Consider the "recommender system" setting (e.g. Netflix)

n users

y_{ij}

m items (e.g. movies)

y_{ij} = # mins user i interacted with item j

← very sparse!

$$\|y\|_0 = \sum_i \sum_j \mathbb{1}(y_{ij} > 0)$$

number of non-zeros

$$\|y\|_0 \ll n \cdot m$$

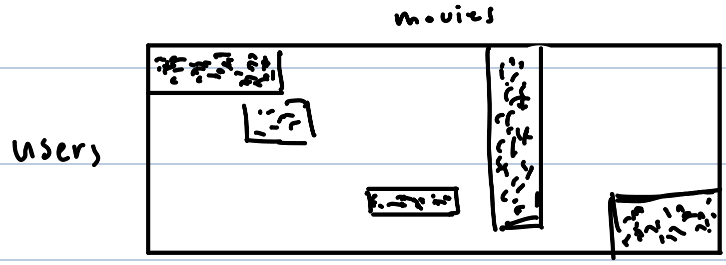
Goal: recommend items to user that they will like.

	Harry Potter movies						Movies about Boston		
	HP1	...	HP8	...	...		Good Will Hunting	The Departed	The Town
user 1	1000	...	10000		0	0	0	0	0
user 2	2000	..	8000	⋱	0	0	100	0	200
user 3	2000	..	1000		0	0	200	300	400

- user 3 has seen all the Boston movies (big Boston fan)
- user 1 has seen none
- user 2 has seen many Boston movies (probably a Boston fan)
  ↳ recommend The Departed to user 2

In practice, we won't know all of the overlapping clusters ("genres") of movies and preferences of users ahead of time. Jointly modelling them is "collaborative filtering".

Intuition : Block structure in matrix



Model:

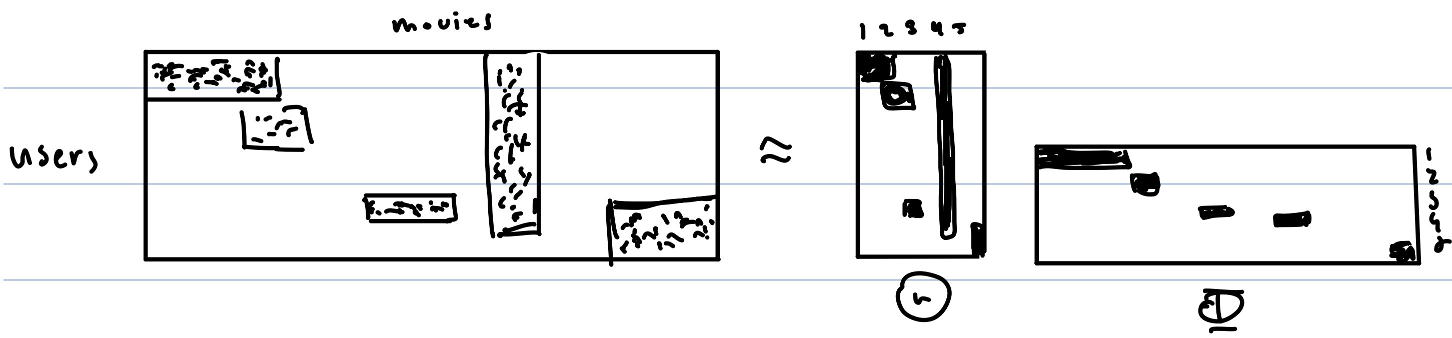
$$\mathbb{E}[y_{ij}] = \sum_k \theta_{ik} \phi_{kj}$$

$\theta_{ik} > 0$: user i's preference for "genre" k

$\phi_{kj} = P(j \mid k) \rightarrow \sum_j^{n} \phi_{kj} = 1$

Matrix factorization

e.g. $K=5$



Likelihood

$$y_{ij} \sim \text{Pois}\Big(\underbrace{\sum_k \theta_{ik} \phi_{kj}}_{= \mathbb{E}[y_{ij}] \overset{\Delta}{=} \mu_{ij}}\Big)$$

MLE :

