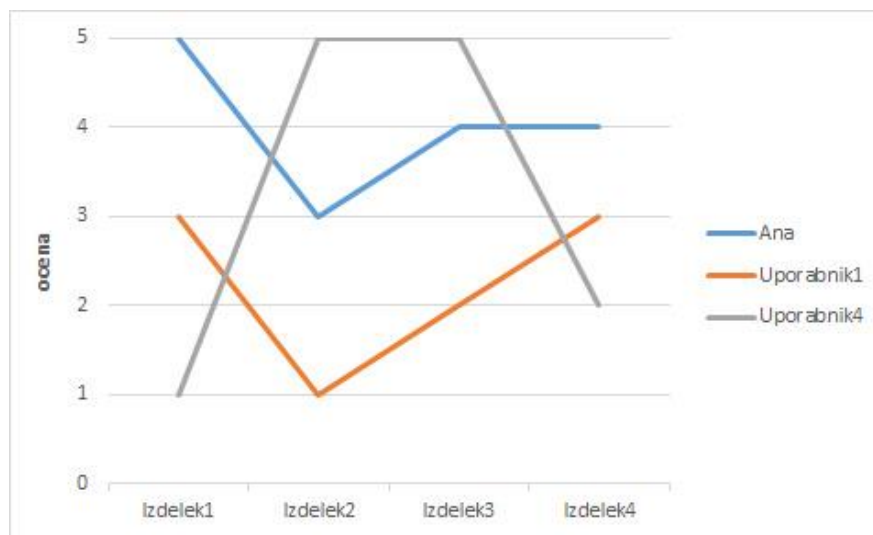
# NERA PODOBNOSTI: PEARSONOV KORELACIJSKI KOEFICIENT

izdelki

	Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
Ana	5	3	4	4	
Uporabnik1	3	1	2	3	3
Uporabnik2	4	3	4	3	5
Uporabnik3	3	3	1	5	4
Uporabnik4	1	5	5	2	1

uporabniki

sim = 0,85  
sim = 0,70  
sim = 0,00  
sim = -0,79



izdelki

	Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5	
uporabniki	Ana	5	3	4	4	
	Uporabnik1	3	1	2	3	3
	Uporabnik2	4	3	4	3	5
	Uporabnik3	3	3	1	5	4
	Uporabnik4	1	5	5	2	1

sim = 0,85  
sim = 0,70  
sim = 0,00  
sim = -0,79

- Kako naj merimo podobnost med uporabniki
- **Kako generiramo napoved z uporabo ocen sosedov**
- Koliko „sosedov“ naj upoštevamo pri tem

		izdelki					
		Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5	
uporabniki	Ana	5	3	4	4		
	Uporabnik1	3	1	2	3	3	sim = 0,85
	Uporabnik2	4	3	4	3	5	sim = 0,70
	Uporabnik3	3	3	1	5	4	sim = 0,00
	Uporabnik4	1	5	5	2	1	sim = -0,79

Upoštevajmo pri izračunu samo najbolj podobna uporabnika (U1 in U2):

$$4 + \frac{(0,85 * (3 - 2,4)) + (0,70 * (5 - 3,8))}{0,85 + 0,70} = 4,87$$

$$pred(a, i) = \bar{r}_a + \frac{\sum_{b \in N} sim(a, b) * (r_{b,i} - \bar{r}_b)}{\sum_{b \in N} sim(a, b)}$$

# IZBIRA RELEVANTNIH „SOSEDOV“

		izdelki				
		Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
uporabniki	Ana	5	3	4	4	
	Uporabnik1	3	1	2	3	3
	Uporabnik2	4	3	4	3	5
	Uporabnik3	3	3	1	5	4
	Uporabnik4	1	5	5	2	1

sim = 0,85  
sim = 0,70  
sim = 0,00  
sim = -0,79

- Kako naj merimo podobnost med uporabniki
- Kako generiramo napoved z uporabo ocen sosedov
- **Koliko „sosedov“ naj upoštevamo pri tem**

*neighborhood selection*

▪ spodnja meja podobnosti

▪ fiksno število sosedov (tipično od 20 do 50)

Ugotavljanje podobnosti med uporabniki temelji na pomnjenju (ang. „*memory-based*“)

**problem: skalabilnost**

**Pristopi, ki temeljijo na izgradnji napovednega modela** (ang. „*model-based*“)

- *offline* predprocesiranje: učenje napovednega modela
- pri procesiranju „v živo“ uporabljamo le naučeni model
- napovedni model periodično nadgrajujemo
- uporabljane so številne tehnike (faktorizacija matrik, povezovalna pravila, ...)
- **podobnosti med izdelki** so tipično **bolj stabilne** kot podobnosti med uporabniki

# PODOBOST MED IZDELKI

		izdelki				
		Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
uporabniki	Ana	5	3	4	4	
	Uporabnik1	3	1	2	3	3
	Uporabnik2	4	3	4	3	5
	Uporabnik3	3	3	1	5	4
	Uporabnik4	1	5	5	2	1

## osnovna ideja

- uporabimo podobnost med izdelki (ne uporabniki) za določanje napovedi

## ilustrativni primer

- poiščemo izdelke, ki so podobni *Izdelku5*
- vzemimo Anine ocene za podobne izdelke in jih uporabimo za napoved ocene

# KOSINUSNA PODOBNOST

		izdelki				
		Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
uporabniki	Ana	5	3	4	4	
	Uporabnik1	3	1	2	3	3
	Uporabnik2	4	3	4	3	5
	Uporabnik3	3	3	1	5	4
	Uporabnik4	1	5	5	2	1

$$\text{sim}(\vec{a}, \vec{b}) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}| * |\vec{b}|}$$

