Recent Works in Multitask Learning

Ayan Acharya, Anish Mittal

UT Austin

Oct 25, 2011

Multiple Tasks Occur Naturally

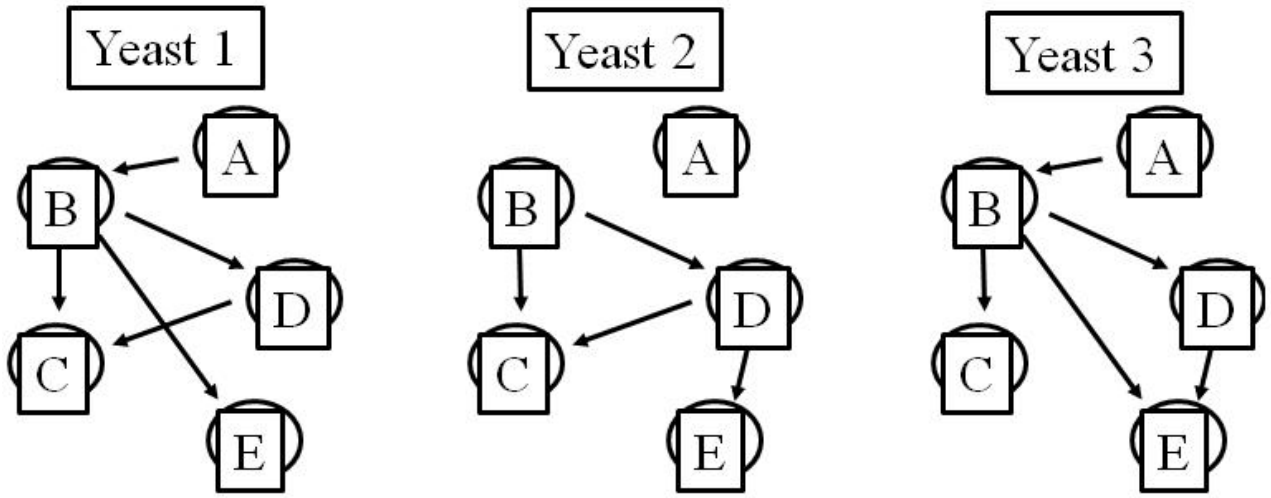
Mitchell's Calendar Apprentice (CAP)

- Time-of-day (9:00am, 9:30am, ...)
- Day-of-week (M, T, W, ...)
- Duration (30min, 60min, ...)
- Location (Tom's office, Dean's office, 5409, ...)

[1]

[1] Credits: Rich Caruana, Computer Science Department, Cornell University

MTL for Bayes Net Structure Learning

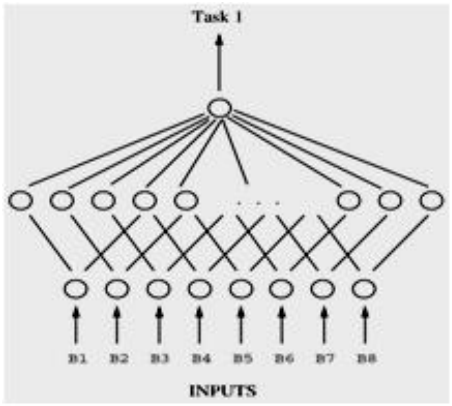


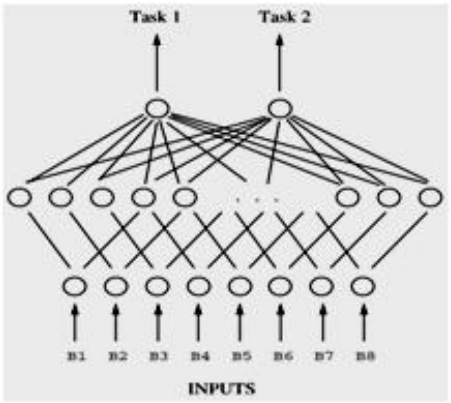
- Bayes Nets for these three species overlap significantly
- Learn structures from data for each species separately? No.
- Learn one structure for all three species? No.
- Bias learning to favor shared structure while allowing some differences? Yes – makes most of limited data.

