Randomized Feature Generation for Markov Network Structure Learning

Jan Van Haaren and Jesse Davis
Dept. of Computer Science
KU Leuven
3001 Heverlee, Belgium
{jan.vanhaaren,jesse.davis}@cs.kuleuven.be

Markov networks are an undirected graphical model for compactly representing a joint probability distribution over a set of random variables. The goal of structure learning is to discover conditional (in)dependences in the data such that the joint distribution can be represented more compactly. Markov networks are often represented as a log-linear model, which means that structure learning can be posed as a feature induction problem.

The structure of a Markov network is typically learned in one of two ways. The first approach is to treat this task as a global search problem. Algorithms that follow this strategy use the current feature set to construct a set of candidate features. After evaluating each feature, the highest scoring feature is added to the model. The search can follow a top-down (i.e., general-to-specific) strategy (e.g., [4, 2]) or bottom-up (i.e., specific-to-general) strategy (e.g., [1, 5]). Search-based approaches tend to be slow due the large number of candidate structures. Furthermore, scoring each candidate structure requires learning the weights of each feature. Weight learning requires iterative optimization, where each iteration requires running inference over the model. Unfortunately, inference is often intractable.

The second approach, which has gained popularity in recent years, involves learning a set of local models and then combining them into a global model. Algorithms that follow this strategy consider each attribute in turn and build a model to predict this attribute's value given the remaining attributes. Each predictive model is then transformed into a set of features, each of which is included in the final, global model. Two successful approaches that use this strategy are Ravikumar et al.'s [6] algorithm, which employs L1 logistic regression as the local model and DTSL [3], which uses a probabilistic decision tree learner as the local model. Still, it can be computational expensive to learn the local models if the dataset contains a large number of variables and/or examples.

This paper pursues a third approach that views Markov network structure learning as a feature generation problem. Our algorithm, called GSSL (Generate Select Stucture Learning), has two main steps: (1) feature generation, and (2) feature selection. The first step involves quickly generating a large set of candidate features by combining aspects from randomization and specific-to-general search. GSSL constructs an initial feature set by converting each training example into a feature. By treating each training example as a feature, every initial feature has support (i.e., occurs) in the data. Consequently, the generalization process is guaranteed to produce features that match at least one training example. It then repeatedly picks a feature at random, generalizes it by dropping an arbitrary number of variables, and adds the generalized feature to the feature set. Note that the same feature can be generated multiple times. The second step selects a subset of features to include in the final model. GSSL prunes all features that were generated fewer times than a pre-defined threshold in order to improve the efficiency of weight learning. The use of threshold is based on the assumption that GSSL is likely to generate features with high support in the data more often. Then, the algorithm performs L1 weight learning on the remaining features to produce the final model. Placing a L1 penalty on the magnitude of the weight forces many of the weights to be zero, which has the effect of removing them from the model and thus selecting the most relevant features.

GSSL combines some of the benefits of both search-based and local approaches to structure learning. In the feature generation phase, the algorithm proceeds in a data-driven, bottom-up fashion to explore the space of candidate features. As a result, GSSL only constructs features that have support in the data. In the feature selection phase, the algorithm performs weight learning only once to select the best features. Here, it follows the philosophy of local model based approaches that try to minimize the computational expense of weight learning.

We performed a large scale empirical evaluation on 20 real-world datasets. The datasets have between 1,500 and 290,000 training examples and 16 and 1,500 variables. We compared GSSL to three other state-of-the-art algorithms: Ravikumar et al.'s L1 approach [6], DTSL [3] and BLM [1]. While striking in its simplicity, GSSL offers outstanding performance, in terms of both accuracy and run time. GSSL results in a more accurate learned model than Ravikumar et al.'s algorithm on 12 datasets. According to a Wilcoxon signed-rank test (with p-value of 0.025), GSSL produces significantly more accurate models than DTSL, which it beats on 15 datasets, and BLM, which it beats on 19 datasets. GSSL exhibits outstanding run time performance and it is on average twice as fast as DTSL, 15 times faster than Ravikumar et al.'s L1 approach, and 4,000 times as fast as BLM. In summary, GSSL is (significantly) more accurate than its competitors in addition to being much faster.

References

[1] J. Davis and P. Domingos. Bottom-up learning of Markov network structure. In *Proceedings of the Twenty-Seventh International Conference on Machine Learning*, Haifa, Israel, 2010. ACM Press.

[2] S. Della Pietra, V. Della Pietra, and J. Lafferty. Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:380–392, 1997.

[3] D. Lowd and J. Davis. Learning Markov network structure with decision trees. In *Proceedings of the 10th IEEE International Conference on Data Mining*, 2010.

[4] A. McCallum. Efficiently inducing features of conditional random fields. In *Proceedings of Conference on Uncertainty in Artificial Intelligence*, pages 403–410, 2003.

[5] L. Mihalkova and R. J. Mooney. Bottom-up learning of Markov logic network structure. In *Proceedings of the International Conference on Machine Learning*, pages 625–632, 2007.

[6] P. Ravikumar, M. J. Wainwright, and J. Lafferty. High-dimensional ising model selection using L1-regularized logistic regression. *Annals of Statistics*, 2009.

Presenter: Jesse Davis
Topics: Graphical models, data mining
Preference: Poster