Learning Probabilistic Logic Models from Probabilistic Examples

Jianzhong Chen[1], Stephen Muggleton[1], and José Santos[2]

[1] Department of Computing, Imperial College London, London SW7 2AZ, UK
{cjz, shm}@doc.ic.ac.uk
[2] Division of Molecular BioSciences, Imperial College London, London SW7 2AZ, UK
jose.santos06@imperial.ac.uk

Abstract. We revisit an application developed originally using Inductive Logic Programming (ILP) by replacing the underlying Logic Program (LP) description with Stochastic Logic Programs (SLPs), one of the underlying Probabilistic ILP (PILP) frameworks. In both the ILP and PILP cases a mixture of abduction and induction are used. The abductive ILP approach used a variant of ILP for modelling inhibition in metabolic networks. The example data was derived from studies of the effects of toxins on rats using Nuclear Magnetic Resonance (NMR) time-trace analysis of their biofluids together with background knowledge representing a subset of the Kyoto Encyclopedia of Genes and Genomes (KEGG). The ILP approach learned logic models from non-probabilistic examples. The PILP approach applied in this paper is based on a general approach to introducing probability labels within a standard scientific experimental setting involving control and treatment data. Our results demonstrate that the PILP approach not only leads to a significant decrease in error accompanied by improved insight from the learned result but also provides a way of learning probabilistic logic models from probabilistic examples.

1 Introduction

One of the key open questions of Artificial Intelligence concerns *Probabilistic Logic Learning* (PLL) [4], i.e. the integration of probabilistic reasoning with first order logic representations and machine learning. This integration is needed in order to face the challenge of real-world learning and data mining problems in which data are complex and heterogeneous and we are interested in finding useful predictive and/or descriptive patterns. The term *probabilistic* refers to the use of probabilistic representations and reasoning mechanisms grounded in probability theory, such as Bayesian networks and stochastic grammars. Such representations have been successfully used across a wide range of applications and have resulted in a number of robust models for reasoning about uncertainty. The term *logic* refers to representations based on first order logic such as those studied within the field of computational logic. The primary advantage of using such representations is that it allows one to elegantly represent complex situations involving a variety of objects as well as relations among the objects. The

term *learning* in the context refers to deriving the different aspects of a model in a probabilistic logic on the basis of data. Typically, one distinguishes various learning algorithms on the basis of the given data (fully or partially observable) or on the aspect being learned (parameter estimation or logical structure learning). The motivation for learning is that it is often easier to obtain data for a given application domain and learn the model than to build the model using traditional knowledge engineering techniques.

Inductive Logic Programming (ILP) [12] studied *logic learning*, i.e. learning and data mining within first order logical or relational representations. PLL is also called Probabilistic ILP (PILP) [15] as it naturally extends ILP to probabilistic case that can explicitly deal with uncertainty such as missing and noisy information. There have been some promising PILP frameworks and systems developed so far to help people build probabilistic logic models and make use of some known PLL utilities, such as Bayesian Logic Programs (BLPs) [9], Stochastic Logic Programs (SLPs) [10] and Markov Logic Networks (MLNs) [16]. Although more and more new developments and successful applications have been published, there are still many challenges in the PLL research. One of such challenging questions is '*should PLL/PILP always learn from categorical examples?*' In other word, the data sets used by most PLL/PILP systems or applications are non-probabilistic, like those used in ILP systems. On the one hand, there are more or less information loss for learning using just categorical examples compared with the raw (possibly continuous) data. On the other hand, the ability of handling examples together with empirical probabilities should be one of the distinct positive features of PILP against ILP. A major reason for the problem is we lack the corresponding methods to extract or estimate empirical probabilities from raw data. We attempt to show a solution to the problem in this paper.

This paper is arranged as follows. Section 2 provides background relating to an introduction of the empirical probability and the biological application area of metabolic network inhibition as well as the previous study of abductive ILP. This is followed by a description of the abductive approach to SLPs used in this paper. A general approach is described in Section 4 for extracting probability labels from scientific data. This approach is employed in the experiments of Section 5 which apply abductive SLP learning to the metabolic network inhibition problem. We show that significant accuracy increases are achieved using the new approach. Section 6 concludes with a comparison to some related approaches and a discussion of the results.

2 Motivation and Background

2.1 Probabilistic ILP with Probabilistic Examples

To address our motivation, we claim the following learning setting of PILP with probabilistic examples, where each observed example is associated with a probability specifying its degree of being sampled from some distribution.

Definition 1 (Learning Setting of Probabilistic ILP with Probabilistic Examples). *Given a set* $E = \{P(e) : e\}$ *of probabilistic examples, a*

background theory B, PILP finds, in a probabilistic logical hypothesis space $\mathcal{H}$, $H^* = \arg\max_{H \in \mathcal{H}} \prod_{e \in E} P(e|H, B)$ *such that* $\forall e \in E : P(e|H^*, B) \geq 0$*, where* $P(e|H, B) = \frac{P(H,B|e)P(e)}{P(H,B)}$.

