Bayesian semi-parametric methods for multi-task learning

Balaji Krishnapuram
Computer-Aided Diagnosis and Therapy
Siemens Medical Solutions, Inc.

Collaborators: Ya Xue, Xuejun Liao, Larry Carin [Duke]

July 19, 2005
Multi-task learning

- In statistical learning, we are often short of data due to high acquisition costs.
  - Sometimes addressed by adaptive (i.e. active) sensing

- Human beings transfer the (inductive) skills learnt while solving one problem to help them solve a different problem
  - After learning to ride a bicycle, the skills learnt in this process greatly reduce the amount of time/effort required for learning to ride a motorcycle: Balance, navigation, traffic rules etc

- How can we use the idea of inductive transfer to improve the accuracy of statistical learning systems?
  - Collaborative filtering: Principled methods for pooling datasets

- For want of time, we will not discuss the other major statistical model for multi-task learning
  - Joint prediction of outputs: exploit correlations between prediction tasks
Pooling data: Collaborative Filtering

- **Recommender systems: books, movies, art…**
  - Use content description (features) and similarity between users’ preferences.

- **Statistical pooling of datasets in other contexts**
  - Radar/Sonar data collected under different environmental conditions
  - Medical data collected from different geographical locations, using different imaging systems etc

- **Common question:**
  - How can the data be pooled in a statistically principled way?
  - Which datasets are similar enough to be grouped together?
Visual Illustration

Red: +  Blue: -  *: training data  ·: testing data
Visual Illustration

Test results: dataset 1
- Classifier trained with each dataset alone
- Classifier trained with all dataset jointly

Test results: dataset 2
- Classifier trained with each dataset alone
- Classifier trained with all dataset jointly
Visual Illustration

Test results: dataset 3
Classifier trained with each dataset alone
Classifier trained with all dataset jointly

Test results: dataset 4
Classifier trained with each dataset alone
Classifier trained with all dataset jointly
Visual Illustration

Test results: dataset 5
- Classifier trained with each dataset alone
- Classifier trained with all dataset jointly

Test results: dataset 6
- Classifier trained with each dataset alone
- Classifier trained with all dataset jointly

©2005 Siemens Medical Solutions. All rights reserved. Siemens confidential
Why does it work?

Training data from 6 datasets
Statistical approach: Intuitive picture

- $x, y$ are **conditionally independent** if I tell you the value of $z=3$ (here cluster indicator)

\[ p(x, y | z) = p(x | z) p(y | z) \]

- If I don’t tell you the value of $z$, then $x, y$ are **dependent** i.e. $p(y | x)$ depends on value of $x$!

\[ p(x, y) = \int p(x | z) p(y | z) p(z) \, dz \]
Simplistic model: all tasks *similar*

- If the prior $p(w|\theta)$ is fixed (i.e. known $\theta$), learning each classifier is independent
  - Can only use $D_i$ to estimate $w_i$
- If the prior is enforced as the same for all classifiers, *but not explicitly specified*, then $w_1$, $w_2$, … $w_{20}$ not independent!
  - E.g., After seeing only $D_1$, $D_2$, $D_3$ posterior of hyper-parameters $p(\theta|D_1, D_2, D_3)$ changes. Thus, while estimating $w_4$, we will effectively use $D_1$, $D_2$, $D_3$ also
  - As a result, all available data used to estimate any single $w_i$!!

![Diagram](image-url)
Modeling groups of similar tasks

Problems:
- Which tasks are similar (i.e. cluster together)?
  - Above, the clusters are \{1,2,3\} and \{4,5,6\}
- How many clusters?
- Given limited training data, how to handle rare sub-groups that have not yet been seen?
Dirichlet Process

- A priori, no reason to believe that real world is simple
  - Why should there be only a small number of clusters of tasks?
- Rarely seen user preference types may only be noticed after obtaining a large number of training samples.
- Bayes: Consider the probability distribution over every possible level of model complexity (number of groups)
  - The a-posteriori distribution over model complexity changes as more and more data becomes available.
  - But marginalize (integrate) over this distribution while making predictions!
- In our model,
  - allow a separate hyper-parameter ($\theta_i$) for every classifier $w_i$
  - $p(\theta_i) = \text{Dirichlet-Process}(G, \alpha)$.
  - Think of distribution $p(\theta_i)$ as weighted sum of countably-infinite delta functions [We will come back to it in a moment]
What is a Dirichlet Process?