- koti med ocenami, le-te so  $n$ -dimenzionalni vektorji
- vrednosti med 0 in 1

Izračunajmo kosinusno podobnost med izdelkoma *Izdelek5* in *Izdelek1*:

$$\text{sim}(I5, I1) = \frac{3*3 + 5*4 + 4*3 + 1*1}{\sqrt{3^2+5^2+4^2+1^2}\sqrt{3^2+4^2+3^2+1^2}} = 0,99$$



# PRILAGOJENA KOSINUSNA PODOBNOST

		izdelki				
		Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
uporabniki	Ana	5	3	4	4	
	Uporabnik1	3	1	2	3	3
	Uporabnik2	4	3	4	3	5
	Uporabnik3	3	3	1	5	4
	Uporabnik4	1	5	5	2	1

$\bar{r}_u = 2,40$   
 $\bar{r}_u = 3,80$   
 $\bar{r}_u = 3,20$   
 $\bar{r}_u = 2,80$

$$sim(\vec{a}, \vec{b}) = \frac{\sum_{u \in U} (r_{u,a} - \bar{r}_u)(r_{u,b} - \bar{r}_u)}{\sqrt{\sum_{u \in U} (r_{u,a} - \bar{r}_u)^2} \sqrt{\sum_{u \in U} (r_{u,b} - \bar{r}_u)^2}}$$

prilagojena kosinusna  
podobnost

- transformacija upošteva povprečne ratinge uporabnikov
- U: množica uporabnikov, ki je ocenila oba izdelka (*a* in *b*)
- vrednosti med -1 in 1 (kot pri Pearsonovem korelacijskem koeficientu)

# PRILAGOJENA KOSINUSNA PODOBNOST

		izdelki				
		Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
uporabniki	Ana	1,00	-1,00	0,00	0,00	
	Uporabnik1	0,60	-1,40	-0,40	0,60	0,60
	Uporabnik2	0,20	-0,80	0,20	-0,80	1,20
	Uporabnik3	-0,20	-0,20	-2,20	2,80	0,80
	Uporabnik4	-1,80	2,20	2,20	-0,80	-1,80

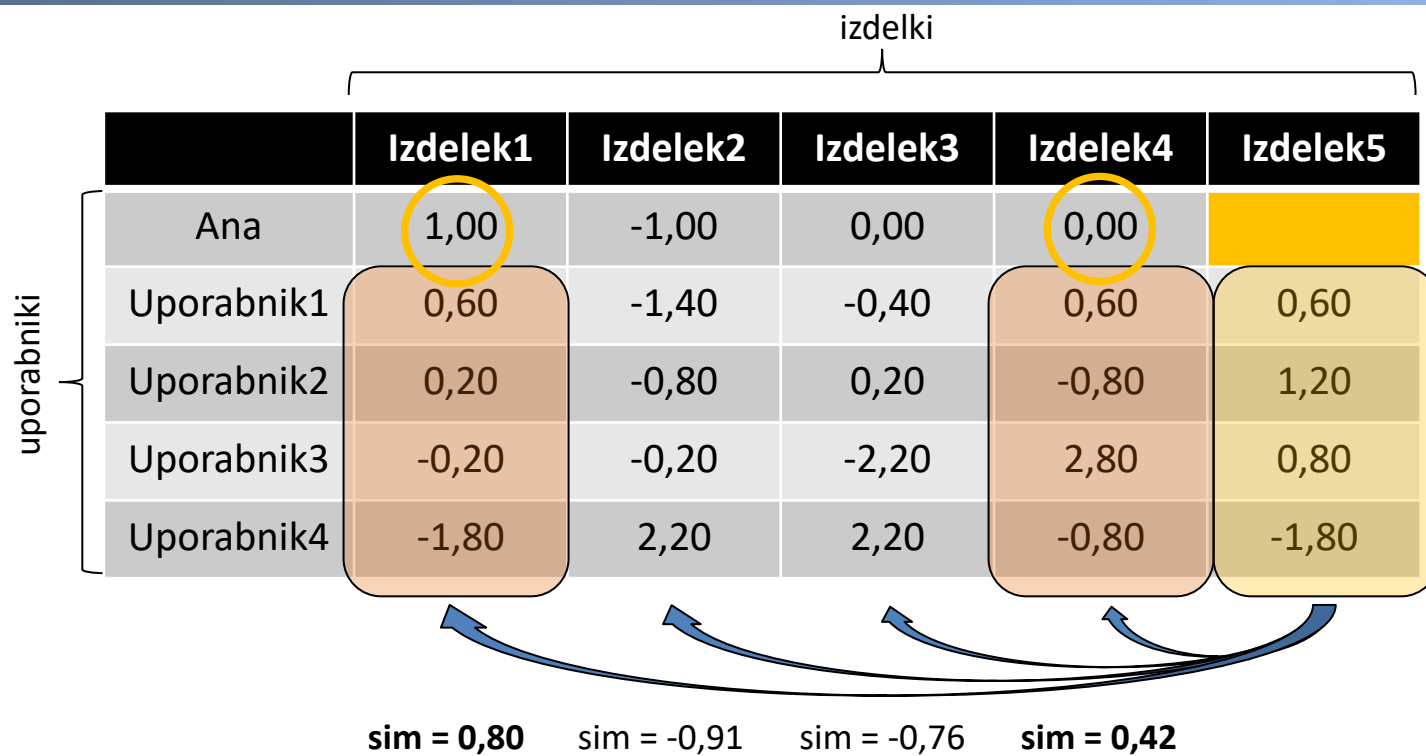
$$sim(\vec{a}, \vec{b}) = \frac{\sum_{u \in U} (r_{u,a} - \bar{r}_u)(r_{u,b} - \bar{r}_u)}{\sqrt{\sum_{u \in U} (r_{u,a} - \bar{r}_u)^2} \sqrt{\sum_{u \in U} (r_{u,b} - \bar{r}_u)^2}}$$

prilagojena kosinusna  
podobnost

Izračunajmo prilagojeno kosinusno podobnost med izdelkoma *Izdelek5* in *Izdelek1*:

$$sim(I5, I1) = \frac{0,6 * 0,6 + 1,2 * 0,2 + 0,8 * (-0,2) + (-1,8) * (-1,8)}{\sqrt{0,6^2 + 0,2^2 + (-0,2)^2 + (-1,8)^2} \sqrt{0,6^2 + 1,2^2 + 0,8^2 + (-1,8)^2}} = 0,80$$

