

– quantil –

Sistemas de Recomendación

Mónica Ribero

3 de diciembre de 2015

Contenido

- 1 Introducción
- 2 Modelos
 - Datos
 - Modelos Básicos
 - Content Filtering
 - Collaborative Filtering
- 3 Añadir Información al Modelo

Introducción

Motivación

- Demasiados contenidos, demasiada oferta.
- Demasiados perfiles de usuarios
- Cómo sugerirle al consumidor?
- Cómo ayudarlo a encontrar productos de su interés ?

Ejemplos

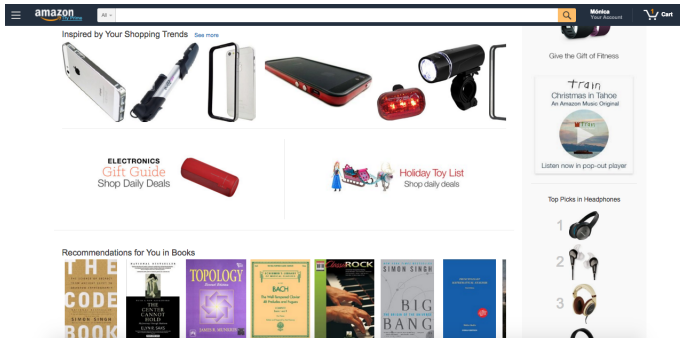


Figura: Página Principal de Amazon

Otros Ejemplos

- Netflix
- YouTube
- Amazon
- Yelp
- Spotify

Definición

Definición

Dado un conjunto de usuarios U y un conjunto de productos P , un sistema de recomendación es un modelo que a partir de datos históricos de cada usuario $u \in U$ predice un producto $p \in P$ que sea de su agrado.

El modelo puede tener diferentes outputs:

- Predicción de una calificación numérica del usuario al producto (Netflix)
- Predicción de si el usuario le gusta o no



Modelos

Datos Históricos

Para realizar recomendaciones se puede utilizar diferentes datos históricos

- 1 Calificaciones de los usuarios a los productos que ya han consumido
- 2 Productos “ vistos”
- 3 Número de visitas

Datos Históricos – Calificaciones

- Se conocen calificaciones de cada usuario a ALGUNOS productos.
- Dado $i \in U$ y $j \in P$, sea r_{ij} la calificación que el usuario i da al producto j
- Objetivo: Predecir r_{ik} para los productos k que el usuario aún no conoce

Subsection 2

Modelos Básicos

Promedios

quantil

Promedios

- Tomar el promedio de las calificaciones
- Dado un usuario $u \in U$, un producto $j \in P$ y U_j el conjunto de usuarios que ya calificaron a j , estimamos

$$\hat{r}_{uj} = \bar{r}_{ij} = \frac{1}{|U_j|} \sum_{i \in U_j} r_{ij}$$

Promedios

- Tomar el promedio de las calificaciones
- Dado un usuario $u \in U$, un producto $j \in P$ y U_j el conjunto de usuarios que ya calificaron a j , estimamos

$$\hat{r}_{uj} = \bar{r}_{ij} = \frac{1}{|U_j|} \sum_{i \in U_j} r_{ij}$$

- Solo captura la mayoría

“Basic Offset”

“Basic Offset”

- Idea: Tener en cuenta el sesgo del usuario y del producto

$$\hat{r}_{uj} = \bar{r} + \alpha_u + \beta_j$$

- $\alpha_u = \frac{1}{|P_u|} \sum_{p \in P_u} r_{pu} - \bar{r}$
- $\beta_j = \frac{1}{|U_j|} \sum_{i \in U_j} r_{ij} - \bar{r}$

Basic Offsets — Ejemplo

Queremos estimar r_{ij} para el usuario i y el producto j en una escala de 1 a 5.

Basic Offsets — Ejemplo

Queremos estimar r_{ij} para el usuario i y el producto j en una escala de 1 a 5.

Suponga:

- Promedio $\mu = 3$

Basic Offsets — Ejemplo

Queremos estimar r_{ij} para el usuario i y el producto j en una escala de 1 a 5.

Suponga:

- Promedio $\mu = 3$
- Usuario i califica en promedio con 4

Basic Offsets — Ejemplo

Queremos estimar r_{ij} para el usuario i y el producto j en una escala de 1 a 5.

Suponga:

- Promedio $\mu = 3$
- Usuario i califica en promedio con 4
- Producto j tiene un promedio de calificaciones de 1

Basic Offsets — Ejemplo

Queremos estimar r_{ij} para el usuario i y el producto j en una escala de 1 a 5.

