Hyperparameter Screening and Optimisation for ILP using Designed Experiments

Ashwin Srinivasan[1,2] and Ganesh Ramakrishnan[3]

[1] IBM India Research Laboratory, 4-C, Vasant Kunj Institutional Area
New Delhi 110070, India

[2] Dept. of Computer Science and Engineering & Centre for Health Informatics,
University of New South Wales, Sydney.
ashwin.srinivasan@wolfson.oxon.org

[3] Deptment of Computer Science and Engineering,
Indian Institute of Technology, Bombay.
ganesh@cse.iitb.ac.in

Abstract. Reports of experiments conducted with an Inductive Logic Programming system rarely describe how specific values of hyperparameters of the system are arrived at when constructing models. Usually, no attempt is made to identify sensitive hyperparameters, and those that are used are often given "factory-supplied" default values, or values obtained from some non-systematic exploratory analysis. The immediate consequence of this is, of course, that it is not clear if better models could have been obtained if some form of hyperparameter selection and optimisation had been performed. Questions follow inevitably on the experiments themselves: specifically, are all algorithms being treated fairly, and is the exploratory phase sufficiently well-defined to allow the experiments to be replicated? In this paper, we investigate the use of hyperparameter selection and optimisation techniques grouped under the study of experimental design. Screening and response surface methods determine, in turn, sensitive hyperparameters and good values for these hyperparameters. Screening is done here by constructing a stepwise regression model relating the utility of an ILP system's hypothesis to its input hyperparameters, using systematic combinations of values of input hyperparameters (technically speaking, we use a two-level fractional factorial design of the input hyperparameters). The hyperparameters used by the regression model are taken to be the sensitive hyperparameters for the system for that application. We then seek an assignment of values to these sensitive hyperparameters that maximise the utility of the ILP model. This is done using the technique of constructing a local "response surface". The hyperparameters are then changed following the path of steepest ascent until a locally optimal value is reached. This combined use of hyperparameter selection and response surface-driven optimisation has a long history of application in industrial engineering, and its role in ILP is investigated using two well-known benchmarks. The results suggest that computational overheads from this preliminary phase are not substantial, and that much can be gained, both on improving system performance and on enabling controlled experimentation, by adopting well-established procedures such as the ones proposed here.

Presented at the 19th International Conference on Inductive Logic Programming (ILP'09).

Keywords: ILP, hyperparameter screening and optimisation, experimental design

1 Introduction

We are concerned in this paper with Inductive Logic Programming (ILP) primarily as a tool for constructing models. Specifications of the appropriate use of a tool, its testing, and analysis of benefits and drawbacks over others of a similar nature are matters for the engineer concerned with its routine day-to-day use. Much of the literature on the applications of ILP have, to date, been once-off demonstrations of either the model construction abilities of a specific system, or of the ability of ILP systems to represent and use complex domain-specific relationships [3, 4]. It is not surprising, therefore, that there has been little reported on practical issues that arise with the actual use of an ILP system.

Assuming some reasonable solution has been found to difficult practical problems like the appropriateness of the representation, choice of relevant "background knowledge", poor user-interfaces, and efficiency[4], we are concerned here with a substantially simpler issue. Like all model-building methods, an ILP system's performance is affected by values assigned to input hyperparameters (not to be confused with the notion of a parameter, as used by a statistician). For example, the model constructed by an ILP system may be affected by the maximal length of clauses, the minimum precision allowed for any clause in the theory, the maximum number of new variables that could appear in any clause, and so. The ILP practitioner is immediately confronted with two questions: (a) Which of these hyperparameters are relevant for the particular application at hand?; and (b) What should their values be in order to get a good model? In an industrial setting, an engineer confronted with similar questions about a complex system—a chemical plant, for example—would try to perform some form of sensitivity analysis to determine an answer to (a), and follow it with an attempt to identify optimal values for the hyperparameters identified. As it stands, experimental applications of ILP usually have not used any such systematic approach. Typically, hyperparameters are given "factory-supplied" default values, or values obtained from a limited investigation of performance across a few pre-specified values. The immediate consequence of this is that it is not clear if better models could have been obtained if some form of hyperparameter selection and optimisation had been performed. A measure of the unsatisfactory state of affairs is obtained by considering whether it would be acceptable for a chemical engineer to take a similar approach when attempting to identify optimal operating conditions to maximise the yield of his plant.