# PRILAGOJENA KOSINUSNA PODOBNOST



# GENERIRANJE NAPOVEDI

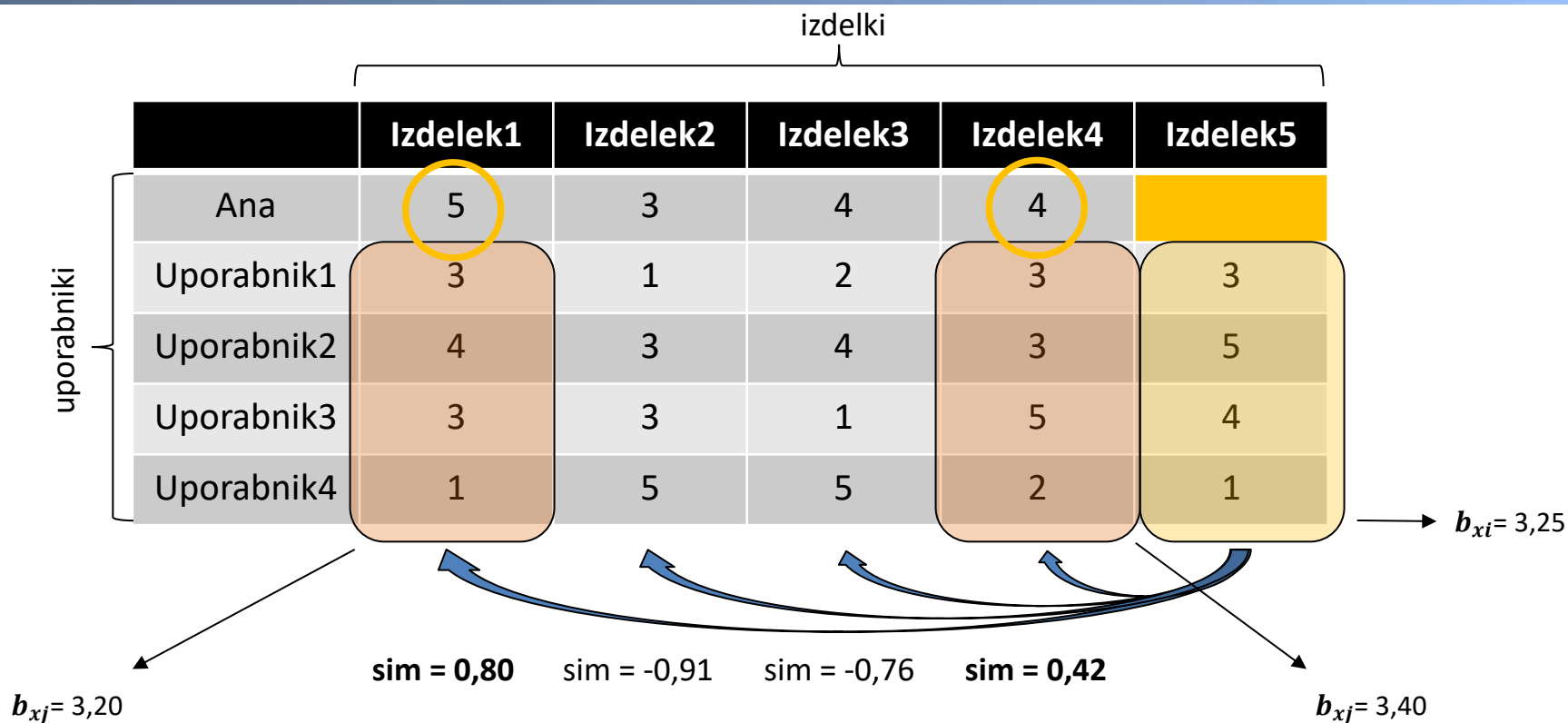
		izdelki				
		Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
uporabniki	Ana	5	3	4	4	
	Uporabnik1	3	1	2	3	3
	Uporabnik2	4	3	4	3	5
	Uporabnik3	3	3	1	5	4
	Uporabnik4	1	5	5	2	1

$sim = 0,80$     $sim = -0,91$     $sim = -0,76$     $sim = 0,42$

$$\frac{0,8*5 + 0,42*4}{0,8+0,42} = 4,66$$

$$pred(u, i) = \frac{\sum_{k \in ocenjenIzdelek(u)} sim(k, i) * r_{u,k}}{\sum_{k \in ocenjenIzdelek(u)} sim(k, i)}$$