Here we distinguish that *probabilistic examples* have empirical probabilities in PILP from that *categorical examples* are positive or negative in ILP. For the probability, we refer to the so called '*empirical probability*' or more precisely, the estimated probability distribution[3], which we could statistically extract from the raw (numerical) data. Empirical probability is not prior probability, which is always used in Bayesian inference [6] and is often the purely subjective assessment made by an experienced expert. Empirical probability could be thought as posterior probability conditional on the experimental data. There are many cases in reality and science involving control and treatment data or observations. For example, the control data of blood could be gotten from a group of normal people and the treatment data of blood are collected from a group of patients with some disease.

2.2 Learning Metabolic Network Inhibition

To demonstrate our method, we revisit an application developed originally using ILP by replacing the underlying logic program description with abductive SLPs [18]. Our study then aims at learning probabilistic logic models of metabolic network inhibition from probabilistic examples. In this section, we summarise the application area as well as the original ILP study [18].

Metabolism provides a source of energy for cells and degrades toxic compounds in preparation for excretion. The graph of these interlinked chemical reactions is known as the *metabolic network* [1]. The reactions that take place in the network are catalysed by highly specialised proteins known as *enzymes*. One of the less understood phenomena in the metabolic network is *inhibition*. Some chemical compounds, known as inhibitors, can affect enzymes, impeding their function. This in turn affects the normal flux in the metabolic network, the result of which is reflected in the accumulation or depletion of certain metabolites. Inhibition is important because many substances designed to be used as drugs can have an inhibitory effect on other enzymes. Any system able to predict such inhibitory effect on the metabolic network would be useful in assessing the potential side-effects of drugs.

Several machine learning techniques have been conducted to use experimental data on the accumulation and depletion of metabolites to model the inhibitory effect of various toxins, such as hydrazine and ANIT, in the metabolic network of rats (Fig. 1) [18]. A group of rats are injected with hydrazine and the changes

[3] In mathematics, empirical probability is also called experimental probability, which is the probability of an event, defining as the ratio of favourable outcomes to the total number of trials (from Wikipedia). In a more general sense, empirical probability estimates probability distribution of a population from some samples and observations.

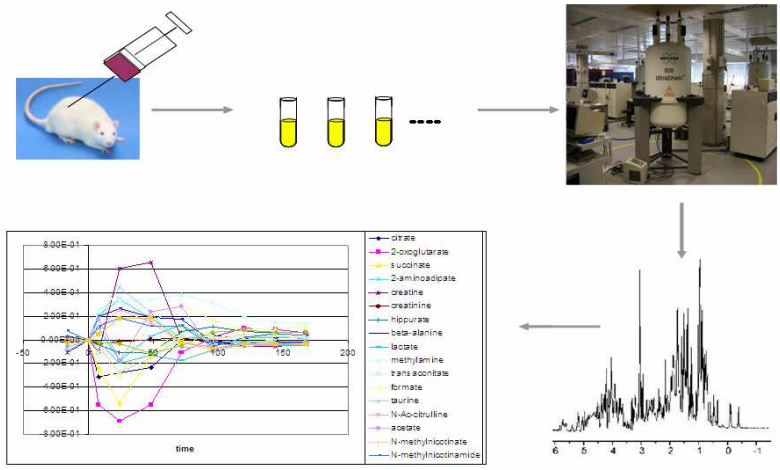

Fig. 1. Description of the scientific experiments for machine learning metabolic network inhibition. The example data was derived from studies of the effects of toxins on rats using NMR time-trace analysis of their biofluids.

on the concentrations of a number of chemical compounds are monitored during a period of time. Relative concentrations of chemical compounds are extracted from complex Nuclear Magnetic Resonance (NMR) spectra of urine.

One of the applied machine learning approaches is abductive ILP [18], a variant of ILP. In that work, the binary information on up/down regulations of metabolite concentrations following toxin treatment is combined with background knowledge representing a subset of the Kyoto Encyclopedia of Genes and Genomes (KEGG) metabolic diagrams. An abductive ILP program is used to learn the potential inhibition occured in the network, which contains a set of *observables* (predicate `concentration/3`), a set of *abducibles* (predicate `inhibited/4`), a set of *background facts* (predicates `reactionnode/3`, `enzyme/1` and `metabolite/1`) and a set of general *background rules* under which the effect of the toxin can increase or reduce the concentration of the meabolites. An example of metabolic network and the learned inhibition are demonstrated in Fig. 2.

