Recommendation systems

Christos Dimitrakakis

October 11, 2018

The recommendation problem

At time t

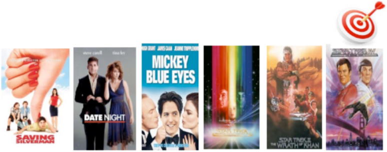
1. A customer x_t appears.
2. We present a choice a_t.
3. The customer chooses y_t.
4. We obtain a reward $r_t = \rho(a_t, y_t) \in \mathbb{R}$.

The two problems in recommendation systems

- The modelling (or prediction) problem.
- The recommendation problem.

How to predict user preferences?

Example: Item-based CF



2			4	5	2.94*
5		4			1
		5		2	2.48*
	1		5		4
		4			2
4	5		1		1.12*

sim(i,j) -1 -1 0.86 1 NA

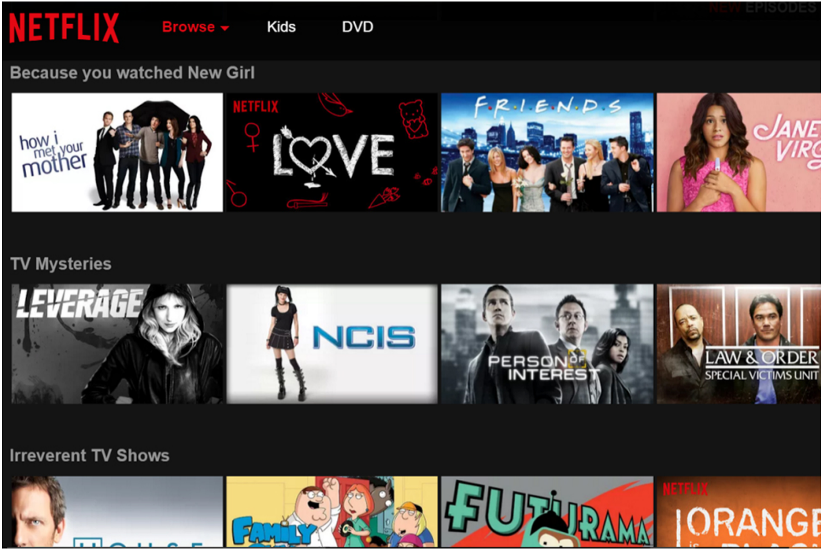


Figure: What to recommend?

Predictions based on similarity

Content-based filtering.

- Users typically like similar items.
- That means we can one user's ratings and item information to predict their ratings for other items.

Collaborative filtering

- Similar users have similar tastes.
- That means we can use similar user's ratings to predict the ratings for other users.

k-NN for similarity

Exercise 1

- Define a distance $d : \mathcal{X}^M \times \mathcal{X}^M \to \mathbb{R}_+$ between user ratings.
- Apply a k-NN-like algorithm to prediction of user ratings from the dataset.

Similarity between users

$$\sum_{j \neq i} w_{i,j} = 1, \qquad w_{i,j}^m \triangleq w_{i,j} \, \mathbb{I}\left\{x_{j,m}\right\} / \sum_k w_{i,k} \, \mathbb{I}\left\{x_{k,m}\right\}.$$

Example 1 (k-nearest neighbours)

$w_{i,j} = 1/k$ for the k nearest neighbours with respect to d.

Example 2 (Weighted distance)

$$w_{i,j} = \frac{\exp[-d(i,j)]}{\sum_{k \neq i} \exp[-d(i,j)]}$$

Inferred ratings

$$\hat{x}_{u.m} = \sum_{j \neq u} w_{u,j}^m x_{u,m}.$$

A naive distance metric

$$d(i, j) \triangleq \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_1.$$

Ignoring movies which are not shared.

$$d(i, j) \triangleq \sum_m \mathbb{I}\left\{x_{i,m} \wedge x_{j,m}\right\} |x_{i,m} - x_{j,m}|$$

Using side-information

Social network data

Inferring a latent representation

$$d(i, j) \triangleq f(\boldsymbol{x}_i, \boldsymbol{x}_j, \theta)$$

Latent representation

The predictive model

- x_{um} rating of user u for movie m.
- $r_{um} = \mathbb{I}\{x_{um} > 0\}$ indicates which movies are rated.
- $\boldsymbol{z}_m \in \mathbb{R}^n$: an n-dimensional representation of a movie.
- $\boldsymbol{c}_u \in \mathbb{R}^n$: an n-dimensional representation of a user.