# GENERIRANJE NAPOVEDIV PRAKSI



$$r_{xi} = b_{xi} + \frac{\sum_{j \in N(i;x)} s_{ij} \cdot (r_{xj} - b_{xj})}{\sum_{j \in N(i;x)} s_{ij}}$$

upoštevamo lahko še več kriterijev,  
ne le povprečnega ratinga predmeta

$$3,25 + \frac{(0,8 * (5 - 3,2)) + (0,42 * (4 - 3,4))}{0,8 + 0,42} = 4,90$$

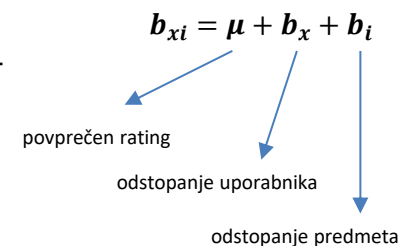
# INTERAKCIJA MED PARI IZDELKOV



„podóbnost -i ž (ó) lastnost, stanje podobnega

subjektivno!

$$r_{xi} = b_{xi} + \frac{\sum_{j \in N(i;x)} s_{ij} \cdot (r_{xj} - b_{xj})}{\sum_{j \in N(i;x)} s_{ij}}$$



## ideja

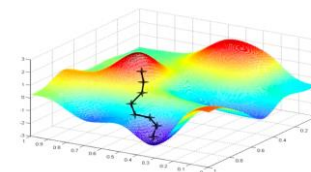
- modeliramo interakcijo med izdelki za določanje napovedi
- učimo se napovednih modelov, zastavimo kot optimizacijski problem

$$\widehat{r}_{xi} = b_{xi} + \sum_{j \in N(i;x)} w_{ij} (r_{xj} - b_{xj})$$

$w_{ij}$  modelira interakcijo med pari izdelkov (neodvisno od uporabnikov!)

minimiziramo to vrednost

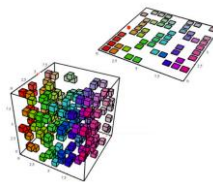
SSE:  $\sum_{(i,x) \in R} (\widehat{r}_{xi} - r_{xi})^2$



# SVD - REDUKCIJA DIMENZIJ IN „PREPOZNAVANJE“ KONCEPTOV

*singular value decomposition*

$$M = U \Sigma V^*$$



- iskanje podobnosti po konceptih

	Izdelek 1	Izdelek 2	Izdelek 3	Izdelek 4	Izdelek 5
--	-----------	-----------	-----------	-----------	-----------

	1	1	1	0	0
filmi	3	3	3	0	0
	4	4	4	0	0
	5	5	5	0	0
	0	2	0	4	4
knjige	0	0	0	5	5
	0	1	0	2	2

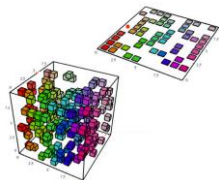
	filmi	knjige	
	0.13	0.02	-0.01
	0.41	0.07	-0.0
	0.55	0.09	-0.04
	0.68	0.11	-0.05
	0	-0.59	0.65
	0	-0.73	-0.67
	0	-0.29	0.32

	filmi	knjige	
	12.3	0	0
	0	9.5	0
	0	0	1.3

	0.56	0.59	0.56	0.09	0.09
	0.12	-0.02	0.12	-0.69	-0.69
	0.40	-0.30	0.40	0.09	0.09

# MATRIČNA FAKTORIZACIJA






$$M = U \Sigma V^*$$



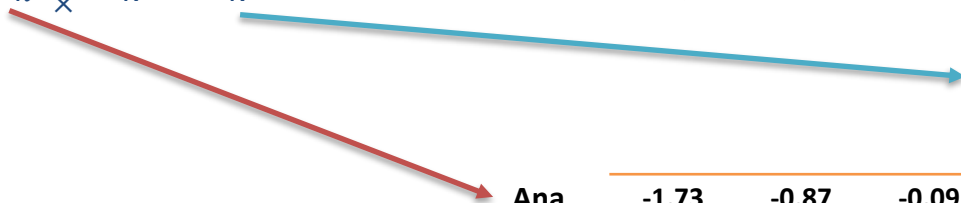
SVD - singular value decomposition

$U_k$	Dim1	Dim2
Ana	0,47	-0,30
Bor	-0,44	0,23
Cene	0,70	-0,06
Domen	-0,31	0,93


$\Sigma_k$	Dim1	Dim2
Dim1	5,63	0
Dim2	0	3,23

$V_k^T$					
Dim1	-0.44	-0.57	0.06	0.38	0.57
Dim2	0.58	-0.66	0.26	0.18	-0.36

$$\widehat{r}_{ui} = \bar{r}_u + U_k \times \Sigma_k \times V_k^T = 3 + 0,83 = 3,83$$



Ana

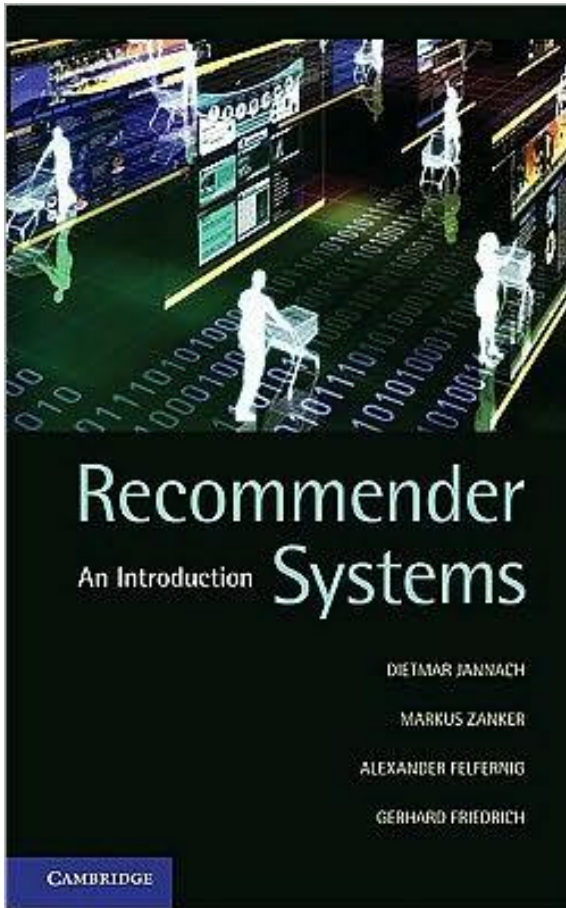
					
	-1,73	-0,87	-0,09	<b>0,83</b>	1,86
	1,52	0,92	0,04	-0,81	-1,68
	-1,85	-2,12	0,19	1,46	2,32
	2,51	-0,99	0,68	-0,12	-2,08