The key point in the abductive ILP study is it supports an integration of *abduction* and *induction* [5] in an ILP setting, which also motivates our study of abductive SLPs in this paper. Abduction is first used to transform the observations to an extensional hypothesis on the abducibles. The induction takes this as input and tries to generalize the extensional information to general rules for the abducible predicates now treating them as observables for its own purposes. The cycle can then be repeated by adding the learned information on the abducibles back in the model as new partial information on the incomplete abducible predicates. This will affect the abductive explanations of new observations to be used again in a subsequent phase of induction. Hence through the integration, the ab-

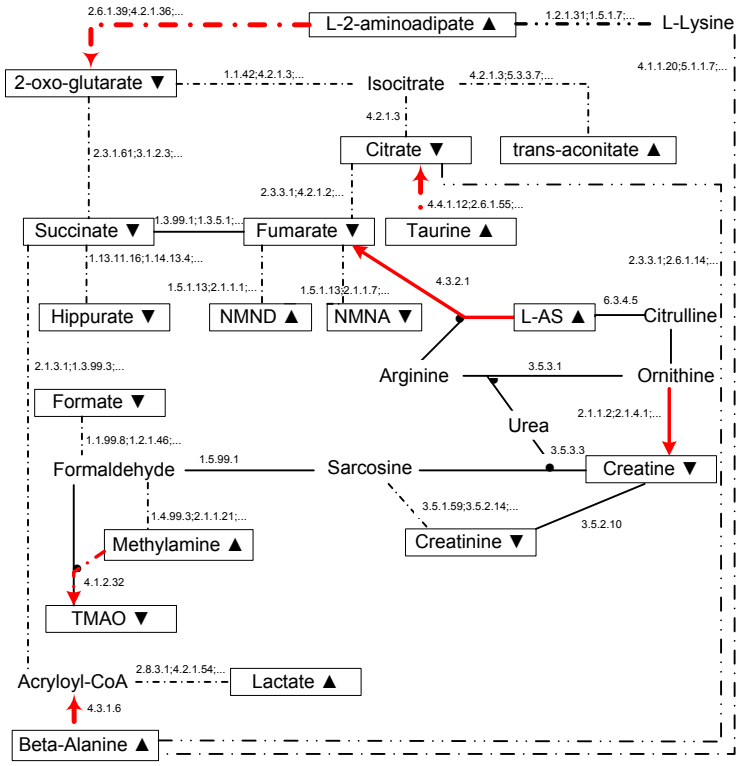

Fig. 2. An example of rat metabolic network and the corresponding inhibition of hydrazine (at hour 8) learned by abductive ILP. Information on up/down changes in metabolite concentrations from NMR spectra is combined with KEGG metabolic diagrams. The enzymes associated with a single reaction (solid line) or a linear pathway (dotted line) are shown as a single enzyme or a sequence of enzymes. Colored arrows show the found inhibition with directions.

ductive explanations of the observations are added to the theory in a generalized form given by a process of induction on them.

3 Abductive SLPs

3.1 Abduction with SLPs based on a Possible Worlds Semantics

Stochastic Logic Programs (SLPs) [10] are an established PLL/PILP formalism for probabilistic reasoning and learning. Despite their use in stochastic contexts, SLPs have not previously been provided with a possible worlds semantics and their interpretation has generally been allied in the literature [14] to Halperns's *domain frequency* based probabilistic models [7]. Abductive SLPs are a learning framework that supports abductive modeling and learning [8] in SLPs to provide

a probability distribution over the abductive hypotheses based on their *possible worlds* [2].

In fact, SLPs have a distributional semantics [17], the one that assigns a probability distribution to the atoms in the Herbrand base of the clauses in an SLP program according to a stochastic SLD resolution strategy [11]. The stochastic proof procedure gives SLPs a domain frequency semantics which define probability distributions over ground atoms through building stochastic proof trees. However, being motivated by Markov Models we can interpret the probabilities assigned to the clauses as conditional probabilities between possible worlds. In the context of abductive SLPs [2], given a clause $p : H \leftarrow B$, the probability p is interpreted as $P(B \mid H)$, ie. the conditional probability of the body B being true (in some possible worlds) given that the head H is true (in the same possible worlds). Such semantics obviously correspond to an *explanatory* semantics of conditional probability that explain the possible causes for a given result, in contrast with the normal *causal* semantics ($P(H \mid B)$) that infer the result given the causes, like the semantics defined in Bayesian Networks and BLPs [9]. Under the explanatory semantics, the possible explanations that are computed for an atom are clearly based on possible worlds, ie. each possible world is an explanation and the posterior probability of the atom is the sum of the probabilities over all the possible worlds (explanations).

When addressing this in logical reasoning and learning, the above semantics also suggest the possibility of introducing abduction into SLPs which can find explanations for observations. In fact, we distinguish two forms of reasoning/learning - *induction* and *abduction* [5]. Induction generates *intensional* knowledge in the form of new general rules that can provide directly or indirectly new relationships between the elements of our model. The inductive hypothesis introduces new links between the relations we are studying and allows new predictions on the observations. On the other hand, abduction typically generates *extensional* knowledge that refers only to some pre-set abducibles and that is specific to the particular state of the world pertaining to the observations and model. Adding an explanation to the model then allows us to predict further observable information but the predictive power of abduction is restricted to come from the already known rules in the model.