Given $\boldsymbol{C}, \boldsymbol{Z}$, our predicted movie rating can be written as

$$\hat{x}_{u,m} \triangleq \boldsymbol{c}_u^\top \boldsymbol{z}_m, \qquad \hat{\boldsymbol{X}} \triangleq \boldsymbol{C}^\top \boldsymbol{Z}.$$

$$f(\boldsymbol{C}, \boldsymbol{Z}) = \|(\boldsymbol{R} \circ \hat{\boldsymbol{X}} - \boldsymbol{R} \circ \boldsymbol{X})^\top (\boldsymbol{R} \circ \hat{\boldsymbol{X}} - \boldsymbol{R} \circ \boldsymbol{X})\|_1$$

A simple preference model

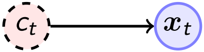


Figure: Basic preference model

Example 3 (Discrete preference model)

- User type $c \in \mathcal{C}$.
- User ratings $\boldsymbol{x}$ with $x_m \in \mathcal{X} = \{0, 1\}$ rating for movie m.
- Preference distribution

$$P_\theta(\boldsymbol{x}|\boldsymbol{c}) = \prod_{m=1}^{M} \theta_{m,c}^{x_m}(1 - \theta_{m,c})^{(1-x_m)}.$$

- $P_\theta(c) = \theta_c$, $\sum_c \theta_c = 1$.

A simple preference model

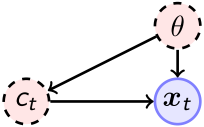


Figure: Basic preference model

Example 3 (Discrete preference model)

- User type $c \in \mathcal{C}$.
- User ratings $\boldsymbol{x}$ with $x_m \in \mathcal{X} = \{0, 1\}$ rating for movie m.
- Preference distribution

$$P_\theta(\boldsymbol{x}|\boldsymbol{c}) = \prod_{m=1}^{M} \theta_{m,c}^{x_m} (1 - \theta_{m,c})^{(1-x_m)}.$$

- $P_\theta(c) = \theta_c$, $\sum_c \theta_c = 1$.

A more complex preference model

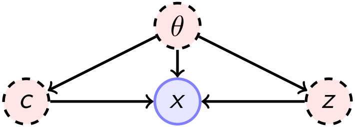


Figure: Preference model

Preference model

- User type $\boldsymbol{c} \in \mathcal{C}$.
- Movie type $\boldsymbol{z} \in \mathcal{Z}$.
- Preference distribution

$$P_\theta(x|\boldsymbol{c}, \boldsymbol{z}) = \mathcal{N}(\boldsymbol{c}^\top \boldsymbol{z}, \sigma_\theta)$$

- Feature prior

$$P_\theta(\boldsymbol{c}) = \mathcal{N}(0, \lambda_\theta)$$

What to recommend

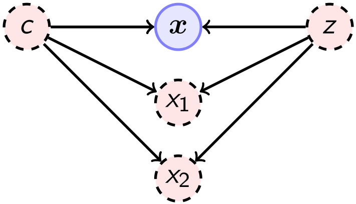


Figure: Preference model

The recommendation problem for a given θ

What to recommend

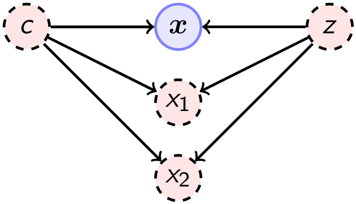


Figure: Preference model

The recommendation problem for a given θ

$$\max_\pi \mathbb{E}^\pi_\theta(U \mid \boldsymbol{x}) = \max_a \sum_{c,z} U(a,y)\, \mathbb{P}(y \mid a, c, z) P_\theta(c, z \mid \boldsymbol{x}) \tag{1.1}$$

$$= \max_a \sum_{c,z} U(a,y) \sum_{x_a} \mathbb{P}(y \mid a, x_a) P_\theta(x_a \mid c, z) P_\theta(c, z \mid \boldsymbol{x}) \tag{1.2}$$