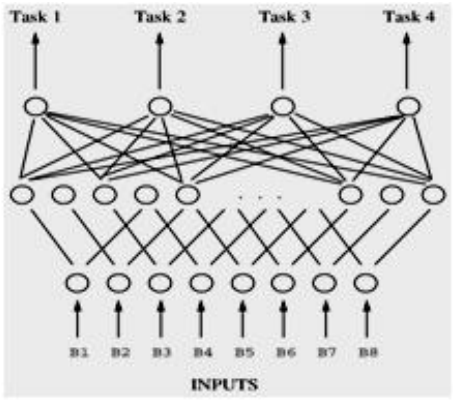
2

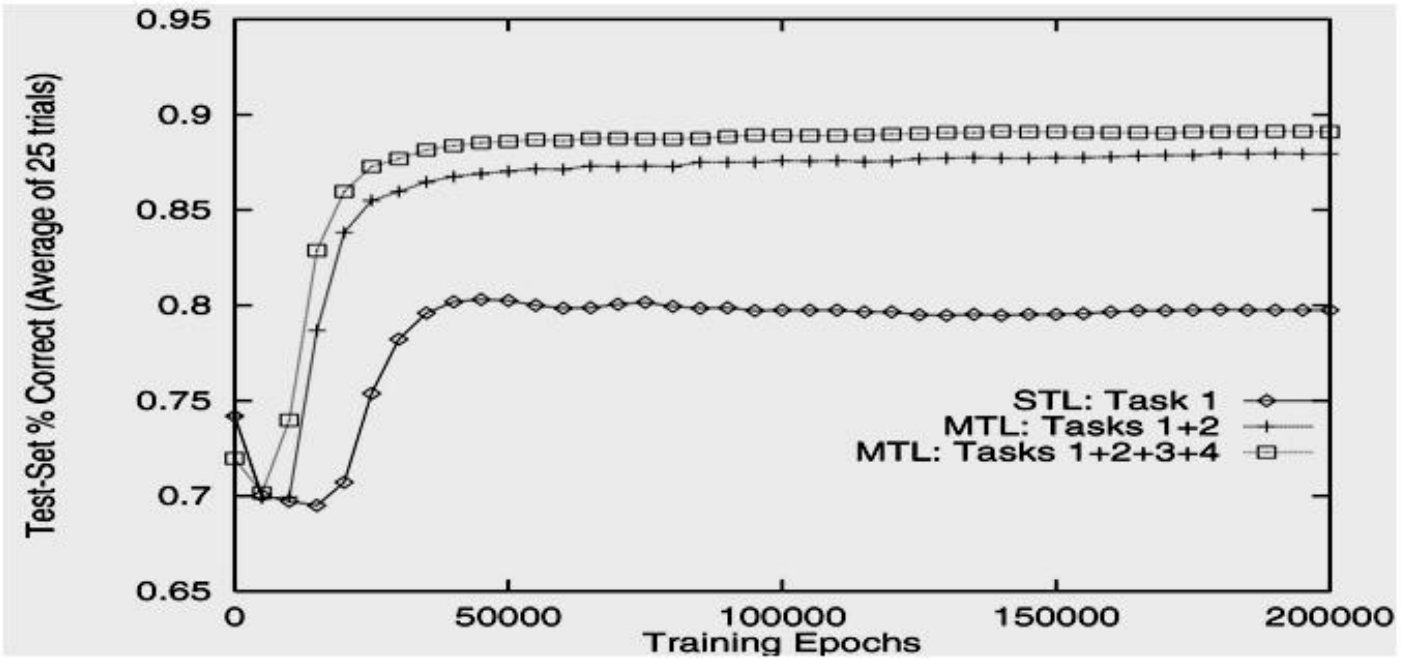
[2] Credits: Rich Caruana, Computer Science Department, Cornell University