Consider distribution $p(\pi, \mu, \Sigma)$ over the space of all Gaussian mixture models (for simplicity, treat covariance $\Sigma$ as constant)

- GMM: $p(x|\pi, \mu) = \Sigma \pi A N(x|\mu A, \Sigma)$;

One such hierarchical distribution with probability mass concentrated on countably-infinite number of components is

- $p(x|\mu, \Sigma) = N(x|\mu, \Sigma)$;
- $P(\mu) = \text{DP}(G_0, \alpha) = \Sigma \pi A \delta(\mu - \mu A)$

Sampling this DP to obtain a GMM involves:

- $p(\pi) \rightarrow$ Stick-breaking interpretation:
  - Break a unit length stick by repeatedly sampling $\text{Beta}(\alpha,1)$
  - $\mu A \sim G_0$; e.g. Base-distribution $G_0=N(0, 1000*I)$
Proposed model:

- For dataset # k, sample # i,
  - \( p(y_k^i|x_k^i, w_k) = \sigma [y_k^i (w_k^T x_k^i)] \), \( y_k^i \in \{\pm 1\} \)
  - \( p(w_k|\theta_k) = \mathcal{N}(w_k| \mu_k, \text{diag}(\lambda_k)) \);
  - \( \theta_k = \{ \mu_k, \text{diag}(\lambda_k) \} \)
  - \( p(\mu_k, \text{diag}(\lambda_k)) = \text{DP}(G, \alpha) \)
  - \( p(\alpha) = \text{Gamma}(\alpha | a_\alpha = 10^{-3}, b_\alpha = 10^{-3}) \)
  - \( p(G) = \text{Normal}(m=0, 100*C) \text{Inverse-Gamma}(C_{ii} | a=10^{-3}, b=10^{-3}) \)

- Posterior distributions can be easily obtained by Gibbs sampling, but this is too slow to be practical for large datasets

- In practice, we used the Variational-Bayesian EM algorithm
  - an approximate, but very computationally efficient alternative
  - Allows us to efficiently estimate an approximate posterior distribution for all the model parameters, even for very large datasets
Results: Collaborative filtering

- 642 paintings in gallery web site, each characterized by a 263 dimension feature vector
- 190 visitors rated images according to their preferences
- Each user rated average of 89 images (range 5 to 300)
- Leave-one-out: treat one user at a time as a test user, keeping others as completely available training data.
- For the test user keep a part of his ratings also as a training set, and the remainder as the test set for him.

Compare against 2 alternatives:
- Single task learning from only the data available from the test user
- Multi-task learning where classifiers for all users share a single prior
  - i.e. claim there is only one user “type” or cluster
Collaborative recommendation

![Graph showing precision on top 20 images vs observed examples. The graph compares Collaborative Ensemble Learning, Content-based Filtering, and Collaborative Filtering.](image)
Studies on Simulated data

True Mean and Variance of All Tasks

Number of Tasks = 6

Pooling based on DP - VB
Pooling all data - RBF kernel
No pooling

Number of training samples for each task

Error rate
Studies on Simulated data

Number of Tasks = 4

- Pooling based on DP - VB
- Pooling all data - RBF kernel
- No pooling

Number of Tasks = 20

- Pooling based on DP - VB
- Pooling all data - RBF kernel
- No pooling
Conclusions

- Multi-task learning improves the accuracy of model estimation when training data is
  - Costly, thus in short supply
  - available in chunks, each chunk known to be from a single distribution
- Relaxes assumption that all data is sampled from the same generative mechanism
  - Relies on an inductive bias that hastens the learning curve
- Most useful when limited training samples are available, but data or model is of high dimensionality
  - In the limit of large sample sizes, the proposed non-parametric model still does not deteriorate performance significantly
- The clustering patterns identified among tasks was biologically sensible on real life medical problems (not presented here).
- Proposed VB implementation is both fast and accurate
  - Even faster than learning a single classifier after pooling all samples!