Two ways to model the effect of actions

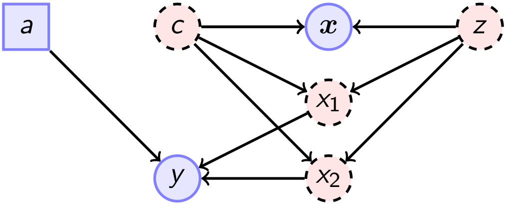


Figure: Preference model

$$\mathbb{E}_\theta(U \mid a, \boldsymbol{x}) = \sum_{c,z} U(a, y) \sum_{x_a} \mathbb{P}(y \mid a, x_a) P_\theta(x_a \mid c, z) P_\theta(c, z \mid \boldsymbol{x}) \quad (1.3)$$

Two ways to model the effect of actions

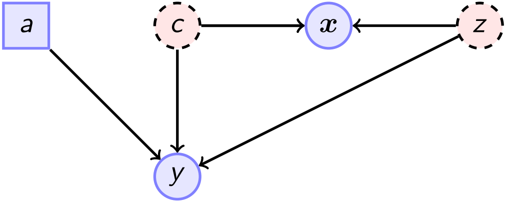


Figure: Preference model

$$\mathbb{E}_\theta(U \mid a, \boldsymbol{x}) = \sum_{c,z} U(a, y)\, \mathbb{P}(y \mid a, c, z) P_\theta(c, z \mid \boldsymbol{x}) \tag{1.3}$$

Clusters as latent variables

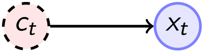


Figure: Graphical model for independent data from a cluster distribution.

The clustering distribution

- Cluster c_t
- Observation x_t
- Parameter θ.

$$x_t \mid c_t = c, \theta \sim P_\theta(x|c), \qquad c_t \mid \theta \sim P_\theta(c), \qquad \theta \sim \xi(\theta)$$

$$P_\theta(c_t \mid x_t) =$$

Clusters as latent variables

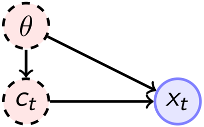


Figure: Graphical model for independent data from a cluster distribution.

The clustering distribution

- Cluster c_t
- Observation x_t
- Parameter θ.

$$x_t \mid c_t = c, \theta \sim P_\theta(x|c), \qquad c_t \mid \theta \sim P_\theta(c), \qquad \theta \sim \xi(\theta)$$

$$P_\theta(c_t \mid x_t) = \frac{P_\theta(x_t \mid c_t) P_\theta(c_t)}{\sum_{c'} P_\theta(x_t \mid c_t = c') P_\theta(c_t = c')}$$

Bayesian formulation of the clustering problem

- Prior ξ on parameter space Θ.
- Data $x^T = x_1, \ldots, x_T$. Cluster assignments c^T unknown.
- Posterior $\xi(\cdot \mid x^T)$.

Posterior distribution

$$\xi(\theta \mid x^T) = \frac{P_\theta(x^T)\xi(\theta)}{\int_\Theta P_{\theta'}(x^T)\,\mathrm{d}\xi(\theta')}, \quad P_\theta(x^T) = \sum_{c^T \in \mathcal{C}^T} \overbrace{P_\theta(x^T \mid c^T)}^{\text{Cluster Density}} \underbrace{P_\theta(c^T)}_{\text{Cluster prior}} \tag{2.1}$$

Bayesian formulation of the clustering problem

- Prior ξ on parameter space Θ.
- Data $x^T = x_1, \ldots, x_T$. Cluster assignments c^T unknown.
- Posterior $\xi(\cdot \mid x^T)$.