1 Task vs. 2 Tasks vs. 4 Tasks

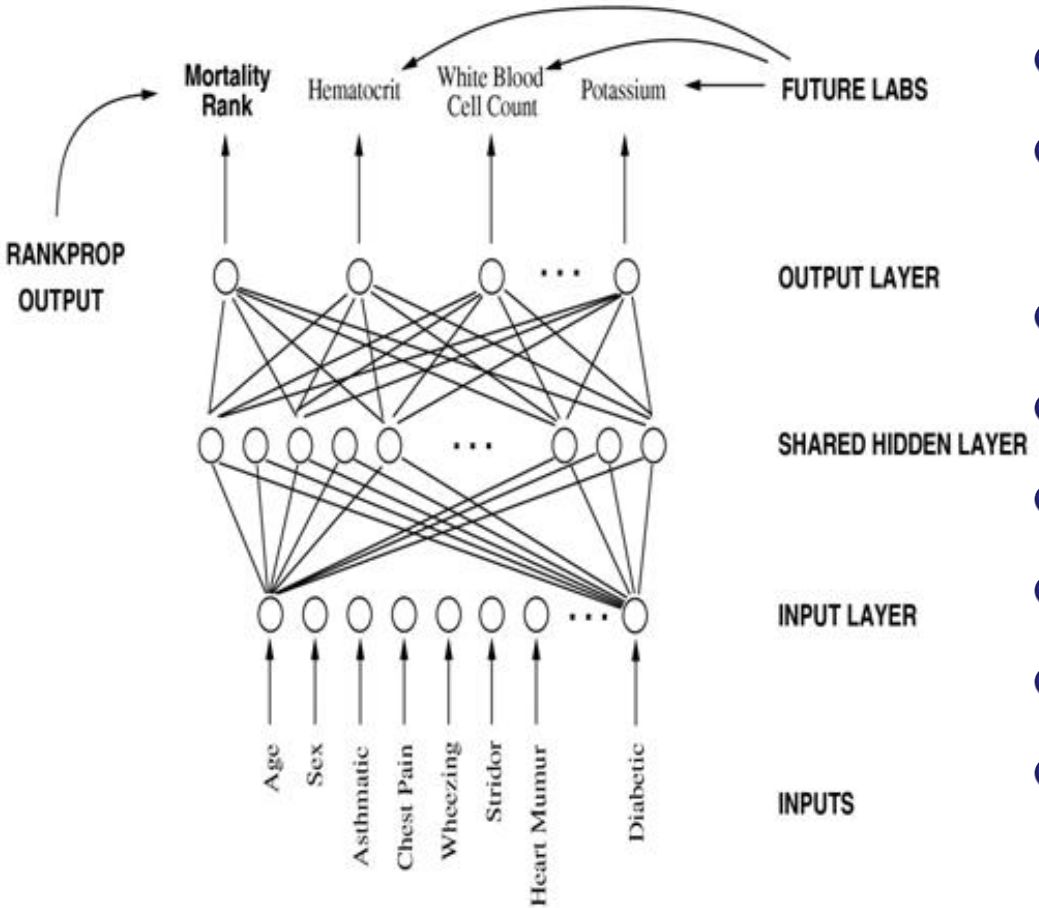








Using Future to Predict Present

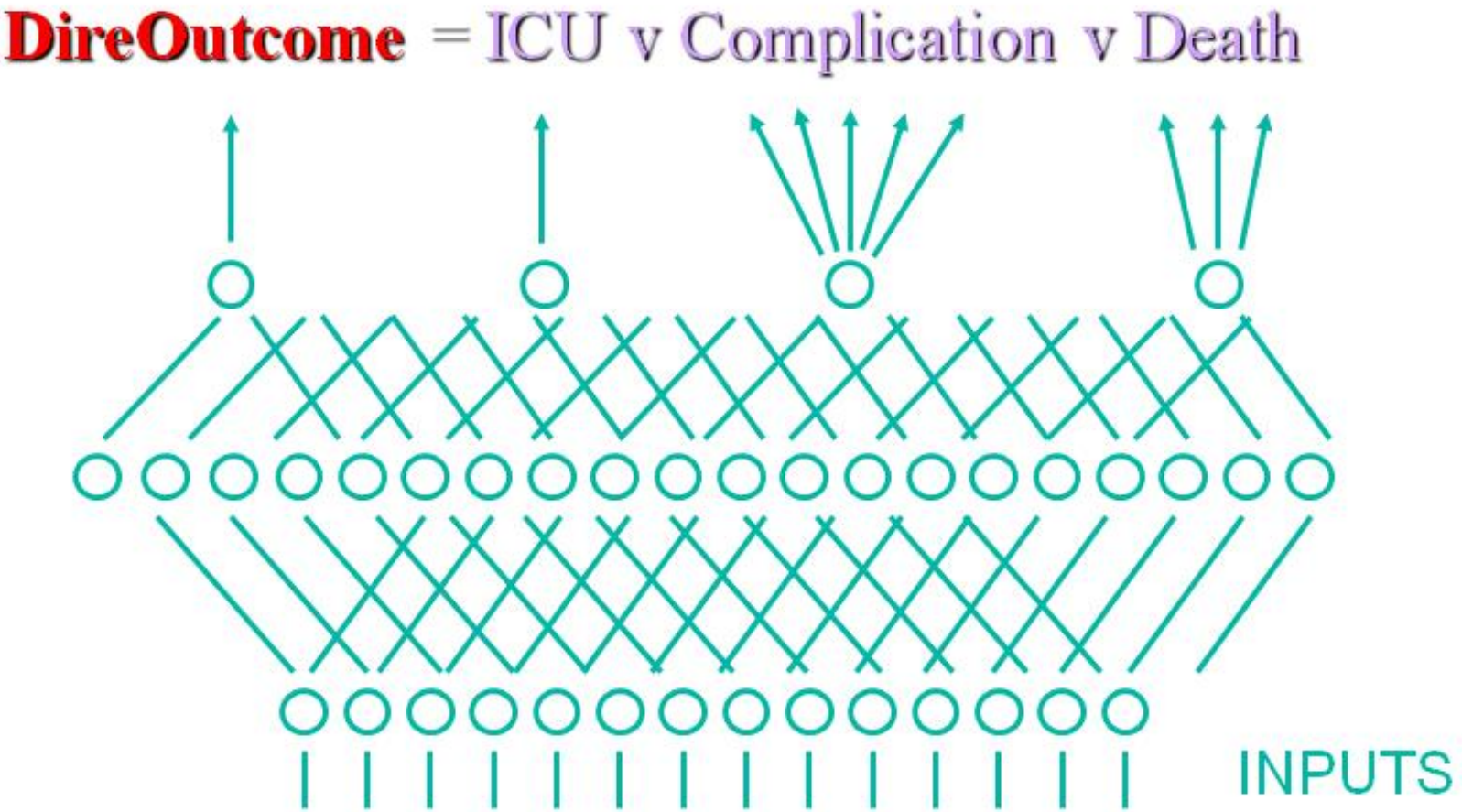


- medical domains
- autonomous vehicles and robots
- time series
- stock market
- economic forecasting
- weather prediction
- spatial series
- many more

4

[4] Credits: Rich Caruana, Computer Science Department, Cornell University

Helpful for decomposable Tasks



5

[5] Credits: Rich Caruana, Computer Science Department, Cornell University

Parallel vs. Serial Transfer

- Where possible, use parallel transfer
- All info about a task is in the training set, not necessarily a model trained on that train set
- Information useful to other tasks can be lost training one task at a time
- Tasks often benefit each other mutually
- When serial is necessary, implement via parallel task rehearsal
- Storing all experience not always feasible

6

[6] Credits: Rich Caruana, Computer Science Department, Cornell University

Papers Covered

1. Transfer Learning for Collective Link Prediction in Multiple Heterogeneous Domains. B. Cao, N. Liu, Q. Yang.
2. Multiple Domain User Personalization. Y.Low, D. Aggarwal, A. Smola.
3. Clustered Multi-Task Learning: A Convex Formulation. Laurent Jacob, Francis Bach, Jean-Philippe Vert.

Other Related Papers:

1. Localized Factor Models for Multi-Context Recommendation. D. Agarwal, B-C Chen, B. Long.
2. Flexible Latent Variable Models for Multitask Learning. J. Zhang, Z. Ghahramani, Y.Yang.
3. One-Shot Learning with a Hierarchical Nonparametric Bayesian Model. R. Salakhutdinov, J. Tenenbaum, A. Torralba.

Other Potential Approach: Learning Structural SVMs with Latent Variables. C-N. J. Yu, T. Joachims.

Clustered Multi-Task Learning: a Convex Formulation L. Jacob, F. Bach and J-P. Vert

Motivation

- Can sharing information across related tasks help ?
- Sharing achieved using apriori information about weight vectors associated with each task
- Similar tasks should have similar weight vectors
- *Which tasks are similar* can be learnt together with *weights* using convex optimization formulation

Motivation

l^p norms used to impose various sparsity patterns in data while learning weights

Can regularization function be designed suited to the problem assuming a prior knowledge?