Suponga:

- Promedio $\mu = 3$
- Usuario i califica en promedio con 4
- Producto j tiene un promedio de calificaciones de 1

$$\hat{r}_{ij} = 3 + (4 - 3) + (1 - 3) = 2$$

Subsection 3

Content Filtering

Content Filtering

- Crear perfiles de usuarios y productos que los caracterizan (datos demográficos, género, actores, etc...)

Content Filtering

- Crear perfiles de usuarios y productos que los caracterizan (datos demográficos, género, actores, etc...)
- Utilizar técnicas de segmentación, asociación y clasificación para recomendar productos.

Content Filtering

- Crear perfiles de usuarios y productos que los caracterizan (datos demográficos, género, actores, etc...)
- Utilizar técnicas de segmentación, asociación y clasificación para recomendar productos.
- Algoritmo utilizado por Pandora

Content Filtering

- Crear perfiles de usuarios y productos que los caracterizan (datos demográficos, género, actores, etc...)
- Utilizar técnicas de segmentación, asociación y clasificación para recomendar productos.
- Algoritmo utilizado por Pandora
- Información difícil de recolectar

Subsection 4

Collaborative Filtering

Collaborative Filtering

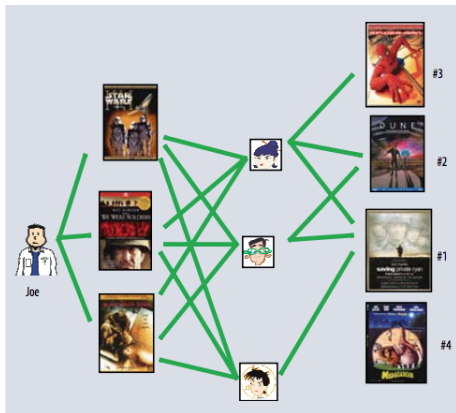
- Basarse únicamente en comportamientos pasados (transacciones, visitas, ...)
- Analizar relaciones entre usuarios e interdependencias entre productos para identificar nuevas asociaciones usuario-producto

Collaborative Filtering

- Basarse únicamente en comportamientos pasados (transacciones, visitas, ...)
- Analizar relaciones entre usuarios e interdependencias entre productos para identificar nuevas asociaciones usuario-producto
- Problema de *cold start*

Métodos de Vecindarios

Trazar relaciones ente usuarios y productos.



Modelos de Factores latentes

Modelos de Factores latentes

- Capturar razones latentes de las calificaciones

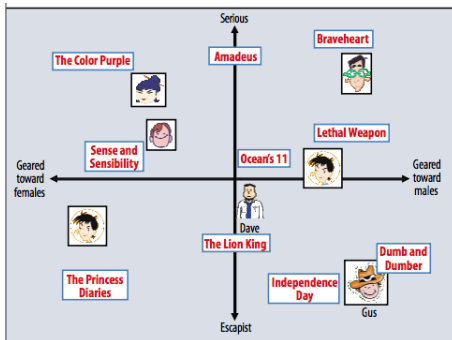
Modelos de Factores latentes

- Capturar razones latentes de las calificaciones
- Metodología: Caracterizar a cada usuario u y cada producto p en K factores (no interpretables) inferidos de patrones de calificación. K es escogido por el modelador.

Modelos de Factores latentes

- Capturar razones latentes de las calificaciones
- Metodología: Caracterizar a cada usuario u y cada producto p en K factores (no interpretables) inferidos de patrones de calificación. K es escogido por el modelador.
- Para los usuarios cada factor significa qué tanto les interesa ese factor particular. Para los productos, el desempeño o contenido del producto de cada factor

Ejemplo Factores Latentes



Factorización de Matrices

Datos: Matriz *sparse* de calificaciones

$$R = \begin{array}{c} \text{usuarios} \\ \left[\begin{array}{ccc} r_{11} & r_{1j} & r_{1n} \\ & r_{22} & \\ \vdots & & \ddots \\ r_{m1} & & r_{mn} \end{array} \right] \end{array}$$

donde r_{ij} = calificación que el usuario i da al producto j .