Posterior distribution

$$\xi(\theta \mid x^T) = \frac{P_\theta(x^T)\xi(\theta)}{\int_\Theta P_{\theta'}(x^T)\,\mathrm{d}\xi(\theta')}, \quad P_\theta(x^T) = \sum_{c^T \in \mathcal{C}^T} \overbrace{P_\theta(x^T \mid c^T)}^{\text{Cluster Density}} \underbrace{P_\theta(c^T)}_{\text{Cluster prior}} \tag{2.1}$$

Marginal posterior prediction

$$P_\xi(c_t \mid x_t, x^T) = \int_\Theta P_\theta(c_t \mid x_t)\,\mathrm{d}\xi(\theta \mid x^T)$$

Example 4 (Preference clustering)

$$\mathcal{C} = \{1, \ldots, C\}, \qquad x_{t.m} \in \{0, 1\}.$$

$\theta = (\theta_1, \theta_2)$.

Model family

$$P_{\theta_1}(c_t = c) = \theta_{1,c}, \qquad c_t \sim \mathcal{Multinomial}(\theta_1) \tag{2.2}$$

$$P_{\theta_2}(x_{t,m} = 1 \mid c_t = c) = \theta_{2,m,c} \qquad x_{t,m} \mid c_t = c \sim \mathcal{Bernoulli}(\theta_{2,m,c}) \tag{2.3}$$

Prior

$$\theta_1 \sim \mathcal{Dirichlet}(\gamma), \qquad \theta_2 \sim \mathcal{Beta}(\alpha, \beta) \tag{2.4}$$

See `src/pymc/beta_bernoulli_clustering.py`

Supervised learning

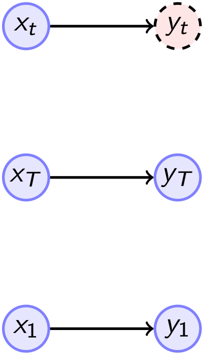


Figure: Graphical model for a classical supervised learning problem.

Semi-supervised learning

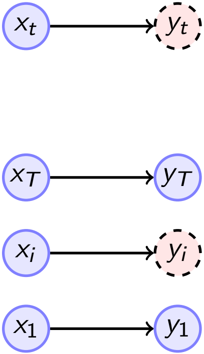


Figure: Graphical model for a classical semi-supervised learning problem.

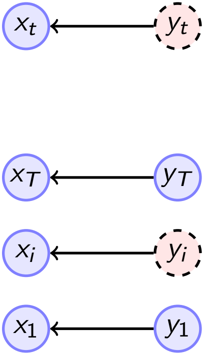


Figure: Generative version of the semi-supervised model

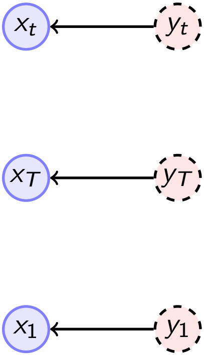


Figure: Basic unsupervised learning model

Applications

- Clustering
- Compression

Network model

Figure: Graphical model for data from a social network.

Network model

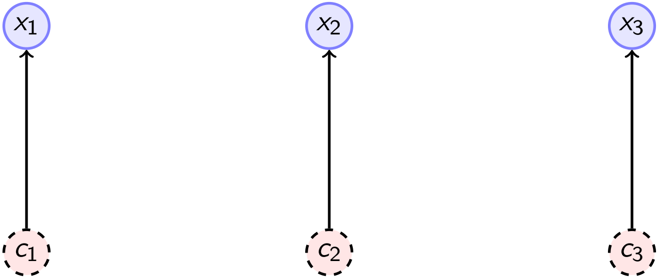


Figure: Graphical model for data from a social network.

