An Implementation of Statistical Default Logic

Gregory R. Wheeler and Carlos Damásio
Centro de Inteligência Artificial (CENTRIA)
Departamento de Informática, Universidade Nova de Lisboa
2829-516 Caparica, Portugal
{greg,cd}@di.fct.unl.pt

December 31, 2003

Abstract

Statistical default logic is a variation of classical (i.e., Reiter's) default logic designed to represent basic inference forms common in inferential statistics. In this paper we present an embedding of an important subset of statistical default theories into stable model semantics.

1 Introduction

Standard statistical inference is non-monotonic. Parameters of a target population may be estimated by measures on a sample that, after tested for bias, serve as a defeasible estimate of the population's corresponding parameters. For example, we may estimate the age of a population by identifying the mean age of a 'representative' sample drawn from the population. However, classifying a sample as representative is not a straightforward affair: for *knowing* that a sample is representative is to be in the position of not needing to do inferential statistics.

The fit between a statistic and target parameter(s) is defeasible in the sense that drawing a carefully selected sample is not a guarantee that it is representative of the target population. Consider again the estimation of a population's mean age. Textbooks advise that drawing a sample at random is a good procedure for selecting representative samples [e.g., Cramér 1946; Moore 1979; Larsen and Marx 2001]. But of course this is no guarantee of representativeness. Suppose a random sample selects only subscribers to *Rolling Stone*, a magazine covering popular culture with exhaustively known demographics. In so far as the population we are interested in is known to differ from the readers of *Rolling Stone* with respect to age, we then have good reason to doubt that *this* statistic gives a close estimate of the population's age.

In [Wheeler, forthcoming] a variation of default logic, called *statistical default logic*, was introduced to capture the defeasible structure of basic inference forms appearing in standard inferential statistics. Building on an insight of

Henry Kyburg and Choh Man Teng [Teng and Kyburg, 1999] that sampling assumptions invoked in standard statistical practice behave rather like default justifications than as explicit premises, statistical default logic extends Reiter's default logic by including an error-bound measure for defaults and formulas. These error bounds are combined additively during inference, giving a fixed upper-bound on frequency of error for each admissible sequence of inferences. Inference stops whenever error-bounds reach a fixed, preassigned threshold.

An idea that we wish to explore is whether we have in statistical default logic a general framework for posing uncertain queries to knowledge bases. The parameterization of statistical inference forms afforded by statistical default logic, if sound, would provide guidance in design or practice by identifying what pieces of knowledge are needed to extract conclusions within a measure of error from some knowledge base. This machinery is of immediate interest to constructing a statistical reasoning assistant but we suspect that the scope of applicability may extend to a broader class of queries in so far as they may be shown to share statistical default form.

This paper may be thought of as a first step in investigating this proposal. We here present an an intuitive introduction to statistical default logic, one based upon an example that illustrates the structural similarity between standard statistical inference and statistical default inference forms as well as with examples that illustrate how inference proceeds. We then present an embedding of the statistical default fragment into answer-set programming, one that faithfully captures the notion of terminating admissible inference sequences at a pre-assigned threshold level.

2 Representing Statistical Inference within Statistical Default Logic

We assume here familiarity with classical default logic, that introduced by Ray Reiter [Reiter 1980]. Statistical default logic [Wheeler forthcoming] extends classical default logic by associating each element in a default theory, first-order formulae and defaults, with a real number ϵ in the unit interval, $0 \leq \epsilon \leq 1$. Whereas a classical default theory $\Delta = \langle F, D \rangle$ consists of a set F of first-order formulae and a countable set D of defaults, a statistical default theory $\Delta_s = \langle W, S \rangle$ is defined as a pair consisting of a set W of formulae–ϵ pairs (called bounded sentences) and a set S of *statistical* defaults.

A statistical default is an inference form that explicitly acknowledges the *upper limit* of its frequency of error.[1] We say that a default in the form of

$$\frac{\alpha : \beta_1, ..., \beta_n}{\gamma} \epsilon, \tag{1}$$

[1]A trivial corollary of the frequency of error $\hat{\alpha}$ for a statistical inference is the upper limit of the frequency of error, denoted by ϵ. So, if $\hat{\alpha} = 0.03$ is understood to mean that the probability of committing a Type I error is 0.03, then $\epsilon = 0.03$ is understood to mean that the probability of committing a Type I error is no more than 0.03.

is an ϵ-*bounded statistical default* and the upper limit on the frequency of error-parameter ϵ an ϵ-*bound* for short, where $\frac{\alpha:\beta_1,\ldots,\beta_n}{\gamma}$ is a Reiter default and $0 \leq \epsilon \leq 1$. The schema (1) is interpreted to express that provided α and no negated β_i's, γ is false no more than ϵ over the long-run application of that rule. A Reiter default is a special case of a statistical default, namely when $\epsilon = 0$; classical default logic is a special case of statistical default logic, namely when all element ϵ-bounds equal zero.