3.2 The Learning Framework of Abductive SLPs

Definition 2 (Stochastic Abduction in Abductive SLPs). *An abductive SLP S_A supports stochastic abduction in the underlying SLP learning. Suppose e is an observed arbitrary first order ground atom in S_A, $\delta(e, S_A)$ is a stochastic ground derivation of e derived from S_A involving a set of ground abducibles (abduced facts) A_e. We say that a model M_e is a least Herbrand model of (S_A, e, A_e) if it contains all and only the ground facts in δ and we have*

$$P(M_e \mid e) = \frac{P(e, M_e)}{P(e)} = P(\delta(e, S_A)) = \prod_{C \mid C \in \delta(e, S_A)} P(C),$$

where C is a stochastic clause with probability $P(C)$ appeared in δ. Now suppose an arbitrary abducible $a \in A_e$, then the probability of a can be defined to be the sum of the probabilities of all the least models that have a in their abduced facts

$$P(e,a) = \sum_{M_e | a \in M_e} P(e, M_e) = \sum_{\delta(e,S_A) | a \in \delta(e,S_A)} P(e)P(\delta(e, S_A)).$$

Based on the underlying process of stochastic abduction, abductive SLPs further provide a learning mechanism which supports an integration of both abduction and induction.

Definition 3 (Learning Setting of Abductive SLPs). *Assume a background knowledge theory B in the form of a complete/pure SLP and a set of independently observed ground probabilistic examples E (ie. each $e \in E$ is associated with an empirical probability $P(e)$), abductive SLPs aim to learn an SLP S_A which particularly contains a set of labelled hypothesised abducibles $H = \{p : a\}$, ie. a complete stochastic definition of some abduced predicate, such that when added to S_A, we have $B \wedge H \models E$ and the labels $\{p\}$ are chosen to maximize the likelihood of H given E and B*

$$L(H \mid E, B) = P(E \mid H, B) = \prod_{e \in E} P(e, a) = \prod_{e \in E} \sum_{\delta \in SS(e,H,B)} P(e)P(\delta(e, H, B)),$$

where $SS(e, H, B)$ denotes the set of all stochastic SLD derivations of e from the model (H, B).

Abductive SLPs support a combination of induction and abduction in the above learning setting, which is a cycle where abduction finds probabilistic explanations for probabilistic observations and induction generalises probabilistic rules from probabilistic explanations. In practice, as SLP structure learning is still a challenging problem in the area, we could apply SLP parameter estimation algorithms, such as FAM [3], to the abductive SLP learning. In this case, we assume the structure of H (ie. a set of abducibles $\{a\}$) are known and we perform FAM to learn the probability labels $\{p\}$ for them. Our experiments in the next two sections show how this method works in learning metabolic network inhibition.

3.3 Possible Worlds Semantics vs. Distributional Semantics

SLPs originally bring a distributional semantics or a proof-theoretic interpretation to the probability labels attached with stochastic clauses: whenever an SLD-resolution procedure has to choose between clauses, the choice is made according to probability labels. A pure SLP thus defines a distribution over instantiations of any top-level goal. Distributional semantics are similar to domain frequency semantics [7], which define a distribution over ground atoms, but NOT over the truth values of atoms. It does make sense to understand the probability label of a stochastic clause in an SLP as a conditional probability

$0.4 : s(X) \leftarrow q(X).$	% computation of the probabilities for the possible worlds
$0.6 : s(X) \leftarrow r(X).$	$P(s(a), q(a), \neg r(a)) = \frac{1}{Z} P(s(a))P(q(a))P(q(a), \neg r(a) \mid s(a)) = 0.12$
$0.3 : q(a).$	$P(s(a), \neg q(a), r(a)) = \frac{1}{Z} P(s(a))P(r(a))P(\neg q(a), r(a) \mid s(a)) = 0.12$
$0.7 : q(b).$	$P(s(b), q(b), \neg r(b)) = \frac{1}{Z} P(s(b))P(q(b))P(q(b), \neg r(b) \mid s(b)) = 0.28$
$0.2 : r(a).$	$P(s(b), \neg q(b), r(b)) = \frac{1}{Z} P(s(b))P(r(b))P(\neg q(b), r(b) \mid s(b)) = 0.48$
$0.7 : r(b).$	% all other possible worlds have probability 0 under CWA

Table 1. An example of SLP and the computation of the probabilities for the possible worlds in the abductive SLP setting; assume the empirical probabilities for the two observations are equal, ie. $\{P(s(a)) = 0.5, P(s(b)) = 0.5\}$, and the abducibles are set to $\{q(a), q(b), r(a), r(b)\}$; Z is the corresponding normalization item.

of its head given its body, but that should not be a standard interpretation in distributional semantical setting. In contrast, possible worlds semantics provide model-theoretic interpretation to the probabilities: some models or atoms are said to be true only in some possible worlds, which are determined by multiple (exclusive) joint instantiations of some facts. Conditional probabilities could be semantically defined by truth values under possible worlds semantics.

