Incremental EM for Latent-Class Relational Models

Michael Chiang MCHC@CS.UBC.CA
David Poole POOLE@CS.UBC.CA
Computer science, 201-2366 Main Mall, Vancouver, B.C., V6T 1Z4, Canada.

Abstract

Examples of latent-class relational models have found much use in key areas such as recommender systems and social network analysis, where hidden group information about domain individuals can help improve predictive accuracy. Fitting such models, however, is generally infeasible due to the large number of latent quantities that must be inferred, necessitating approximate inference techniques. In this work, we show how an instance of Neal and Hinton's *incremental* EM algorithm can be derived for fitting latent-class relational models tractably. Empirical evaluations show good accuracy of models learned with the proposed algorithm in social networks and movie rating domains.

1. Latent-class Relational Models

Latent-class relational models (LRM) are lifted models of relational data that model non-observable class membership of individuals in addition to the observable relationships amongst individuals. Examples of LRMs, e.g. (Xu et al., 2006; Kemp et al., 2006), have found much use in modelling complex relational domains. Fitting LRMs to data generally require approximate inference, and in this work we show how this task can be addressed in a simple way using the *incremental EM* framework of Neal and Hinton (1998).

Relational domains that LRMs model are characterised by a collection of sets of objects/individuals $\mathbf{O} = \{O_1, \ldots, O_m\}$ and a set of relations $\mathbf{R} = \{r_1 \ldots r_n\}$ defined over elements of $\mathbf{O}$. A relation $r \in \mathbf{R}$ is defined as $r : D_1 \times \ldots \times D_k \rightarrow \{v_1, \ldots, v_{|r|}\}$, where k is the arity of r, $D_1 \ldots D_k \in \mathbf{O}$, and there are $|r|$ truth values of r represented by $v_1, \ldots, v_|r|$. A simple example may be the binary movie-rating relation $likes : User \times Movie \rightarrow \{true, false\}$, where $User, Movie$ denote the set of user and movies respectively, and $likes(x, y)$ can be true or false for each user-movie pair (x, y).

For a given relational domain, the LRM includes each $r \in \mathbf{R}$ and assigns an accompanying parameter θ_r to characterise the distribution over values of r. For each object domain $D \in \mathbf{O}$, a unary latent relation[1] $h_D : D \rightarrow \{c_1, \ldots c_{g_D}\}$ is introduced, and accompanied by parameter θ_{h_D}. g_D represents the number of latent classes/groups that elements of D can belong to, and is itself a random quantity, probabilistically distributed over $\mathbb{Z}^+$.

The grounding of an LRM can be seen as a directed graphical model whose nodes represent instantiations of all observed and latent relations. Each observed node representing some ground relation has latent parents matching individuals the ground relation implicates. For example, $r(x', y')$ where $x' \in X, y' \in Y$ has latent parents $h_X(x')$ and $h_Y(y')$. Parameter-sharing is assumed for all instances of the same relation. Latent class parameters $g_{(\cdot)}$ are parents of the ground instances of latent relations and their parameters. For a simple example domain where $\mathbf{O} = \{X, Y\}$ and $\mathbf{R} = \{r\}$ where $r : X \times Y \rightarrow \{v_1, \ldots, v_{|r|}\}$, Fig. 1 illustrates the corresponding LRM using plate notation.

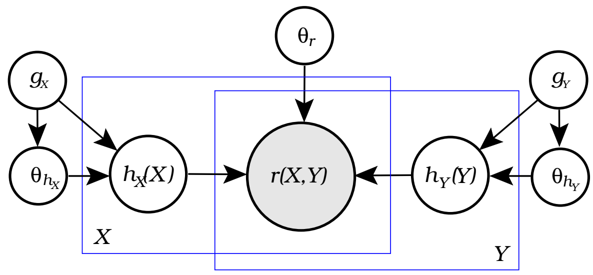

Figure 1. The LRM for a domain consisting of a single relation r over domains X and Y.