# OCENJEVANJE KAKOVOSTI PRIPOROČIL



Jannach D, Zanker M, Felfernig A in Friedrich G. (2010). Recommender systems: an introduction. Cambridge University Press.



*Kaj naj bo na seznamu priporočil?*

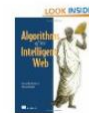
*Kako pa naj vemo, da so ta priporočila ustrezna?*

*Kaj so dobra priporočila?*

Customers Who Bought This Item Also Bought



Recommender Systems Handbook  
Francesco Ricci  
Hardcover  
\$167.73



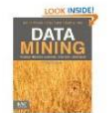
Algorithms of the Intelligent Web  
Haralambos Marmanis  
★★★★★ (14)  
Paperback  
\$26.76



Programming Collective Intelligence: ...  
> Toby Segaran  
★★★★★ (91)  
Paperback  
\$25.20



Machine Learning: A Probabilistic ...  
> Kevin P. Murphy  
★★★★★ (15)  
Hardcover  
\$81.00



Data Mining: Practical Machine Learning ...  
> Ian H. Witten  
★★★★☆ (29)  
Paperback  
\$42.61

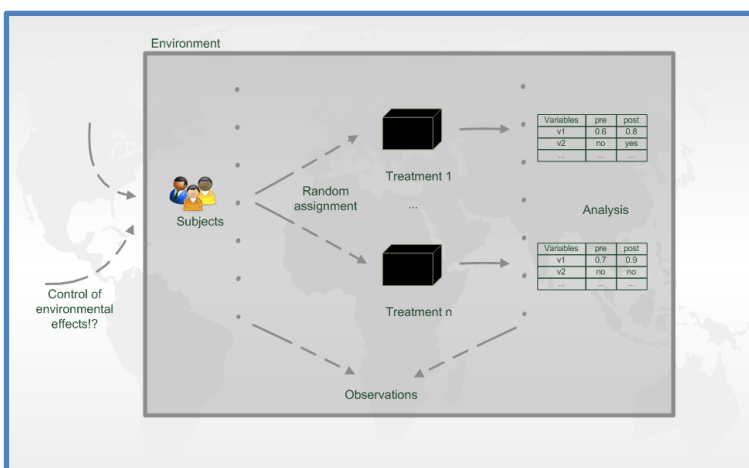
# KAJ SO DOBRA PRIPOROČILA?

- povečati št. zadetkov in kvaliteto obiskov
- optimizirati prodajo, maksimizirati dobiček



## testiranja z resničnimi uporabniki

- A/B testiranja
- mere: prihodki od prodaje, promocija izbranih izdelkov, št. klikov, % ponovnih obiskov...



## laboratorijska testiranja

- nadzorovani eksperimenti
- mere: zadovoljstvo uporabnikov...

## „offline“ testiranja

- zgodovinski podatki
- mere: natančnost napovedi...

Mean Absolute Error (MAE) meri razliko med napovedanimi in dejanskimi ocenami

$$MAE = \frac{1}{n} \sum_{i=1}^n |p_i - r_i|$$

Root Mean Square Error (RMSE) prav tako, vendar bolj občutljiv na večja odstopanja

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (p_i - r_i)^2}$$

Ali je zmotiti se za 20 enot dvakrat slabše kot zmotiti se za 10 enot?

Ali nas res zanimajo vrednosti za vse izdelke? Ali pa morda samo za najbolj relevantne?

# MERJENJE NAPAK PRI NAPOVEDIH

		RESNIČNOST	
		dobro	slabo
NAPOVED	dobra ocena	<b>TRUE POSITIVE</b> tp	<b>FALSE POSITIVE</b> fp
	slaba ocena	<b>FALSE NEGATIVE</b> fn	<b>TRUE NEGATIVE</b> tn

**vsi priporočeni izdelki**

$$Precision = \frac{tp}{tp + fp} = \frac{|good\ movies\ recommended|}{|all\ recommendations|}$$

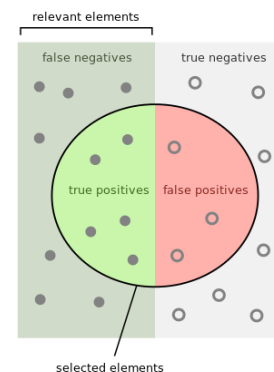
% priporočenih filmov, ki so res dobri

**vsi dobri izdelki**

$$Recall = \frac{tp}{tp + fn} = \frac{|good\ movies\ recommended|}{|all\ good\ movies|}$$

% priporočenih od vseh dobrih filmov

**priklic**  
recall



How many selected items are relevant?

$$Precision = \frac{\text{green}}{\text{green} + \text{red}}$$

How many relevant items are selected?

$$Recall = \frac{\text{green}}{\text{green} + \text{grey}}$$

# PRIPOROČANJE S POMOČJO POVEZOVALNIH PRAVIL



zaupanje (ang. confidence)



“**90%** transakcij, ki vsebujejo kruh in maslo, vsebujejo tudi mleko;  
**30%** transakcij vsebuje vse tri izdelke (kruh, maslo in mleko).”



podpora (ang. support)

## Primer 1

“90% **uporabnikom**, ki sta jim všeč izdelek A in izdelek B, je všeč tudi izdelek C;  
30% uporabnikom so všeč vsi trije izdelki.”