We could try to fit using MLE

$$\hat{\theta}, \hat{\phi} \leftarrow \underset{\theta, \phi}{\operatorname{argmax}} \underbrace{\log \prod_i \prod_j \operatorname{Pois}(y_{ij} ; \mu_{ij})}$$

$$= \sum_i \sum_j \log \frac{\mu_{ij}^{y_{ij}}}{y_{ij}!} e^{-\mu_{ij}}$$

$$\propto \sum_i \sum_j y_{ij} \log \mu_{ij} - \sum_i \sum_j \mu_{ij}$$

$$\equiv \sum_i \sum_j y_{ij} \log(\gamma_i \theta_i^T \phi_j) - \sum_i \sum_k \theta_{ik}$$

only need to compute at the non-zeros.

This is the (negative) "I-divergence loss" aka "generalized KL".

Various algorithms for efficient MLE, including EM.

"Non-negative matrix factorization" (NMF) [Lee and Seung 2000]

Using the fitted model

- $\hat{\phi}_k \in \Delta_M$ for $k \in [K]$: K learned "genres" of movies

  simplex

- $\hat{\theta}_i \in \Delta_K$: learned user preferences

e.g. recommend $\underset{j : y_{ij} = 0}{\operatorname{argmax}} \sum_k \hat{\theta}_{ik} \hat{\phi}_{kj}$

Bayesian model averaging

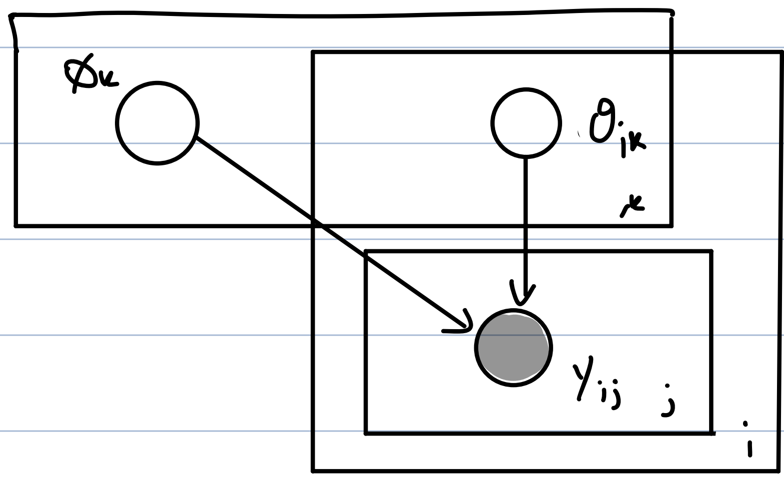
MLE only learns one solution, but there might be many ways to represent overlapping clusters which are each good in different contexts.

Say we have samples from a posterior distribution:

$$(\theta_i^s, \phi_1^s \cdots \phi_k^s) \sim P(\theta_i, \phi_1, \cdots \phi_k \mid Y)$$

then recommend $\underset{j : y_{ij}=0}{\operatorname{argmax}} \frac{1}{S} \sum_s \sum_k \theta_{ik}^s \phi_{kj}^s$

Priors



$\theta_{ik} \sim \text{Gamma}(a, b)$

$\phi_k \sim \text{Dir}(\alpha_1 \cdots \alpha_k)$

Data augmentation and auxiliary variables

$$P(\theta_i \mid \cdots) \underset{\theta_{ik}}{\propto} \text{Gamma}(\theta_{ik} \cdots) \prod_j \text{Pois}(y_{ij}; r_i \theta_i^T \phi_j)$$

Not conjugate (even conditioned on Φ)

However we can re-express the generative process by introducing **auxiliary variables** which will provide conditional conjugacy.