Motivation

Objective = L(W) + $\lambda\Omega(W)$

Empirical risk of set of linear classifiers given in matrix W

$$L(W) = \frac{\Sigma_{t=1}^{N}\Sigma_{i\epsilon I(t)} l(w_t^T x_i, y_i)}{n}$$

Regularizer $\Omega(W)$ learnt from prior knowledge to constrain sharing of info across tasks

λ controls the relative weighting of loss function and regularizer

Regularizer

Assuming that we know how tasks are partitioned in to clusters: $\Omega(W)$ consists of

- Global Penality: how large are the weight vectors: tr(WUW^T)
- Between Cluster Variance: how close clusters are to each other: tr(W(M-U)W^T)
- With in Cluster Variance: how compact are clusters: tr(W(I-M)W^T)

U is a mean matrix with all entries equal to inverse of number of tasks
M is normalized adjacency matrix with both rows and columns summing to 1

$$\Omega(W) = \epsilon_M \Omega_{mean}(W) + \epsilon_B \Omega_{between}(W) + \epsilon_W \Omega_{within}(W)$$

Objective

Objective = L(W) + λ tr(WΣ^{-1}W^T)

where $\Sigma^{-1} = \epsilon_M$U + ϵ_B(M-U) + ϵ_W(I-M)
Σ^{-1} is a quadratic penality depending on the normalized adjacency matrix M
ϵ_M, ϵ_B , ϵ_W can balance the importance of components of the penality

Effect of ϵ_M, ϵ_B , ϵ_W

- $\epsilon_M = \epsilon_B = \epsilon_W$
  Doesn't put any constraint on relationship between tasks
- $\epsilon_B = \epsilon_W > \epsilon_M$
  Global similarity between tasks is enforced in additional to constraint on mean. Also, structure in clusters play no role
- $\epsilon_W > \epsilon_B = \epsilon_M$
  Penalise the norm and their variance
- Optimum $\epsilon_W > \epsilon_B > \epsilon_M$ Penalize more with in cluster variance than between cluster variance promoting compact clusters

Convex relaxation

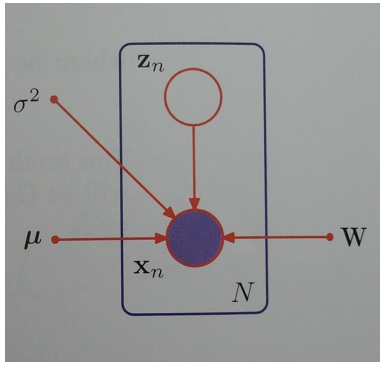
- Σ^{-1} is dependent on normalized adjacency matrix M whose values are quantized so as to make sum of rows and columns to be 1
- Values assume discrete values by construction making the problem non-convex, hence necessary to relax the assumption
- After convex relaxation, the set S_c for Σ_c can be expressed as
  $S_c = \{\Sigma_c \epsilon S_+^m : \alpha I \leq \Sigma_c \leq \beta I, tr\Sigma_c = \gamma\}$
  $\alpha = \epsilon_W^{-1}, \beta = \epsilon_B^{-1} and \gamma = (m - r + 1)\epsilon_W^{-1} + (r - 1)\epsilon_B^{-1}$

Reinterpretation in terms of norms

Depending on the constraints on set S_c , different norms on W can be obtained and all multi- task formulations can be cast in this framework

Transfer Learning for Collective Link Prediction in Multiple Heterogeneous Domains.
B. Cao, N. Liu, Q. Yang.

Probabilistic PCA (Tipping & Bishop, 1999)



- $z \sim \mathcal{N}(0, I)$. ($z \in \mathbb{R}^M$)
- $x \sim \mathcal{N}(Wz + \mu, \sigma^2 I)$ ($x \in \mathbb{R}^D$ and $W \in \mathbb{R}^{D \times M}$)
- $p(x|\mu, W, \sigma) = \mathcal{N}(\mu, C)$ where $C = WW^\dagger + \sigma^2 I$.