Abductive SLPs are a learning framework that provides possible worlds semantics for SLPs with the help of abduction. Under possible worlds semantics, not only the probability label of a clause can be interpreted by a conditional probability of its body given its head, but also distributions could be defined and discussed over the truth values of atoms. Previous section already shows how stochastic abduction works and how the distributions are computed in the underlying SLD-resolution proof procedure. Another advantage of the possible worlds semantics lies in that there is implicitly a *closed world assumption* (CWA) set in the stochastic abduction procedure in which the atoms that are not in the derivations are considered false in the world of the derivations. This assumption efficiently solves the computation of the probabilities for the *ambiguous* atoms/abducibles, which have more than one overlapping derivations that can yield them, in SLPs [3].

To illustrate the difference, an example SLP S_0 is shown in Fig.1. S_0 defines a distribution $\{0.24, 0.76\}$ over a sample space $\{s(a), s(b)\}$ under the traditional distributional semantics; whereas in the abductive SLP setting under possible worlds semantics, it defines a distribution $\{0.12, 0.12, 0.28, 0.48\}$ over a set of 64 possible worlds, in which 4 have non-zero probabilities (as shown in Fig.1) and all other possible worlds have probability 0 under CWA. We could consider $q(X)$ and/or $r(X)$ to be abducible predicates in the case. In particular, the clauses $0.4 : s(a) \leftarrow q(a)$ and $0.6 : s(a) \leftarrow r(a)$ are interpreted as $P(q(a), \neg r(a) \mid s(a)) = 0.4$ and $P(\neg q(a), r(a) \mid s(a)) = 0.6$ respectively in abductive SLPs under CWA that further implies $P(q(a), r(a) \mid s(a)) = 0$, but the interpretation could be ambiguous in the distributional semantics, ie. it lacks of explanation to the probability of the overlapping $P(q(a), r(a))$.

1. Initialize a matrix MR with column=2 and row=number of metabolites;
2. for each metabolite α do:
   2.1. C_α=$\{concentration(\alpha)\}$, a set of α values observed in the **control** cases;
   2.2. M_α=MEAN(C_α),SD_α=STANDARDDEVIATION(C_α);
   2.3. T_α=$\{concentration'(\alpha)\}$, a set of τ_α values observed in the **treatment** cases;
   2.4. $MR[\alpha, 1] = M_\alpha <$ MEAN(T_α) ? Up : Down;
   2.5. $MR[\alpha, 2]$= MEAN($\{$PNORM($\tau_\alpha, M_\alpha, SD_\alpha$)$\}$);
3. Apply matrix MR in the abductive SLP learning

Table 2. Algorithm of estimating empirical probabilities from control/treatment data for metabolic network inhibition

4 Extracting Probabilistic Examples from Scientific Data

Assume we have a scientific data set involving a set of data values collected from some control cases as well as a set of data points from some treated cases. All the data are mutually independent. The method consists in constructing, for each attribute α in the control case, a normal distribution $\mathcal{N}_\alpha$ with parameters μ and σ calculated from a set of α values in all the control cases. Then, for each value τ_α that corresponds to α and is observed in the treatment cases, the integral from -∞ to τ_α is calculated in $\mathcal{N}_\alpha$ (eg. using the function PNORM in the R Language[4]). Finally, the average of the integrals (each in $[0, 1]$), ρ_α is taken. We claim that ρ_α indicates to what extent τ_α in the treated case differs from α in the control case. It follows that a value of $\rho_\alpha < 0.5$ specifies α is less expressed in the treated case compared to that in the control case, $\rho_\alpha > 0.5$ indicates α is more expressed, and $\rho_\alpha = 0.5$ shows that α has no difference from the control case. Furthermore, we could say that $\mathcal{C}_\alpha = \rho_\alpha$ if $\rho_\alpha > 0.5$ or $\mathcal{C}_\alpha = 1 - \rho_\alpha$ otherwise, where $\mathcal{C}_\alpha$ represents the confidence of the assertion 'α is more or less expressed in the treated cases relative to the control cases'. From our point of view, $\mathcal{C}_\alpha$ is the estimated empirical probability of α happened in the treatment cases against the control cases. Table 2 presents a pseudo code for the above explained algorithm applied to our rat metabolic network inhibition data set.

In our sample data file, after some pre-processing, we had the raw data values of 20 rows (one per rat) and 20 columns (one per metabolite). The first 10 rows represent control rats (injected with a placebo) and the latter 10 represent treated rats which were injected with 30mg dose of hydrazine (and ANIT respectively). Each column has information on the concentration of a given metabolite at the 8th hour after the injection. The above method has been applied to the raw data set by developing a small R script. We are aware that using only 10 data points to build a normal distribution for control case is not ideal but believe it is the best possible approximation with the data at hand[5]. The result matrix

[4] PNORM(x, m, sd) calculates the area to the left of its first argument in a normal distribution defined by the other two arguments.