Our LRM description can be seen as an alternative description of the LRMs proposed in (Kemp et al., 2006; Xu et al., 2006), although we define the distribution over the number

[1] Only unary latent relations are considered, representing latent class membership of individuals. Latent relations of higher arity are beyond the scope of this work.

Presented at *ILP-MLG-SRL*, Leuven, Belgium, 2009.

of latent classes is defined parametrically instead of via the Chinese Restaurant Process as in (Kemp et al., 2006; Xu et al., 2006).

2. Incremental EM for LRMs

Estimating parameters of an LRM using the conventional approach of EM (expectation-maximisation, Dempster et al. (1977)) in general incurs an intractable expectation step. However, it turns out that *incremental EM* due to Neal and Hinton (1998) represents an appropriate and simple approximate algorithm for this task (we shall use iEM for short). In the remainder, we assume a simpler LRM with fixed g_D for all $D \in \mathbf{O}$ and focus on iEM's ability to find good LRM parameters.

Like EM, an iEM iteration consists of an expectation (E) step followed by the maximisation (M) step. In the E step, the marginal posterior is computed for each latent variable z whilst fixing all other latent variables $z' \neq z$ to their previously calculated distributions. Given the marginal posteriors from the E step, the M step sets model parameters to maximises the expected-likelihood of data with respect to the distributions obtained in the E step.

The applicability of iEM for LRMs becomes evident by observing that the marginal probability of a variable is dependent on its *Markov blanket* only, when all latent variables in the model have fixed distributions. In other words, statistical influences external of the Markov blanket are effectively blocked. As such, expensive global marginalisation that renders expected-likelihood intractable is replaced by efficient local marginalisation.

The Markov blanket approximation results in a factorised expected-likelihood of data. To see this, first we define some notation. Let $\mathbf{Y}$ be a set of random variables, where each element represents an observed instance (ground relation) of a relation $r \in \mathbf{R}$. (Non-observed cases can be omitted directly via the "missing at random" assumption.) Similarly, let $\mathbf{Z}$ denote the set of latent variables corresponding to ground instances of the latent relations.

For some $X \in \{\mathbf{Y}, \mathbf{Z}\}$, let $B_o(X) \subseteq \mathbf{Y}$ and $B_l(X) \subseteq \mathbf{Z}$ be the observed and latent variables in the Markov blanket of X, and $b_0(X), b_l(X)$ the corresponding instantiations. Given the observation $\mathbf{Y} = \mathbf{y}$, factorisation of the expected-likelihood is shown as follows

$$\begin{aligned} L(\theta) &= \log \sum_{\mathbf{z}} P(\mathbf{y}, \mathbf{z}|\theta) \\ &\approx \log \prod_{y' \in \mathbf{y}} \sum_{b_l(y')} P(y', b_o(y'), b_l(y'), \theta) \end{aligned} \quad (1)$$

Similarly, the original variational lower-bound of Eq. (1)

$$L(\theta) \geq \sum_{\mathbf{z}} q(\mathbf{z}) \log P(\mathbf{y}, \mathbf{z}|\theta) - \sum_{\mathbf{z}} q(\mathbf{z}) \log q(\mathbf{z}) \quad (2)$$

(where q is the unknown marginal distribution of latent variables) becomes

$$\begin{aligned} \tilde{L}(\theta) \geq & \sum_{y' \in \mathbf{y}} \sum_{b_l(y')} q(b_l(y')) \log P(y', b_o(y'), b_l(y')|\theta) \\ & - \sum_{y' \in \mathbf{y}} \sum_{b_l(y')} q(b_l(y')) \log q(b_l(y')) \end{aligned} \quad (3)$$

As shown here, the iEM provides us a variational approximation by rendering the intractable entropy term in Eq. (2) tractable (3).