Network model

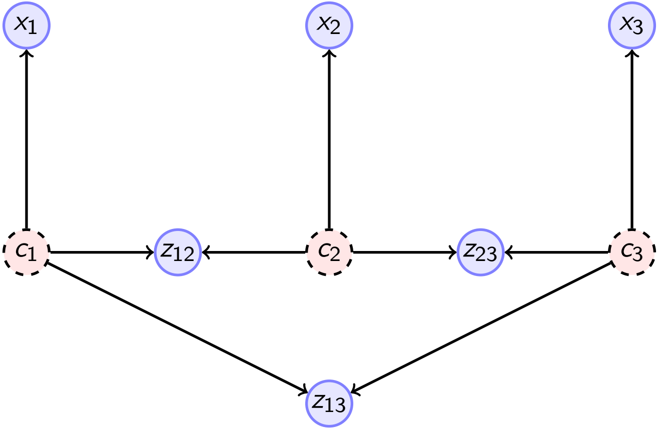
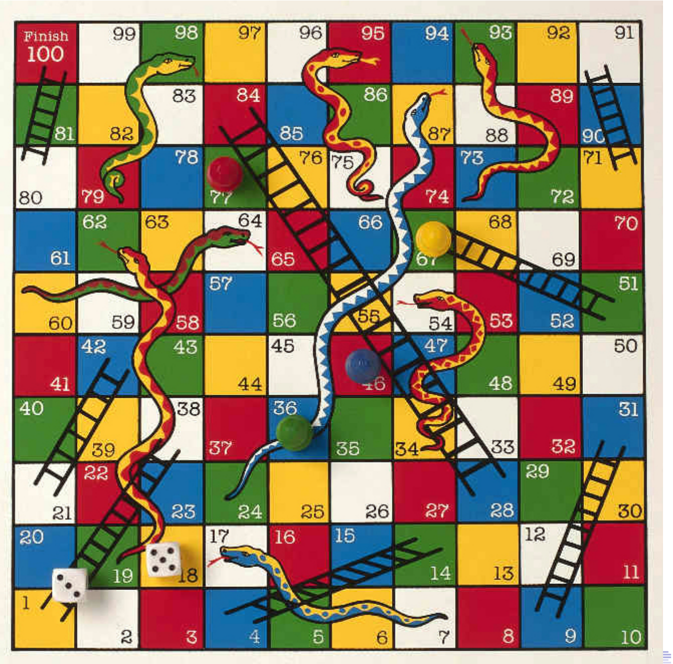


Figure: Graphical model for data from a social network.


Finish 100
99 98 97 96 95 94 93 92 91
81 82 83 84 85 86 87 88 89 90
80 79 78 77 76 75 74 73 72 71
61 62 63 64 65 66 67 68 69 70
60 59 58 57 56 55 54 53 52 51
41 42 43 44 45 46 47 48 49 50
40 39 38 37 36 35 34 33 32 31
21 22 23 24 25 26 27 28 29 30
20 19 18 17 16 15 14 13 12 11
1 2 3 4 5 6 7 8 9 10

Markov process

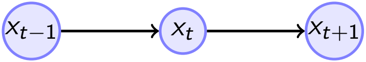


Figure: Graphical model for a Markov process.

Definition 5 (Markov process)

A Markov process is a sequence of variables $x_t : \Omega \to \mathcal{X}$ such that $x_{t+1} \mid x_t \perp\!\!\!\perp x_{t-k} \forall k \leq 1$.

Application

- Sequence compression (especially with variable order Models).
- Web-search (Page-Rank)
- Hidden Markov Models.
- MCMC.

Hidden Markov model

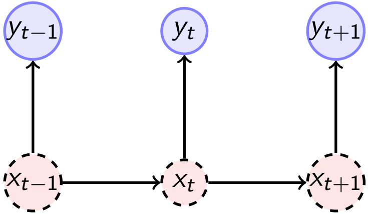


Figure: Graphical model for a hidden Markov model.

$$P_\theta(x_{t+1} \mid x_t) \qquad \text{(transition distribution)}$$

$$P_\theta(y_t \mid x_t) \qquad \text{(emission distribution)}$$

Application

- Speech recognition.
- Filtering (Kalman Filter).
- DNA analysis