[5] Please note that experiments in some scientific areas, such as metabolic network inhibition, are very expensive.

Metabolite	Concentration	Empirical Probability	ILP Prediction	SLP_C Prediction	SLP_P Prediction
citrate	down	0.9843	down	0.6900	0.6860
2-og	down	1.0000	up	0.5680	0.6900
succinate	down	0.9368	up	0.2590	0.2970
l-2-aa	up	0.9962	up	0.6580	0.8280
creatine	down	0.5052	up	0.3070	0.4430
creatinine	down	0.5798	up	0.3220	0.4930
hippurate	down	0.7136	up	0.3030	0.1660
beta-alanine	up	0.9659	up	0.5670	0.6860
lactate	up	0.9503	up	0.5400	0.5160
methylamine	up	1.0000	down	0.3010	0.5250
trans-aconitate	down	0.6488	up	0.3920	0.4410
formate	down	0.9368	up	0.3920	0.4230
taurine	up	0.7362	up	0.6500	0.8100
acetate	up	0.6727	up	0.5560	0.5390
nmna	up	0.5239	up	0.4890	0.4920
nmnd	up	0.6414	up	0.4890	0.4990
tmao	up	0.5166	up	0.3100	0.1120
fumarate	up	0.6970	down	0.2970	0.5020
l-as	up	0.6748	up	0.5040	0.5070
glucose	up	0.8096	up	0.5570	0.5310
Ratio of Correct Prediction			11/20	9/20	11/20
Predictive Accuracy			55%	68.31%	72.74%

Table 3. Experiment Results for hydrazine inhibition. The predictive accuracy of a metabolite is defined to be (1 - the error of SLP prediction from the corresponding empirical probability); and the model predictive accuracy is defined to be the average predictive accuracy over all the metabolites.

with the estimated concentration level and empirical probabilities for hydrazine are presented in column 2 and 3 of Table 3.

5 Experiments - Learning Metabolic Network Inhibition

The experiments include two learning tasks – learning abductive SLP_C from categorical examples (as done in the ILP learning) and learning abductive SLP_P from probabilistic examples. In particular, each observation inputted into SLP_P is associated with an estimated empirical probability ρ we have obtained in last section. In addition, our learning framework also allow us to provide the complementary observations with probability $(1-\rho)$ (like the negative examples in ILP). The current FAM algorithm implementation (Pe-pl software) indirectly supports the introduction of probabilities in the observation list by allowing the same observation to appear an arbitrary (integer) number of times. For instance, while in SLP_C a partial input would be simply *concentration(citrate, down)-1*, in SLP_P the input would be *[concentration(citrate, down)-98, concentration(citrate, up)-*

2], which implicitly corresponds to 'the concentration of citrate is down with probability 98% and is up with probability 2%'. So, probabilistic examples are applied in the abductive SLP framework rather than positive and negative categorical examples in the ILP learning.

Null hypotheses: The predictive accuracy of an SLP_P model **does not** outperform an SLP_C model for predicting the concentration level of metabolites in a given rat metabolic network inhibition (of a given toxin) experiment.

Materials and Input: The (estimated) empirical probabilities are extracted from the raw data consisting of the concentration level of 20 metabolites on 20 rats (10 control cases and 10 treated cases) after 8 hours of the injection of hydrazine and ANIT. The initial SLP uses background knowledge derived from the ILP model (see section 2.2) and adapted to our SLP models.

Methods: We apply a leave-one-out cross validation technique to do the prediction and evaluation, in which 20 SLP_C models and 20 SLP_P models are built for hydrazine and ANIT respectively. Each model is trained by 19 metabolites and tested by the left one. We perform the learning tasks by playing FAM (using Pe-pl software) under Yap 5.1.1, which has been proved particularly CPU and memory intensive.

Results: The learning framework estimates the posterior probabilities for a set of pre-set abducibles (definition of predicate inhibited) and probabilistic background knowledge from the input categorical or probabilistic examples. Fig. 3 illustrates a complete model built from all the observations. The evaluation of the prediction models is made by calculating the predictive accuracy of SLP_C and SLP_P against the estimated empirical probabilities respectively (shown in Table 3). In particular, for hydrazine, when evaluating only with the categorical observations, SLP_C and SLP_P correctly predicted 9 and 11 out of 20 metabolites respectively, while the ILP model has 11 correct predictions; however, there is a close world assumption in the ILP model where a metabolite is assumed to have a default concentration level (down or up) when the prediction is neither up nor down, which actually increases the predictive accuracy of ILP model (default accuracy is 12/20); when evaluating with the probabilistic examples, SLP_P outperforms SLP_C by 72.74% against 68.31% in average predictive accuracy (with a significance level of 0.041); for the ANIT inhibition in the same settings, the average predictive accuracy of SLP_P (90.1%) is better than that of SLP_C (88.4%), but not statistically significant (0.24) for 20 metabolites, and the main reason of that is because the default predictive accuracy (18/20) is already very high and the SLP_P predictor is not consistently better than the SLP_C predictor in each test case.