The E step of iEM computes the *approximate* marginal posterior of each latent variable $Z' \in \mathbf{Z}$ according to

$$P(z'|\mathbf{y}, \theta^{(t)}) \approx \sum_{b_l(z')} P\left(z', b_l(z')|b_o(z'), \theta^{(t-1)}\right) \quad (4)$$

where superscript t is the iteration index. The new posteriors yields a new bound (Eq. (3)), and in the M step new LRMs parameters are chosen to maximise this bound. For any relation $r : D_1 \times \ldots \times D_k \to \{v_1, \ldots, v_{|r|}\}$ (latent or observed), the new parameter $\theta_r^{(t+1)}[u, \mathbf{g}]$ is the conditional probability that $r = u$ given $\mathbf{H} = \mathbf{g}$, where $\mathbf{H} = \{h_{D_1}, \ldots, h_{D_k}\}$, and $\mathbf{g} = \{g_1, \ldots, g_k\}$ is a vector instantiation of $\mathbf{H}$. Let $\mathbf{y}_r$ be the observations of r, then $\theta_r^{(t+1)}[u, \mathbf{g}]$ is updated according to

$$\begin{aligned} \theta_r^{(t+1)}[u, \mathbf{g}] &= \frac{N(r = u \wedge \mathbf{H} = \mathbf{g})}{N(\mathbf{H} = \mathbf{g})} \\ &= \frac{\sum_{y_j \in \mathbf{y_r}} P\left(y_j = u|\mathbf{y_r}, \theta^{(t)}\right) w_j^{(t)}}{\sum_{y_j \in \mathbf{y_r}} w_j^{(t)}} \end{aligned} \quad (5)$$

$N(\cdot)$ denotes expected counts. As y_j represents the value of some ground relation $r(a_1, \ldots, a_k)$, $w_j^{(t)}$ is the weight of case y_j given by

$$w_j^{(t)} = \prod_{i=1}^{k} P(h_{D_i}(a_i) = g_i|b_o(h_{D_i}(a_i)), \theta^{(t-1)}) \quad (6)$$

where the terms in the product are derived from the E step. For latent relations, all $w_j^{(t)} = 1$, and $\mathbf{y}_r$ is replaced with a vector of posterior probabilities (from the E step).

Convergence of iEM is assured as shown by Neal and Hinton (1998). The complexity each inference made in the E step depends on the size of the Markov blanket and method of inference. Assuming an exact inference algorithm, for each of the q individuals a posterior calculation is done that is proportional to the t observed ground relations the individual participates in, and the size of the conditional probability table for each relation (call this d). The E step complexity is then $\mathcal{O}(q \cdot t \cdot d)$. Note that d is exponential in the number of parents, but is independent of the number of individuals and observations.

3. Experiments

Here we validate the performance of iEM procedure for finding accurate LRM parameters. For benchmarking purposes, we also use the Infinite Relational Model (IRM) of Kemp et al. (2006) in our experiments, trained using MCMC.

Social Networks – We use a collection small social networks from UCINet[2]. Each social network models one relationship type between individuals of a single object domain. We perform leave-one-out validated prediction performance (up to a maximum of 200 cases) and report the log-loss. In these trials, the iEM is run with 10 random restarts with the best outcome returned, with cluster variable size ($g_{(\cdot)}$) nominally set to 2, we thus call it iEM(2). Default learning parameters are used for the IRM. The following results (Table 1) were obtained with margins indicating standard error. These results indicate a general advantage towards the iEM(2), which may be attributed to the fact iEM is a soft-clustering method whilst the IRM performs hard-clustering. Particularly in the case where IRM finds few clusters, e.g. in 'taro', the cluster structure is less predictive of the relation.

Table 1. Leave-One-Out validated log-loss and standard error for iEM(2) and IRM on 15 social networks from UCINet. Lower values indicate better performance.