ML estimates of model parameters are:

- $\mu_{ML} = \bar{x}$, $\sigma^2_{ML} = \frac{1}{D-M} \sum_{i=M+1}^{D} \lambda_i$.
- $W_{ML} = U_M(L_M - \sigma^2 I)^{1/2} R$, where, $U_M \in \mathbb{R}^{D \times M}$ and $L_M \in \mathbb{R}^{M \times M}$ (diagonal matrix) – catch – W_{ML} spans the principal subspace of the data.

Dual Probabilistic PCA (Lawrence, 2005)

Can we marginalize out parameters and maximize the likelihood over hidden variables?

- $W \sim \prod_{j=1}^{D} \mathcal{N}(w_j|0, I)$. ($W \in \mathbb{R}^{D \times M}$)
- $z \sim \mathcal{N}(0, I)$. ($z \in \mathbb{R}^{M}$)
- $x \sim \mathcal{N}(Wz + \mu, \sigma^2 I)$ ($x \in \mathbb{R}^{D}$)
- $p(X|\mu, Z, \sigma) = \prod_{j=1}^{D} \mathcal{N}(x_{:,j}|\mu_j, C)$ where $C = ZZ^{\dagger} + \sigma^2 I$.

Turns out that DPPCA also has similar interpretation as PCA when we take MAP estimates of Z. Marginalizing over both W and Z leads to Bayesian PCA (Bishop, 1999) – analytically intractable.

Notations

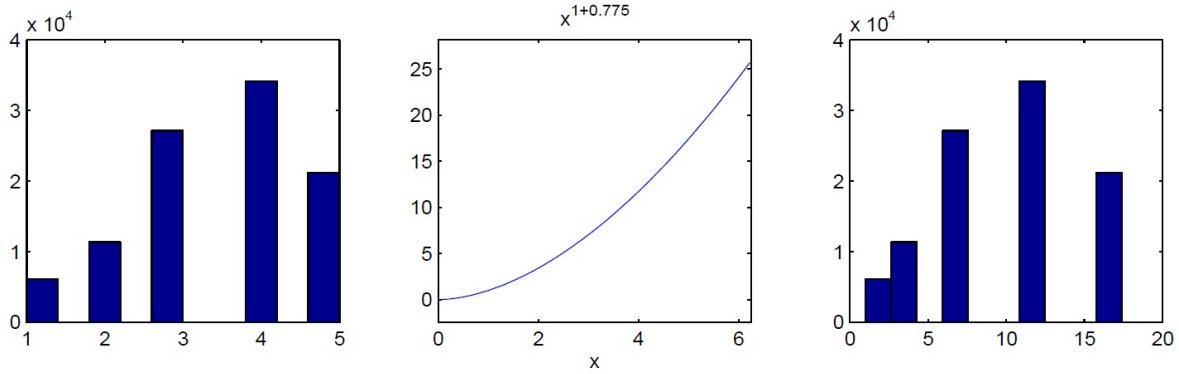
- $\{X^{(t)}\}_{t=1}^{T}$: Collection of matrices across different tasks – subset of which are the observed values. $X^{(t)} \in \mathbb{R}^{m \times n}$.
- $Y : X = f(Y)$, where f is some suitable transformation (link function) over Y – depends on distribution of observed X's. $Y \in \mathbb{R}^{m \times n}$.
- $U \in \mathbb{R}^{m \times d}$ is the entity latent factor matrix of first type (users).
- $V \in \mathbb{R}^{n \times d}$ is the entity latent factor matrix of second type (items).
- Objective: predicting missing values in $\{X^{(t)}\}$.

Non-linear Matrix Factorization

- PMF: $p(Y|U, V, \sigma^2) = \mathcal{N}(UV^\dagger + E)$ where, $E \sim \mathcal{N}(0, \sigma^2)$, $U \sim \mathcal{N}(0, \beta_u^{-1})$, and $V \sim \mathcal{N}(0, \beta_v^{-1})$.
- Optimize over U, V and all model parameters – how about marginalizing over either U or V?
- $p(Y|V, \sigma^2, \beta_u) = \prod_{i=1}^{m} \mathcal{N}(y_{i,:}|0, \beta_u^{-1} VV^\dagger + \sigma^2 I_n)$ – PPCA.
- $p(Y|U, \sigma^2, \beta_v) = \prod_{j=1}^{n} \mathcal{N}(y_{:,j}|0, \beta_v^{-1} UU^\dagger + \sigma^2 I_m)$ – Dual PPCA.
- Inner product allows kernelization – non-linear matrix factorization (Lawrence & Utrasun, 2009) –

  $p(Y|V, \sigma^2, \beta_u) = \prod_{i=1}^{m} \mathcal{N}(y_{i,:}|0, K + \sigma^2 I_n)$.