## Primer 2

“90% **izdelkov**, ki so všeč uporabnikoma A in B, je všeč tudi uporabniku C;  
30% vseh izdelkov je všeč vsem trem.”

## Primer 3

“90% izdelkov, ki so všeč uporabniku A in **niso všeč** B-ju, je všeč tudi uporabniku C;  
30% vseh izdelkov je všeč uporabnikoma A in C, medtem ko uporabniku B niso všeč.”



zaupanje (ang. confidence)



**“90% transakcij, ki vsebujejo kruh in maslo, vsebujejo tudi mleko;  
30% transakcij vsebuje vse tri izdelke (kruh, maslo in mleko).”**



podpora (ang. support)

**Kako pretvoriti ocene uporabnikov v „transakcije“**

**Kako povezovalna pravila uporabiti za namen priporočanja**


# PRIPOROČANJE S POMOČJO POVEZOVALNIH PRAVIL

	Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
Ana	1	0	0	0	
Uporabnik1	1	0	1	0	1
Uporabnik2	1	0	1	0	1
Uporabnik3	0	0	0	1	1
Uporabnik4	0	1	1	0	0

## najenostavnejši pristop

- pretvorimo ocene v binarna števila
- 1 = nadpovprečna ocena uporabnika

## povezovalna pravila

- npr. *Izdelek1* → *Izdelek5*  brez Ane
- podpora (2/4), zaupanje (2/2)

## priporočila

- poiščemo relevantna pravila (zgornje pravilo je relevantno, ker je Ana kupila *Izdelek1*)
- poiščemo izdelke, ki jih Ana še ni kupila
- razvrstimo te izdelke glede na vrednost zaupanja pravil

## datatne možnosti

- „ni mi všeč“ opcije v pravilih, povezovalna pravila z uporabniki (ne z izdelki)...

	U1:D	U1:N	U2:D	U2:N	U3:D	U3:N	Tomaž:D
Izdelek1	1	0	1	0	1	0	0
Izdelek2	0	1	0	0	1	0	1
Izdelek3	1	0	0	1	1	0	1
Izdelek4	0	1	0	1	0	1	0
Izdelek5	1	0	0	1	0	1	1

“90% **izdelkov**, ki so všeč uporabnikoma A in B, je všeč tudi uporabniku C;  
30% vseh izdelkov je všeč vsem trem.”

$$U_A:D \wedge U_B:D \rightarrow U_C:D \quad (\text{z 90\% zaupanjem in 30\% podporo})$$

“90% izdelkov, ki so všeč uporabniku A in **niso všeč** B-ju, je všeč tudi uporabniku C;  
30% vseh izdelkov je všeč uporabnikoma A in C, medtem ko uporabniku B niso všeč.”

$$U_A:D \wedge U_B:N \rightarrow U_C:D \quad (\text{z 90\% zaupanjem in 30\% podporo})$$

	U1:D	U1:N	U2:D	U2:N	U3:D	U3:N	Tomaž:D
Izdelek1	1	0	1	0	1	0	0
Izdelek2	0	1	0	0	1	0	1
Izdelek3	1	0	0	1	1	0	1
Izdelek4	0	1	0	1	0	1	0
Izdelek5	1	0	0	1	0	1	1

$$U_A:D \wedge U_B:D \rightarrow U_C:D \quad (\text{z } \_\_\_\% \text{ zaupanjem in } \_\_\_\% \text{ podporo})$$

“50% izdelkov, ki so všeč uporabnikoma U1 in U3, je všeč tudi Tomažu; 20% vseh izdelkov je všeč vsem trem.”

$$U_A:D \wedge U_B:N \rightarrow U_C:D \quad (\text{z } \_\_\_\% \text{ zaupanjem in } \_\_\_\% \text{ podporo})$$

“100% izdelkov, ki so všeč uporabniku U1 in **niso všeč** uporabniku U2, je všeč tudi Tomažu; 40% vseh izdelkov je všeč uporabniku U1 in Tomažu, medtem ko uporabniku U2 niso všeč.”

## Korak 1

Poiščemo pravila, ki prožijo za ciljnega uporabnika. (kaj mu je všeč?)

$$U_A:D \wedge U_B:D \rightarrow U_x:D$$

$$U_A:D \wedge U_B:N \rightarrow U_x:D$$

## Korak 2

Za vsako pravilo izračunamo zmnožek vrednosti *podpora* in *zaupanje* :

$$\mathit{rezultat}_{\mathit{pravilo}_k} = \mathit{podpora}_{\mathit{pravilo}_k} \times \mathit{zaupanje}_{\mathit{pravilo}_k}$$

## Korak 3

Za vsak izdelek izračunamo vrednost po naslednji formuli:

$$\mathit{rezultat}_{\mathit{izdelek}_i} = \sum_{k=1}^n \mathit{rezultat}_{\mathit{pravilo}_k} \mid \text{če } \mathit{pravilo}_k \text{ proži za } \mathit{izdelek}_i$$

## Korak 4

Priporočimo izdelke z največ zbranimi točkami (oz. z najvišjim rezultatom).

# POVEZOVALNA PRAVILA Z UPORABNIKI: ILUSTRATIVNI PRIMER

*user associations*

Kateri izdelek naj priporočimo uporabniku UX?