Data	i**EM**(2)	**IRM**
bkfratb	0.9454104 ± 0.043	1.000722 ± 0.0003
bkhamb	0.6196214 ± 0.070	1.010833 ± 0.0009
bkoffb	0.9239662 ± 0.043	1.002303 ± 0.0005
bktecb	0.8649078 ± 0.039	1.002473 ± 0.0005
kapfmu	0.8070652 ± 0.051	1.011978 ± 0.0014
kapfmm	0.6822585 ± 0.060	1.008061 ± 0.0008
kapfts1	0.6864323 ± 0.052	1.005166 ± 0.0006
kapfts2	0.8116041 ± 0.041	1.002140 ± 0.0004
szcid	1.013652 ± 0.005	1.000696 ± 0.0013
szcig	0.8957450 ± 0.039	1.002208 ± 0.0008
taro	0.6247031 ± 0.058	1.032004 ± 0.0024
thurm	0.8842990 ± 0.042	1.000356 ± 0.0014
wolfn	0.9345817 ± 0.034	1.002101 ± 0.0006
zachc	0.5264381 ± 0.065	1.004338 ± 0.0010
zache	0.5523457 ± 0.065	1.002961 ± 0.0009

Movie Ratings – In this experiment we use the MovieLens dataset[3], associated with a single relation (*rating*($User, Movie$)) defined over object domains users and movies. The input data consist of five subsets of the original data, limited to 500 users and approximately 4000 movies each. Original label values from 1–5 are thresholded to be binary values by the global mean. For each set, we obtain the average log-loss from using 5-fold cross-validation, shown in Table 2. The IRM achieves lower loss than the iEM(2), and better overall score than it achieved in the social network experiments. This may be explained by the larger object domains and greater heterogeneity of individuals, i.e. more amenable to clustering. Nominally increasing the clusters variable sizes from 2 to 4 (e.g. 4 latent classes of users and 4 for movies) for iEM, a notable gain in accuracy is achieved, suggesting that a richer cluster structure is present. Relative to the IRM, iEM achieves comparable performance, and may be suitably incorporated into a model selection strategy that allows learning the number of latent classes.

Table 2. Log-loss and standard error for iEM(2), iEM(4) and IRM on the MovieLens dataset.

IRM	i**EM**(2)	i**EM**(4)
0.858140 ± 0.0039	0.865970 ± 0.0046	0.828238 ± 0.0035
0.834761 ± 0.0046	0.836508 ± 0.0108	0.789821 ± 0.0029
0.854991 ± 0.0140	0.875009 ± 0.0105	0.824437 ± 0.0138
0.849444 ± 0.0088	0.858763 ± 0.0090	0.815944 ± 0.0071
0.855176 ± 0.0180	0.862548 ± 0.0147	0.816785 ± 0.0169

[2]http://vlado.fmf.uni-lj.si/pub/networks/data/Ucinet/UciData.htm
[3]www.grouplens.org.

4. Remarks

This paper described a method for learning parameters of latent-class relational models using a simple strategy. The method uses standard probabilistic inference under a structural assumption (e.g. the Markov blanket assumption), and the estimation procedure is realised through the incremental EM algorithm of Neal and Hinton (1998). Good empirical performance was found compared to an existing LRM modelling framework. LRMs of rich structure (e.g. for domains with multiple inter-dependent relations) is possible with the iEM, and it would be interesting to extend the current method to this problem class, bringing forth the possibility of simultaneously learning relational as well as cluster structures.

References

Dempster, A.P., Laird, N.M., and Rubin, D.B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 34: 1–38.

Kemp, C., Tenenbaum, J.B., Griffiths, T.L., Yamada, T., and Ueda, N. (2006). Learning systems of concepts with an infinite relational model. In *AAAI*.

Neal, R. and Hinton, G. (1998). A view of the em algorithm that justifies incremental, sparse, and other variants. In M.I. Jordan (Ed.), *Learning in Graphical Models*. Kluwer.

Xu, Z., Tresp, V., Yu, K., and Kriegel, H. (2006). Infinite hidden relational models. In *UAI 2006*.