Def'n: Poisson additivity:

if: $y_k \overset{ind.}{\sim} \text{Pois}(\mu_k)$,

$$y_\cdot \triangleq \sum_k y_k$$

then:

$$y_\cdot \sim \text{Pois}(\mu_\cdot)$$

where $\mu_\cdot \triangleq \sum_k \mu_k$

$$y_{ij} \sim \text{Pois}\Big(\sum_k \theta_{ik}\phi_{kj}\Big)$$

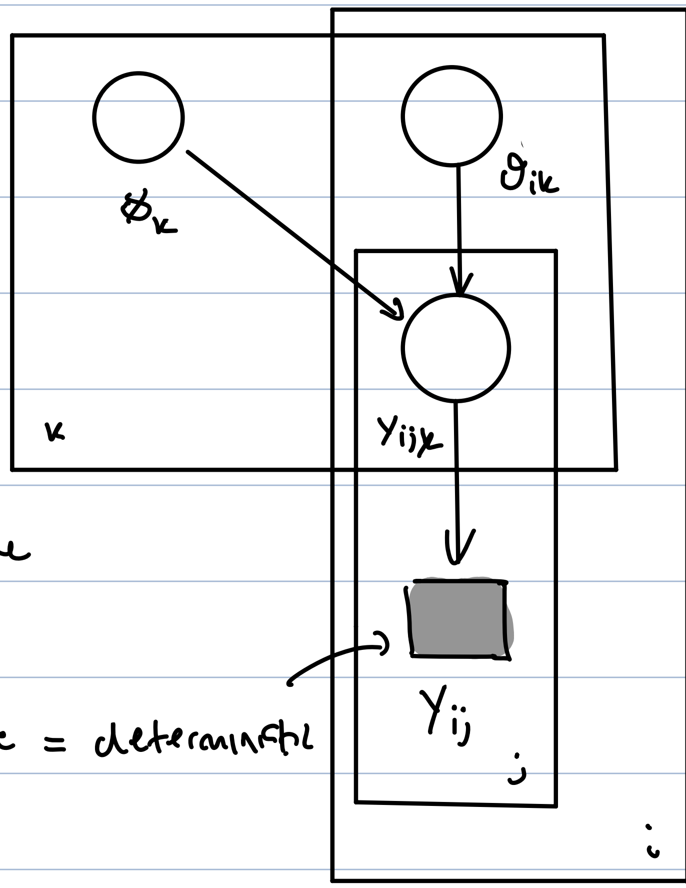
$$\Updownarrow$$

$$y_{ijk} \overset{ind}{\sim} \text{Pois}(\theta_{ik}\phi_{kj})$$

$$y_{ij} = \sum_k y_{ijk}$$

latent sub-counts

Alternative model with auxiliary variables



square = deterministic

The key property is that both this and the original model have the same marginal $P(y_{ij})$.

If y_{ijk} are observed, then:

$$P(\theta_{ik} \mid \cdots) \propto_\theta \text{Gam}(\theta_{ik} \cdots) \prod_j \prod_k \text{Pois}(y_{ijk}; \theta_{ik}\phi_{kj})$$

$$\propto \text{Gam}\Big(\theta_{ik};\, a + \sum_j y_{ijk},\, b + \underbrace{\sum_j \phi_{kj}}_{=1}\Big)$$

Conditional conjugacy after data augmentation.

Notice that $p(\theta_{ik} | \cdots)$ above is the same as it would be in the following model:

$$\theta_{ik} \sim \text{Gamma}(a, b)$$
$$y_{ik} \sim \text{Pois}(\theta_{ik})$$

This is not a coincidence.

Yet another way to write the model:

$$\theta_{ik} \sim \text{Gamma}(a, b)$$
$$y_{ik} \sim \text{Pois}(\theta_{ik})$$
$$(y_{ik1} \cdots y_{ikm}) \sim p(y_{ik1} \cdots y_{ikm} \mid y_{ik\cdot}, \theta_{ik}, \phi_k)$$
$$y_{ij} = \sum_k y_{ikj}$$

what is this dist?

<u>Def'n</u> : Joint distribution of Poissons and their sum.

