

Sistemas de Recomendación

- Netflix (Peliculas)
- Amazon (Productos)
- Youtube (Contenido)
- Ebay, Ali express, Banggood, etc. (Productos)

Ej: Recomendar Peliculas

Usuario califica una pelicula con un rating de 0 a 5 estrellas.

Pelicula	Alicia (1)	Juan (2)	Pedro (3)	Carolina (4)
Interstelar	5	5	0	1
Blade runner	5	?=4	?	1
Va Robot	?	4	1	?
La Monja	0	0	4	4
Chuckie	0	0	5	?

$n_u = \# \text{ usuarios}$
 $n_m = \# \text{ peliculas}$
 $r(i,j) = 1$ si el usuario califico
 $y(i,j) = \text{rating del usuario } j$
para la pelicula i (solo si $r(i,j) = 1$)

Enfoques (Tradiconal)

- 1: Sistemas Recomendadores basados en Contando
Content-based R.S. (Content-based filtering)
- 2: S. R. basados en filtros colaborativos
Collaborative Filtering
- 3*: Sistema hibrido

Content-based Recommender Systems:

Pelicula	Alicia (1)	Juan (2)	Pedro (3)	Carolina (4)	x_1 Ciencia ficcion	x_2 Terror
Interstelar	5	5	0	1	1.0	0
Blade runner	5	?=4	?	1	0.9	0.02
Va Robot	?=4.95	4	1	?	0.99	0
La Monja	0	0	4	4	0	1
Chuckie	0	0	5	?	0.1	0.98

$n_u = 4$
 $n_m = 5$
 $x^{(i)} = 1 \cdot \underline{x_0} + 1 \cdot \underline{x_1} + 0 \cdot \underline{x_2} = x^{(i)} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$

Para cada usuario j , aprender $\theta^{(j)} \in \mathbb{R}^3$
Para predecir el rating de i para j : $(\theta^{(j)})^T x^{(i)}$

$x^{(3)} = \begin{bmatrix} 1 \\ 0.99 \\ 0 \end{bmatrix}$ $\theta^{(1)} = \begin{bmatrix} 0 \\ 5 \\ 0 \end{bmatrix}$ $(\theta^{(1)})^T x^{(3)} = [0.5 \ 0] \begin{bmatrix} 1 \\ 0.99 \\ 0 \end{bmatrix} = 4.95$

x_0
 $x_1 \rightarrow 5$
 x_2

Objetivo de optimizacion

Para un usuario j , aprender $\theta^{(j)}$: Penaliza θ_k grandes
$$\min_{\theta^{(j)}} \underbrace{\frac{1}{2} \sum_{i:r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2}_{\text{regression lineal}} + \underbrace{\frac{\lambda}{2} \sum_{k=1}^n (\theta_k^{(j)})^2}_{\text{regularizacion}}$$

Para todas los usuarios: $\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(n_u)}$
$$\min_{\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(n_u)}} \underbrace{\frac{1}{2} \sum_{j=1}^{n_u} \sum_{i:r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2}_{J(\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(n_u)})} + \frac{\lambda}{2} \sum_{j=1}^{n_u} \sum_{k=1}^n (\theta_k^{(j)})^2$$

Descenso de gradiente:

$\theta_k^{(j)} = \theta_k^{(j)} - \alpha \frac{\partial}{\partial \theta_k^{(j)}} J(\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(n_u)})$

$$\theta_k^{(j)} = \theta_k^{(j)} - \alpha \left(\sum_{i:r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)}) x_k^{(i)} + \lambda \theta_k^{(j)} \right)$$

Content-based

$x^{(i)}$ = caracteristicas de cada pelicula i
 $\theta^{(j)}$ = gustos del usuario j de acuerdo a caracteristicas

Alicia {Accion: 5, Romance: 0.3, Direccion: 0.2, Subt: 0.01, Hollywood: 6}
Carolina
Juan
Pedro

$x \in \mathbb{R}^n$ $n = \# \text{ caract.}$
 $\theta \in \mathbb{R}^n$

$(\theta^{(j)})^T x^{(i)}$
rating predicho del user j para peli i