Interpretability: By comparing the learned SLP model with the previous ILP model (eg. Fig. 2 and Fig. 3 for hydrazine), apart from the inhibition patterns found in both models, at least two promising new findings have been discovered in the SLP model. The inhibition from 'beta-alanine' to 'citrate' that was not shown in ILP model has been confirmed to be crucial by the expert; and the inhibition between 'creatine' and 'creatinine' showed a contradictory result (the learned probabilities of both directions are very close), which can be explained

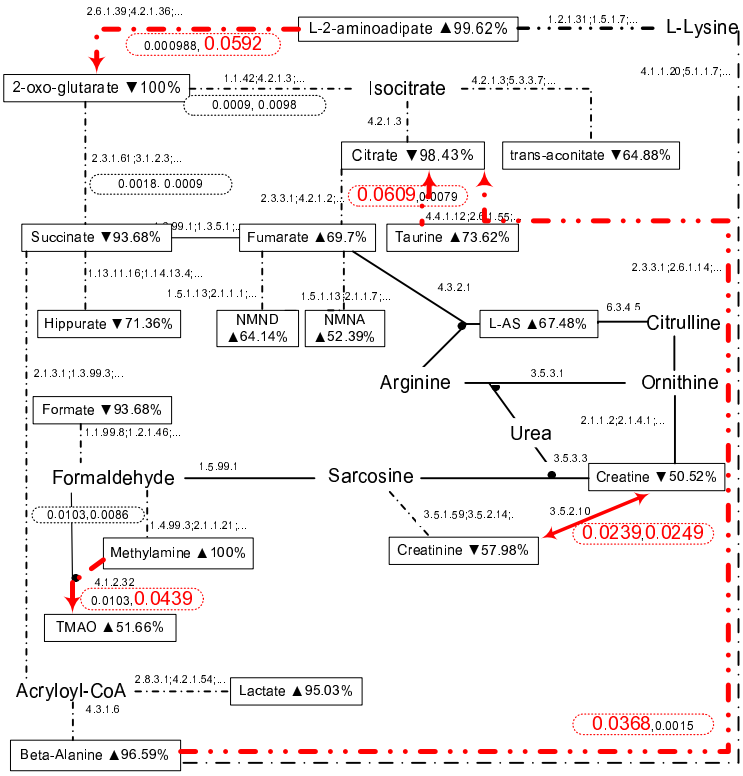

Fig. 3. Metabolic network inhibition of hydrazine learned by abductive SLPs from probabilistic examples. Each observed metabolite is associated with its concentration and the estimated empirical probability. The learned posterior probabilities for each inhibition (in two directions) are shown in the associated ellipse. For example, the left corner ellipse specifies a learned inhibition in the form of SLP clauses: '0.000988:inhibited(2.6.1.39,2-og',l-2-aa).' and '0.0592:inhibited(2.6.1.39,l-2-aa,2-og).'

by their empirical probabilities (ie. their up/down regulations are less expressed to decide the possible inhibition). In addition, the SLP models learned not only the patterns but also the probabilities or the degree of belief of the patterns which improve the insight from the learned models.

6 Discussion and Conclusions

6.1 Related Work

Given that the setting studied in the abductive SLP framework is closely related to that of the Independent Choice Logic (ICL) [13] and Programming in Statistical Modelling (PRISM) [17], it is interesting to describe the relationship between these frameworks that combine probability and logic.

From the syntax perspective, clauses or rules are treated as probabilistic (associated with probability labels) in SLPs but purely logical in both ICL and

PRISM. Logical rules are used to deterministically map a base probability distribution to an induced distribution, however, there is no mechanism of choosing between rules that have the same head. We claim that the ability of dealing with probabilistic clauses is one of the distinct features of SLPs and argue the semantical meaning of a deterministic logic rule that has probabilistic elements in the body. From the point of view of semantics, the distributional semantics are used in both the traditional SLPs and PRISM, whereas the possible worlds semantics are assumed in the frameworks of abducive SLPs and ICL (see section 3.3). The possible worlds are determined by alternatives in ICL and by the SLD-derivations in abductive SLPs.

Furthermore, the uniqueness condition set in PRISM - that exactly one atomic formulae representing observed data is derivable from any instantiation of the base distribution - requires that the set of logical rules is failure-free. As a result, the resulting distribution over observables is essentially that defined by a stochastic-context free grammar. A relaxing assumption is made in SLPs so that the resulting distributions over observables are log-linear models [3]. In terms of applying abduction, abductive SLPs provide a way to directly learn the parameters for a set of abducibles, while PRISM computes the induced distribution by searching explanation graph for observations and ICL assumes all the atomic choices as abducibles to find consistent explanations that imply the observations.