	U1:D	U1:N	U2:D	U2:N	U3:D	U3:N	U4:D	U4:N	UX:D
Izdelek1	1				1			1	0
Izdelek2				1			1		0
Izdelek3	1		1			1		1	1
Izdelek4		1		1		1			?
Izdelek5		1				1			1
Izdelek6	1		1			1		1	?
Izdelek7	1					1		1	1
Izdelek8	1		1						?

Za katera pravila velja:  $UX:D = 1$ ?

Za katere izdelke pravilo proži?

Izračunamo rezultat za vsak izdelek.

	supp	conf	score
$U1:D=1 \rightarrow UX:D=1$	0,250	0,400	0,100
$U4:N=1 \rightarrow UX:D=1$	0,250	0,500	0,125
$U1:D=1 \ U4:N=1 \rightarrow UX:D=1$	0,250	0,500	0,125
$U3:N=1 \rightarrow UX:D=1$	0,375	0,600	0,225
$U1:D=1 \ U3:N=1 \rightarrow UX:D=1$	0,250	0,667	0,167
$U3:N=1 \ U4:N=1 \rightarrow UX:D=1$	0,250	0,667	0,167
$U1:D=1 \ U3:N=1 \ U4:N=1 \rightarrow UX:D=1$	0,250	0,667	0,167

Izdelek4	Izdelek6	Izdelek8
0	1	1
0	1	0
0	1	0
1	1	0
0	1	0
0	1	0
0	1	0

Izdelek4	Izdelek6	Izdelek8
0,000	0,100	0,100
0,000	0,125	0,000
0,000	0,125	0,000
0,225	0,225	0,000
0,000	0,167	0,000
0,000	0,167	0,000
0,000	0,167	0,000

Na osnovi izračuna uporabniku priporočimo Izdelek6:

0,23	<b>1,08</b>	0,10
------	-------------	------

	I1:D	I2:D	I3:D	I4:D	I5:D	I6:D	Ix:D
Uporabnik1	1	0	0	0	1	0	0
Uporabnik2	0	1	0	0	1	0	1
Uporabnik3	1	0	0	1	1	0	1
Uporabnik4	0	1	1	1	0	1	0
Uporabnik5	1	0	0	1	0	0	1

“90% uporabnikom, ki sta jim všeč izdelek A in izdelek B, je všeč tudi izdelek C; 30% uporabnikom so všeč vsi trije izdelki.”

$$I_A:D \wedge I_B:D \rightarrow I_C:D \quad (\text{z } 90\% \text{ zaupanjem in } 30\% \text{ podporo})$$

“100% uporabnikom, ki sta jim všeč izdelek I1 in izdelek I4, je všeč tudi izdelek Ix; 40% uporabnikom so všeč vsi trije izdelki.”

Ali naj izdelek AX priporočimo uporabniku Uporabnik6?

	A1:D	A2:D	A3:D	A4:D	A5:D	A6:D	A7:D	A8:D	AX:D
Uporabnik1	1				1			1	0
Uporabnik2				1			1		0
Uporabnik3	1		1			1		1	1
Uporabnik4		1		1		1			?
Uporabnik5		1				1			1
Uporabnik6	0		1			1			?
Uporabnik7	1					1		1	1
Uporabnik8	1		1						?

## KORAK 1:

Za katera pravila  $I_A:D \wedge I_B:D \dots \wedge I_N:D \rightarrow I_X:D$  velja, da so izbranemu uporabniku vseh **vsi izdelki pred „→“**?

## KORAK 2:

Določimo minimalno zahtevano podporo:  
npr. za katera pravila velja  $\text{supp} > 0,25$ ?

	supp
<del>A1:D=1</del> → AX:D=1	0,250
A6:D=1 → AX:D=1	0,375
<del>A8:D=1</del> → AX:D=1	0,250
<del>A1:D=1 A8:D=1</del> → AX:D=1	0,250
<del>A1:D=1 A6:D=1</del> → AX:D=1	0,250
<del>A6:D=1 A8:D=1</del> → AX:D=1	0,250

Pravilo A6:D=1 → AX:D=1 ima zadostno podporo, zato izbranemu uporabniku izdelek AX priporočimo.



Možnost uporabe mešane strategije, kombiniramo lahko:

- ❑ povezave med uporabniki:

tipično vodijo do bolj točnih napovedi

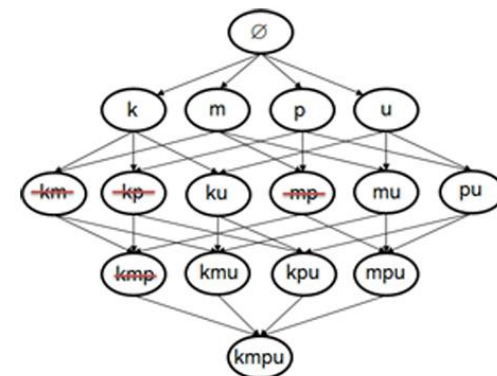
iščemo povezovalna pravila le za enega uporabnika hkrati!

- ❑ povezave med izdelki:

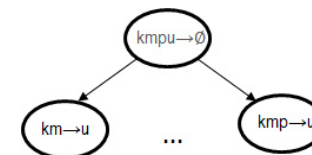
tipično hitrejše delovanje

uporabne še zlasti v primerih nizke podpore pri povezavah med izdelki

iščemo povezovalna pravila le za en izdelek hkrati!



V glavi pravila je le en uporabnik/izdelek hkrati!



Že nizko število pravil lahko vodi do dobrih priporočil!