Factorización de Matrices

- Para cada usuario $u \in U$ y cada producto $p \in P$ encontrar los vectores x_u y y_p en \mathbb{R}^K
- x_u : preferencias del usuario por cada factor
- y_p : desempeño del producto en cada factor
- $r_{up} = x_u^t y_p$

Factorización de Matrices – Ejemplo Productos

$$\vec{y}_p \in \mathbb{R}^k$$
$$\vec{y}_p = \begin{matrix} k \text{ factores} \\ \left[\begin{array}{c} \textit{servicio} \\ \textit{calidad} \\ \textit{precio} \\ \vdots \end{array} \right] \end{matrix}$$

Factorización de Matrices – Ejemplo Productos

$$\vec{y}_p \in \mathbb{R}^k$$
$$\vec{y}_p = \begin{matrix} k \text{ factores} \\ \left[\begin{array}{c} \text{servicio} \\ \text{calidad} \\ \text{precio} \\ \vdots \end{array} \right] \end{matrix} = \begin{bmatrix} 2 \\ 5 \\ 1 \\ 2 \end{bmatrix}$$

Factorización de Matrices – Ejemplo Usuarios

$$\vec{x}_u \in \mathbb{R}^k$$
$$\vec{x}_u = \begin{bmatrix} \text{servicio} \\ \text{calidad} \\ \text{precio} \\ \vdots \end{bmatrix} = \begin{bmatrix} 1 \\ 5 \\ 1 \\ 5 \end{bmatrix}$$

Predicción

Nuevamente,

$$r_{up} = x_u^t y_p$$

Factorización de Matrices

- $r_{up} = x_u^t y_p$
- Encontrar las matrices X y Y tales que

$$X^t Y = R$$

Factorización de Matrices

- $r_{up} = x_u^t y_p$
- Encontrar las matrices X y Y tales que

$$X^t Y = R$$

- Se pueden utilizar métodos de descomposición por valores singulares
- Generalmente falta demasiada información. Se puede estimar entradas faltantes pero esto es muy costoso por el significativo incremento de datos

Factorización de Matrices

- $r_{up} = x_u^t y_p$
- Encontrar las matrices X y Y tales que

$$X^t Y = R$$

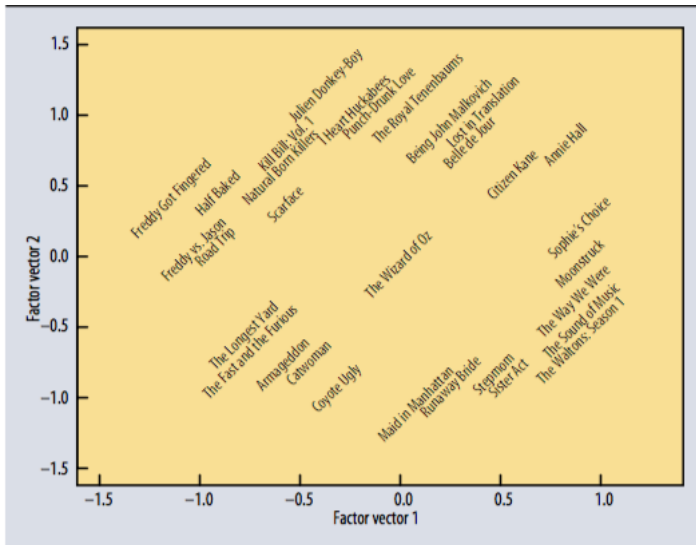
- Se pueden utilizar métodos de descomposición por valores singulares
- Generalmente falta demasiada información. Se puede estimar entradas faltantes pero esto es muy costoso por el significativo incremento de datos
- Utilizar solo las entradas conocidas puede llevar a “overfitting”

Factorización de Matrices – Modelos de regularización

Minimizar el error regularizado sobre el conjunto de calificaciones conocidas

$$\min_{x,y} \sum_{(i,j) \in \kappa} (r_{ij} - x_i^t y_j)^2 + \lambda(\|x_i\|^2 + \|y_j\|^2) \quad (1)$$

- κ es el conjunto de calificaciones conocidas (conjunto de entrenamiento).
- λ controla la regularización y es determinado por crossvalidación



Optimización



Optimización



Descenso estocástico de gradiente

Optimización

Descenso estocástico de gradiente

$$\begin{aligned}e_{up} &:= r_{up} - x_u^t y_p \\ y_p &\leftarrow y_p + \eta(e_{up} x_u - \lambda y_p) \\ x_u &\leftarrow x_u + \eta(e_{up} y_p - \lambda x_u)\end{aligned}$$

- η se conoce como el “paso” o tasa de aprendizaje. La actualización se hace hasta alcanzar cierto error máximo o un número fijo de iteraciones
- Existen otras técnicas como “alternating least squares” que se pueden paralelizar

Añadir Información al Modelo

Sesgo

- Capturar características del usuario o del producto que no tienen que ver con su interacción