In addition, it could be possible to do the same experiments with PRISM system and compare the corresponding results, but the task is beyond the research purpose of this paper and may be done as future work.

6.2 Conclusions and Future Work

We revisit an application developed originally using ILP by replacing the underlying logic program description with SLPs. In both cases a mixture of abduction and induction are used. The ILP approach learned logic models from non-probabilistic examples. The PILP approach applied here is based on a general approach to introducing probability labels within a standard scientific experimental setting involving control and treatment data. The estimation of empirical probabilities could introduce errors compared with the unknown real distribution of control data due to the limited number of data points. However, our method shown here aims to save some probabilistic information that may have lost in pure categorical examples, so that PILP makes better prediction.

The future work, in theory, include further research of the relationship between the underlying probabilistic semantics: possible worlds, domain frequency and distributional semantics. In practice, it is interesting to set up experiments for learning the same targets using PRISM and compare the results. Efforts are needed to improve the current SLP models for ANIT inhibition with respect to both predictive accuracy and model interpretability.

In conclusion, the null hypotheses we have set in the paper and experiments were rejected (for inhibition of hydrazine) on the bases of the abductive SLP models we are using and the experimental results. Our results demonstrate that

the PILP approach not only leads to a significant decrease in error accompanied by improved insight from the learned result but also provides a way of learning probabilistic logic models from probabilistic examples.

7 Acknowledgement

The third author would like to acknowledge the funding from Wellcome Trust for his PhD program.

References

1. E. Alm and A. P. Arkin. Biological networks. *Curr. Opin. Struct. Biol.*, 13(2):193–202, 2003.
2. A. Arvanitis, S.H. Muggleton, J. Chen, and H. Watanabe. Abduction with stochastic logic programs based on a possible worlds semantics. In *Short Paper Proceedings of the 16th International Conference on Inductive Logic Programming*. University of Corunna, 2006.
3. J. Cussens. Parameter estimation in stochastic logic programs. *Machine Learning*, 44(3):245–271, 2001.
4. L. De Raedt and K. Kersting. Probabilistic Logic Learning. *ACM-SIGKDD Explorations: Special issue on Multi-Relational Data Mining*, 5(1):31–48, 2003.
5. P. Flach and A. Kakas (editors). *Abductive and Inductive Reasoning*. Pure and Applied Logic. Kluwer, 2000.
6. Andrew Gelman, John B. Carlin, Hal S. Stern, and Donald B. Rubin. *Bayesian Data Analysis*. CRC Press, 2 edition, 2003.
7. J. Y. Halpern. An analysis of first-order logics of probability. *Artificial Intelligence*, 46:311–350, 1989.
8. A.C. Kakas, R.A. Kowalski, and F. Toni. Abductive logic programming. *Journal of Logic and Computation*, 2, 1992.
9. K. Kersting and L. De Raedt. Bayesian logic programs. In *Proceedings of the Work-in-progress Track at the 10th International Conference on Inductive Logic Programming*, pages 138–155, 2000.
10. S.H. Muggleton. Stochastic logic programs. In L. de Raedt, editor, *Advances in Inductive Logic Programming*, pages 254–264. IOS Press, 1996.
11. S.H. Muggleton. Learning structure and parameters of stochastic logic programs. *Electronic Transactions in Artificial Intelligence*, 6, 2002.
12. S.H. Muggleton and L. De Raedt. Inductive logic programming: Theory and methods. *Journal of Logic Programming*, 19,20:629–679, 1994.
13. D. Poole. The independent choice logic for modelling multiple agents under uncertainty. *Artificial Intelligence*, 94(1), 1997.
14. A. Puech and S.H. Muggleton. A comparison of stochastic logic programs and Bayesian logic programs. In *IJCAI03 Workshop on Learning Statistical Models from Relational Data*. IJCAI, 2003.
15. L. De Raedt and K. Kersting. Probabilistic inductive logic programming. In S. Ben-David, J. Case, and A. Maruoka, editors, *Proceedings of the 15th International Conference on Algorithmic Learning Theory*, volume 3244 of *Lecture Notes in Computer Science*. Springer-Verlag, 2004.

16. Matthew Richardson and Pedro Domingos. Markov logic networks. *Mach. Learn.*, 62(1-2):107–136, 2006.
17. T. Sato. A Statistical Learning Method for Logic Programs with Distribution Semantics. In *Proceedings of the 12th International Conference on Logic Programming (ICLP-1995)*, pages pp. 715 – 729, 1995.
18. A. Tamaddoni-Nezhad, R. Chaleil, A. Kakas, and S.H. Muggleton. Application of abductive ILP to learning metabolic network inhibition from temporal data. *Machine Learning*, 64:209–230, 2006. DOI: 10.1007/s10994-006-8988-x.