Negatively Skewed Distribution and Link Function



- Skewness: $\mathbb{E}\left[\frac{X-\mu}{\sigma}\right]^3$.
- $g(x) = f^{-1}(x) = x^{1+\alpha}$ where $\alpha \geq 0$.
- $p(X|V, \sigma^2, \beta_u) = \prod_{i=1}^{N} \mathcal{N}(g(x_{i,:})|0, K + \sigma^2 I_N)|g'(x_{i,:})|$.
- Connections established so far: PMF $\rightarrow$ PPCA (and DPPCA) $\rightarrow$ Non-linear MF $\rightarrow$ Non-linear MF with link function.

Collective Link Modeling

- Multiple "related" tasks where one entity type is common (U) and there are multiple matrices $\{V^t\}$ of other entity type.
- Naïve option: model each task independently –

$$p(\{Y^{(t)}\}|V, \sigma^2, \beta_u) = \prod_{t=1}^{T} \prod_{i=1}^{m^{(t)}} \mathcal{N}(g(x_{i,:}^{(t)})|0, K^{(t)} + \sigma^2 I_n).$$

- Smarter option: joint modeling of the tasks –

$$p(\{Y^{(t)}\}|V, \sigma^2, \beta_u) = \prod_{i=1}^{m} \mathcal{N}(g(X_{i,:})|0, C), \text{ where } C = T \otimes K + \sigma^2 I.$$

- $\langle v^{(s)}, v^{(t)} \rangle = T_{s,t} k(v^{(s)}, v^{(t)})$, $T = LL^{\dagger}$.
- Task specific link function: $g^{(t)}(x) = c^{(t)} x^{1+\alpha^{(t)}} + b^{(t)}$, where $c^{(t)}, \alpha^{(t)} > 0$.

Collective Link Modeling Continued

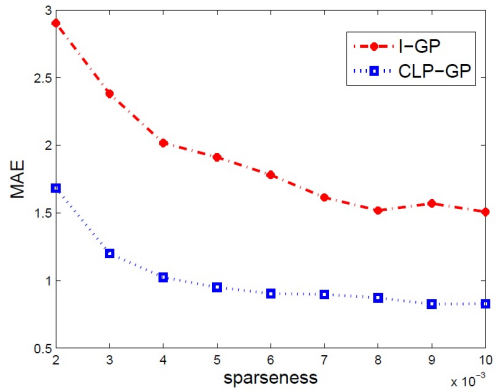
- $\mathbb{E}(y) = T_{t,t} \sum_{x_j \in X^{(t)}} w_j k(v, v_j) + \sum_s T_{s,t} \sum_{x_i \in X^{(s)}} w_j k(v, v_i)$ where $w_i = \left(C_{\mathbb{O}}^{-1} k_y\right)_i$.
- Parameters (T, kernel parameters, $\{c^{(t)}, b^{(t)}, \alpha^{(t)}\}$ and V) are learnt using stochastic gradient descent.
- Steps are expensive as each of them involves matrix inverse ($C_{\mathbb{O}}^{-1}$).

Experiments and Results

Three datasets – MovieLens, Book-Crossing, Douban.

MovieLens	I-GP	M-GP	CMF	CLP-GP
-Link	1.4827	0.6569	0.7120	0.6440
+Link	1.3487	0.6353	-	0.6385
Book-Crossing	I-GP	M-GP	CMF	CLP-GP
-Link	0.9385	0.7018	0.8054	0.6547
+Link	0.9317	0.6488	-	0.6014
Douban	I-GP	M-GP	CMF	CLP-GP
-Link	0.7789	0.7772	0.9917	0.7446
+Link	0.7726	0.7625	-	0.7418

MAE for i) Independent Link Prediction using NMF via GP (I-GP), ii) Joint Link Prediction using multi-relational GP (G-MP), iii) CMF, iv) CLP-GP.



The influence of sparseness on MovieLens dataset.

Multiple Domain User Personalization.
Y.Low, D. Aggarwal, A. Smola.