① $$y_j \overset{ind}{\sim} \text{Pois}(\mu_j) \quad j = 1 \ldots m$$
$$y_\cdot = \sum_j y_j$$

② $$y_\cdot \sim \text{Pois}(\mu_\cdot)$$
$$y_{1 \ldots m} \sim \text{Mult}(y_\cdot, \overbrace{\frac{\mu_1}{\mu_\cdot} \cdots \frac{\mu_m}{\mu_\cdot}}^{= \tilde{\mu}_{1:m}})$$

$$p(y_{1:m}, y_\cdot \mid \mu_{1:m}) =$$

$$\underbrace{p(y_{1:m} \mid \mu_{1:m})}_{= \prod_j \text{Pois}(y_j; \mu_j)} \underbrace{p(y_\cdot \mid y_{1:m}, \mu_{1:m})}_{= \delta(y_\cdot = \sum_j y_j)} = \underbrace{p(y_\cdot \mid \mu_{1:m})}_{= \text{Pois}(y_\cdot; \mu_\cdot)} \underbrace{p(y_{1:m} \mid y_\cdot, \mu_{1:m})}_{= \text{Mult}(y_{1:m}; y_\cdot, \tilde{\mu}_{1:m})}$$

① $$\prod_j \text{Pois}(y_j; \mu_j) \quad \delta(y_\cdot = \sum_j y_j) = \text{Pois}(y_\cdot; \mu_\cdot) \, \text{Mult}(y_{1:m}; y_\cdot, \tilde{\mu}_{1:m})$$ ②

To confirm:

$$\prod_j \frac{\mu_j^{y_j}}{y_j} \cancel{e^{-\mu_j}}\, \delta(\cdots) = \left[\frac{\mu_\cdot^{y_\cdot}}{y_\cdot!} \cancel{e^{-\mu_\cdot}}\right] \frac{y_\cdot!}{\prod_j y_j!} \prod_j \tilde{\mu}_j^{y_j}\, \delta(\cdots)$$

$$\frac{y_\cdot!}{\prod_j y_j!} \prod_j \left(\frac{\mu_j}{\mu_\cdot}\right)^{y_j} \delta(\cdots) = \frac{y_\cdot!}{\prod_j y_j!} \prod_j \tilde{\mu}_j^{y_j}\, \delta(\cdots)$$

Applying this definition to our model:

$$\vartheta_{ik} \sim \text{Gamma}(a, b)$$

$$y_{ik} \sim \text{Pois}(\vartheta_{ik})$$

$$(y_{ik1} \cdots y_{ikm}) \sim \underbrace{p(y_{ik1} \cdots y_{ikm} \mid y_{ik}, \vartheta_{ik}, \phi_k)}$$

$$y_{ij} = \sum_k y_{ikj}$$

$$= \text{Mult}(y_{ik1} \cdots y_{ikm};\ y_{ik}, \cdots)$$

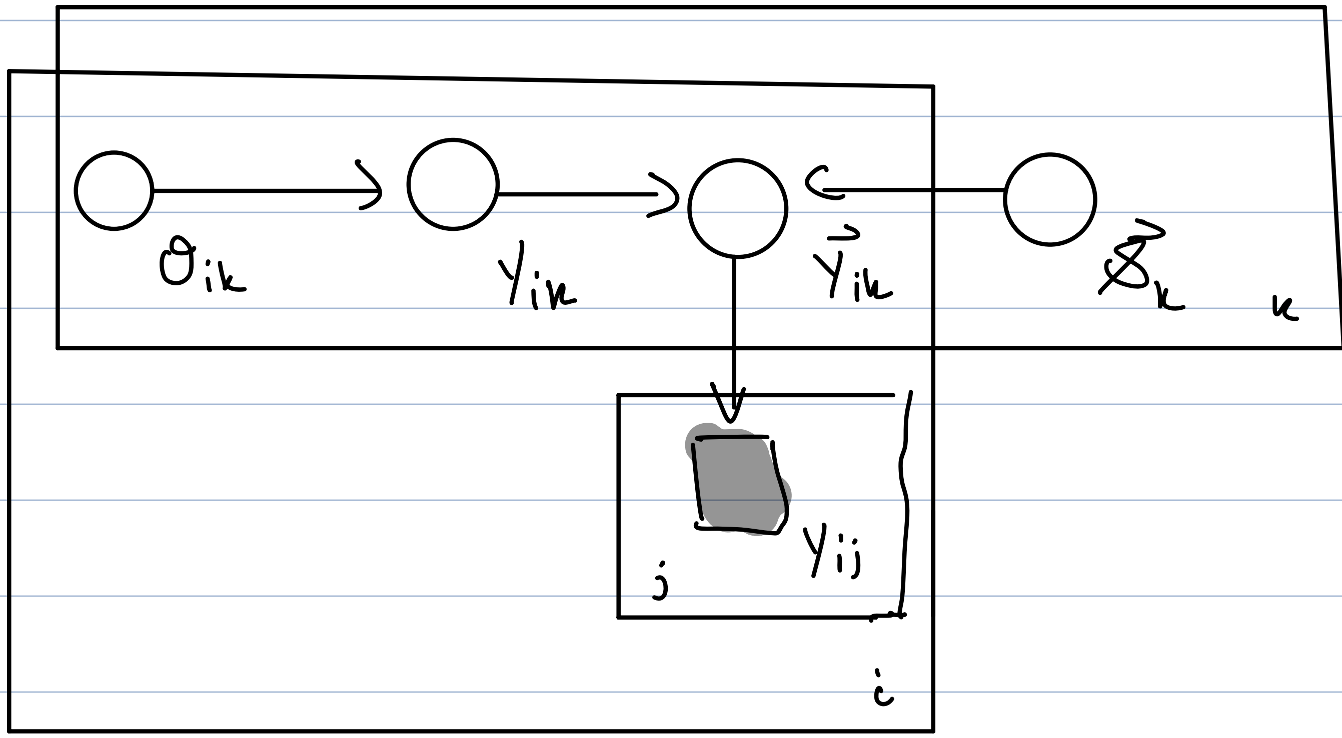
$$\left(\frac{\cancel{\vartheta_{ik}}\phi_{k1}}{\underbrace{\sum_j \cancel{\vartheta_{ik}}\phi_{kj}}_{=1}} \cdots \frac{\cancel{\vartheta_{ik}}\phi_{km}}{\underbrace{\sum_j \cancel{\vartheta_{ik}}\phi_{kj}}_{=1}}\right) = (\phi_{k1} \cdots \phi_{km}) \equiv \phi_k.$$