Sesgo

- Capturar características del usuario o del producto que no tienen que ver con su interacción
- Sesgo r_{ij} : $b_{ij} = \mu + \alpha_i + \beta_j$
- μ es el promedio general
- α_i y β_j son las desviaciones del usuario i y producto j respectivamente del promedio general

Sesgo

- Capturar características del usuario o del producto que no tienen que ver con su interacción
- Sesgo r_{ij} : $b_{ij} = \mu + \alpha_i + \beta_j$
- μ es el promedio general
- α_i y β_j son las desviaciones del usuario i y producto j respectivamente del promedio general

$$\min_{x,y,b} \sum_{(i,j) \in \kappa} (r_{ij} - \mu - \alpha_i - \beta_j - x_i^t y_j)^2 + \lambda(\|x_i\|^2 + \|y_j\|^2 + \alpha_i^2 + \beta_j^2)$$

Incluir comentarios

Incluir comentarios

Julian McAuley y Jure Leskovec - “Hidden Factors as Topics (HFT)”

Incluir comentarios

Julian McAuley y Jure Leskovec - “Hidden Factors as Topics (HFT)”

- Incorporar Texto
- Descubrir atributos del usuario y los productos
- combina:
 - Modelo Factores Latentes
 - “Latent Dirichlet Allocation”

HFT—The Model

$$f(\tau | \Theta, \Phi, \kappa, z) = \sum_{r_{u,i} \in \tau} (\text{rec}(u, i) - r_{u,i})^2 - \lambda \ell(\tau | \theta, \phi, z)$$

HFT—Modelo

$$f(\tau \mid \Theta, \Phi, \kappa, z) = \sum_{r_{u,i} \in \tau} \underbrace{(\text{rec}(u, i) - r_{u,i})^2}_{\text{error calificación}} - \lambda \underbrace{\ell(\tau \mid \theta, \phi, z)}_{\text{corpus likelihood}}$$

HFT—Modelo

$$f(\tau | \Theta, \Phi, \kappa, z) = \sum_{r_{u,i} \in \tau} \underbrace{(\text{rec}(u, i) - r_{u,i})^2}_{\text{error calificación}} - \lambda \underbrace{\ell(\tau | \theta, \phi, z)}_{\text{corpus likelihood}}$$

error calificación — Diferencia entre calificación real y predecida

corpus likelihood — Probabilidad de ver un corpus particular τ

HFT—Error Calificación

$$f(\tau | \Theta, \Phi, \kappa, z) = \sum_{r_{u,i} \in \tau} \underbrace{(\text{rec}(u, i) - r_{u,i})^2}_{\text{error calificación}} - \lambda \ell(\tau | \theta, \phi, z)$$

- Utilizando modelo de factores latentes
- Queremos minimizar este término

HFT—Corpus Likelihood

$$f(\tau | \Theta, \Phi, \kappa, z) = \sum_{r_{u,i} \in \tau} (\text{rec}(u, i) - r_{u,i})^2 - \lambda \underbrace{\ell(\tau | \theta, \phi, z)}_{\text{corpus likelihood}}$$

HFT—Corpus Likelihood

$$f(\tau | \Theta, \Phi, \kappa, z) = \sum_{r_{u,i} \in \tau} (\text{rec}(u, i) - r_{u,i})^2 - \lambda \underbrace{\ell(\tau | \theta, \phi, z)}_{\text{corpus likelihood}}$$

$$\ell(\tau | \theta, \phi, z) = \prod_{d \in \tau} \prod_{j=1}^{N_d} \theta_{z_{d,j}} \phi_{z_{d,j}, w_{d,j}}$$

- Utilizando modelo LDA
- $\theta_{z_{d,j}}$ — verosimilitud de observar este tópic
- $\phi_{z_{d,j}, w_{d,j}}$ — Verosimilitud de ver palabras particulares en ese tópic.
- z — Asignación de tópicos para cada palabra
- N_d : número de palabras en documento d
- Maximizar este término

HFT—The Model

HFT:

$$f(\tau | \Theta, \Phi, \kappa, z) = \sum_{r_{u,i} \in \tau} (\text{rec}(u, i) - r_{u,i})^2 - \lambda \ell(\tau | \theta, \phi, z)$$

λ intercambia la importancia entre el error de predicción y la verosimilitud del corpus.

Diversidad de productos y de usuarios

- Diversidad Geográfica

Diversidad de productos y de usuarios

- Diversidad Geográfica
- Diversidad de productos

GRACIAS