Lin, W., Alvarez, S. A., & Ruiz, C. (2002). Efficient adaptive-support association rule mining for recommender systems. *Data mining and knowledge discovery*, 6(1), 83-105.

$$\text{Sparsity} = 1 - |R|/|I| \cdot |U|$$

R = ocene ... *ratings*

I = izdelki ... *items*

U = uporabniki ... *users*

ime	domena	uporabniki	izdelki	ocene	razpršenost
BX	knjige	278.858	271.379	1.149.780	0,9999
EachMovie	filmi	72.916	1.628	2.811.983	0,9763
Jester	vici	73.421	101	~ 4.100.000	0,4471
MovieLens	filmi	71.567	10.681	~ 10.000.000	0,9869
Netflix	filmi	~ 480.000	~ 18.000	~ 100.000.000	0.9999

Priljubljene množice podatkov. Vir: *Recommender systems: an introduction* (2010)

## „hladen zagon“ *cold start problem*

Kako priporočati nove izdelke?

Kaj priporočati novim uporabnikom?

### neposreden pristop

- od uporabnikov zahtevamo ocene izbrane množice izdelkov
- na začetku uporabimo neosebna priporočila, priporočila temelječa na vsebini, ...

### alternative

- uporabimo algoritme, ki predpostavljajo „**tranzitivnost**“ med sosedi

	Izdelek1	Izdelek2	Izdelek3	Izdelek4	Izdelek5
Ana	5	3	4	4	?
Uporabnik1	3	1	2	3	?
Uporabnik2	4	3	4	3	5
Uporabnik3	3	3	1	5	4
Uporabnik4	1	5	5	2	1

sim = 0,85

predvidimo najprej oceno podobnega uporabnika

## ideja

- predpostavljamo, da obstaja zelo podoben uporabnik, ki še ni ocenil izbranega izdelka
- izbrano metodo CF uporabimo rekurzivno in predvidimo to oceno
- uporabimo predvideno oceno namesto ocene bolj oddaljenih sosedov

Zhang, J., & Pu, P. (2007). A recursive prediction algorithm for collaborative filtering recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems* (pp. 57-64). ACM.

# PROBLEM RAZPRŠENOSTI OCEN: VEČ VREDNOSTI ZA MIN. PODPORO

*data sparsity*

Kateri izdelek naj priporočimo uporabniku Uporabnik1?

	A1:D	A2:D	A3:D	A4:D	...	A6:D	A7:D	A8:D
Uporabnik1	1	0	1	0	...	?	?	?
Uporabnik2	1	0	1	0	...	1		
Uporabnik3	1	0	1	0	...	1		
Uporabnik4	1	0	1	1	...		1	1
Uporabnik5	1	0	1	1	...			1
Uporabnik6	1	1	1	1	...			1
...	...	...	...	...	...	...	...	...

redke izdelke

**A1:D  $\wedge$  A3:D  $\rightarrow$  A6:D** (s 33% zaupanjem in 33% podporo)

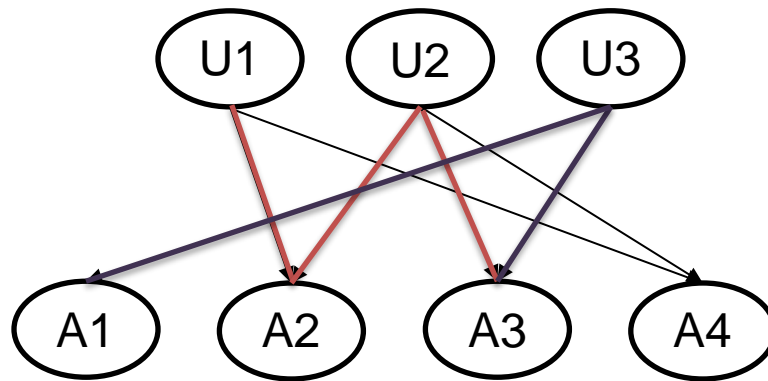
**A1:D  $\rightarrow$  A8:D** (s 50% zaupanjem in 50% podporo)

## ideja

- min. podpora preprečuje „eksplozijo pravil“, vendar preprečuje pravila z redkimi izdelki(!)
- uporabimo lahko več vrednosti za min. podporo, pri tem upoštevamo relativno frekvenco ocen
- pri odločanju glede priporočila damo večji pomen bližnjim sosedom

Gedikli, F., & Jannach, D. (2010). Neighborhood-restricted mining and weighted application of association rules for recommenders. In *Web Information Systems Engineering–WISE 2010* (pp. 157-165). Springer Berlin Heidelberg.

Kateri izdelek naj priporočimo uporabniku U1?

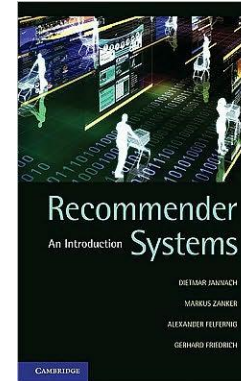


## ideja

- uporabimo daljšo pot (>3 povezave) za dajanje priporočil
- dolžina 3 (klasične metode): priporočimo izdelek A3
- dolžina 5: priporočimo lahko tudi izdelek A1

Huang, Z., Chen, H., & Zeng, D. (2004). Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. *ACM Transactions on Information Systems (TOIS)*, 22(1), 116-142.

- Jannach D., Zanker M., Felfernig A. in Friedrich G. (2010).  
Recommender systems: an introduction.  
Cambridge University Press.



- Segaran T. (2007) Programming Collective Intelligence, drugo poglavje (Making recommendations).
- Lin, W. (2000). Association rule mining for collaborative recommender systems (Doctoral dissertation, Worcester Polytechnic Institute).

**PRIPOROČANJE S POMOČJO  
POVEZOVALNIH PRAVIL**