So:

$$y_{ik} \sim \text{Pois}(\vartheta_{ik})$$

$$y_{ik1} \cdots y_{ikm} \sim \text{Mult}(y_{ik}, \phi_k)$$

The model with all auxiliary variables:



$$\theta_{ik} \sim \mathrm{Gam}(a, b) \qquad \vec{\phi}_k \sim \mathrm{Dir}(\alpha_1 \dots \alpha_m)$$

$$Y_{ik} \sim \mathrm{Pois}(\theta_{ik}) \qquad \vec{Y}_{ik} \sim \mathrm{Mult}(Y_{ik}, \vec{\phi}_k)$$

$$y_{ij} = \sum_k y_{ijk}$$

conditionally conjugate!

Gibbs sampler:

$$\theta_{ik}^s \sim \mathrm{Gamma}\left(a + \sum_j y_{ijk},\ b + 1\right)$$

$$\phi_k^s \sim \mathrm{Dir}\left(\alpha_1 + \sum_i y_{i1k}, \dots \alpha_m + \sum_i y_{imk}\right)$$

$$(y_{ij1}^s \cdots y_{ijk}^s) \sim \mathrm{Mult}\left(y_{ij},\ \frac{\theta_{i1}^s \phi_{1j}^s}{\theta_i^T \phi_j} \dots \frac{\theta_{ik}^s \phi_{kj}^s}{\theta_i^T \phi_j}\right)$$

Auxiliary variable MCMC

At a high level, all MCMC methods generate samples of latent variables:

$$z^s \sim p(z \mid Y)$$

such that:

$$\lim_{S \to \infty} \frac{1}{S} \sum_s \mathbb{1}(f(z_s) \in B) \to P(f(z) \in B \mid Y)$$

Auxiliary variables are variables A such that

$$P(Z, Y, A) = \int p(Z, Y, A = a)\, da$$

Adding such variables into MCMC is always valid:

Define: $f(Z, A) = Z$

$$\frac{1}{S} \sum_s \mathbb{1}(f(Z^s, A^s) \in B) \to P(Z \in B \mid Y).$$

e.g. Gibbs:

$\| \; P(Z_i \mid Z_{\setminus i}, Y)$ intractable

$\| \; P(Z_i \mid Z_{\setminus i}, Y, A)$ tractable

$ \; P(A \mid Z, Y)$ tractable