The work in [6] addressed the second question—that of optimal values for the input hyperparameters—somewhat indirectly by first constructing an "operating characteristic curve" that describes the performance of the ILP system

[4] In [7], experience gained from applications of ILP to problems in biochemistry were used to extract some guiding principles of relevance to these problems for any ILP application.

across a range of values for the relevant variables. While no specific method is proposed for identifying either the hyperparameters or their values, the characteristic curve provides a way of selecting amongst models obtained by varying hyperparameter values, provided model goodness is restricted to a specific class (that of cost functions that are linear in the error-rates). The work of Bengio [1] comes closer to our's, in that it presents a methodology to optimize several hyperparameters, based on the computation of the gradient of a model selection criterion with respect to the hyperparameters. The main restriction is that the training criterion must be a continuous and differentiable function of the hyperparameters almost everywhere. In almost all ILP settings, the training criterion cannot be even expressed in closed form, let alone being a differentiable and continuous function of the hyperparameters. That is, what can be done at best is to treat the ILP system is a black box and its variation as a function of the hyperparameters can be measured only empirically in terms of the response of the system to changes in the values of the hyperparameters. In this work, we directly approximate the evaluation function as a function of hyperparameters using response surface methodology[2].

Here we take up both selection of hyperparameters and assignment of their values directly with the only restrictions being that hyperparameter and goodness values are quantitative in nature. The methods we use have origins in optimising industrial processes [2] and been developed under the broad area concerned with the design and analysis of experiments. This area is concerned principally concerned with discovering something about a system by designing deliberate changes to the system's input variables, and analysing changes in its output response. The representation of a system is usually as shown in Fig. 1(a) (from [5]). The process being modelled transforms some input into an output that is characterised a measurable response y. The system has some controllable factors, and some uncontrollable ones and the goals of an experiment could be to answer questions like: which of the controllable factors are most influential on y; and what levels should these factors be for y to reach an optimal value. The relevance of the setting to the ILP problem we are considering here is evident from Fig. 2.

There are a wide variety of techniques developed within the area of experimental design: we will be concentrating here on some of the simplest, based around the use of regression models. Specifically, using designed variations of input variables, we will use a stepwise linear regression strategy to identify variables most relevant to the ILP system's output response. This resulting linear model, or response surface, is then used to change progressively the values of the relevant variables until a locally optimal value of the output is reached. We demonstrate this approach empirically on some ILP benchmarks.

References

1. Yoshua Bengio. Gradient-based optimization of hyperparameters. *Neural Comput.*, 12(8):1889–1900, 2000.

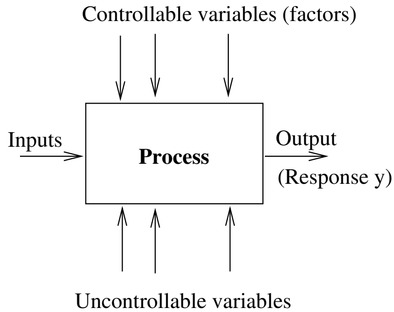

Fig. 1. Model of a system used in experimental design (from [5]). The process can be a combination of systems, each modelled by some input-output behaviour.

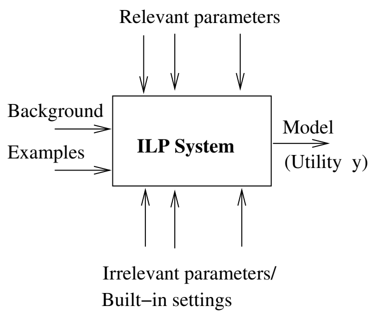

Fig. 2. An system engineer's view of an ILP system. We are assuming here that "Background" includes syntactic and semantic constraints on acceptable models. "Built-in settings" are the result of decisions made in the design of the ILP system. An example is the optimisation function used by the system.

2. G.E.P. Box and K.B. Wilson. On the Experimental Attainment of Optimum Conditions. *Journal of the Royal Statistical Society, Series B (Methodological)*, 13(1):1–45, 1951.
3. I. Bratko and S.H. Muggleton. Applications of Inductive Logic Programming. *Communications of the ACM*, 38(11):65–70, 1995.
4. S. Dzeroski. Relational Data Mining Applications: An Overview. In S. Dzeroski and N. Lavrac, editors, *Relational Data Mining*, pages 339–360. Springer, Berlin, 2001.
5. D.C. Montgomery. *Design and Analysis of Experiments (5th Ed.).* John Wiley, New York, 2005.
6. A. Srinivasan. Extracting context-sensitive models in Inductive Logic Programming. *Machine Learning*, 44:301–324, 2001.
7. A. Srinivasan. Four Suggestions and a Rule Concerning the Application of ILP. In Nada Lavrac and Saso Dzeroski, editors, *Relational Data Mining*, pages 365–374. Springer-Verlag, Berlin, 2001.