The parameter ϵ is an essential feature of statistical defaults, just as explicit error control is essential to standard statistical inference in general, and marks an important difference between statistical defaults and Reiter defaults.

To get a sense of how representations work within this formalism consider the example from the previous section concerning the estimate of the mean age of a population, X. This problem is an instance of inference to the mean of a normal distribution when the standard deviation is known. Presented with a sample s drawn on X we calculate the mean age of s. Suppose the corrected reading is $\overline{s} = 24$ years. It is reasonable for us to infer that the mean age of X is in the interval 24 years $\pm 1.96\sigma$, where σ is the standard deviation of in years derived from the cardinality of s. Given the s-default rule schemata $(\alpha : \beta_1, ..., \beta_n /_{[1-\in]} \gamma)$, we may suppose that

> α : The calculated mean age of s is 24 years $\wedge$ Measurement errors are distributed normally with mean zero and variance σ^2.
>
> γ : The age of X is within two standard deviations of 24 years.
>
> β_1 : This is the only statistic we have for X.
>
> β_2 : There is no prior statistical knowledge of the distribution of age in the class that s belongs to that would lead to a conflicting inference.
>
> β_3 : There is no information concerning the condition of the sample that preëmpts the information provided by the calculation of s.
>
> $\epsilon = 0.05$.

Notice that we could collect additional statistics of the age of X and undermine the conclusion drawn from *this* rule. Surely if we have two statistics, we should use a distribution for the average of the two values (in most cases) and that uses a smaller variance.

Whether this, or one of the other justifications $\beta_1, ..., \beta_3$ is triggered does not undermine the prerequisite. It remains the case that the calculated mean age of s is 24 years and that the distribution of errors is normal, with a mean of zero and its characteristic variance. It is the consequent, the conclusion that claims that the population X is 24 $\pm 2\sigma$ years, that is blocked. Notice that it is blocked when we have additional not necessarily non-contrary information.

Justification β_2 says that if there is prior statistical information regarding the mean age of X, then that information should take precedence over any conclusion drawn from the measurement report. For instance, if we are dealing with a population with known descriptive statistics (e.g., given by a census), this knowledge should be taken account of: we typically would not infer that the estimate based upon s supersedes the census description of X, for suitably small populations not affected by data recording errors. If we already have knowledge of the age of X this knowledge should block the application of this particular default rule.

The last default, β_3, concerns general conditions that should be in place to get good estimators of age. If, for instance, the sampling procedure is carried out from a direct-mail advertisers database, we should ensure that the database is not biased with respect to age. We don't accept this as an explicit assumption: there are infinite reference classes to which s belongs. Rather, if we know that s is a member of a biased class with respect to age—such as readers of *Rolling Stone* are when compared to the population of American retirees, say—we have grounds to block the application of the default. The point isn't that knowing all members of s are *Rolling Stone* readers entails that s fails to be representative, but that knowing this calls off the bet we are making with the statistical model that we accept s as an estimate of X within two standard deviations and face being wrong no more than 5% of the time.

3 Statistical Default Extensions

Extensions for statistical default logic are constructed in the usual way, except that the operator 'terminates' when inference reaches a specified threshold and a function *Crop*() is called on the resulting set of bounded sentences, returning the set of wffs without their corresponding ϵ-bound. For details the reader is referred to [Wheeler forthcoming]. For our purposes we illustrate the mechanics of statistical default extensions with two examples.

Example 1 *Let $\Delta_s^1 = \langle W, S_1 \rangle$ be a statistical default theory, where $W = \emptyset$ and S_1 contains four s-defaults:*

$$S_1 = \left\{ \frac{:A}{A} 0.01, \frac{:B}{B} 0.01, \frac{A:B,C}{C} 0.01, \frac{A \wedge B : \neg C}{\neg C} 0.01 \right\}$$

For an error-bound parameter $\epsilon_1 = 0.02$, there is one statistical default extension Π^1 where $Crop(\Pi^1)$ contains

$$A, B, A \wedge B, C.$$

The bounded sentence A at ϵ_A is included in extension Π^1 by applying the default $\frac{:A}{A}$ and bounded sentence B at ϵ_B is included by applying the default $\frac{:B}{B}$, where each inference has an error bound of 0.01, so $(A)_{0.01}$ and $(B)_{0.01}$. $(A \wedge B)_{\epsilon_{A \wedge B}}$ is included in the extension, since the sum of the error bounds of conjoining A and B is 0.02, that is $(A \wedge B)_{0.02}$. The bounded sentence C at ϵ_C is included by using A, whose error bound is 0.01, to apply the default $\frac{A:B,C}{C}$, whose error bound is also 0.01. Hence $(C)_{0.02}$. The default $\frac{A \wedge B:\neg C}{\neg C}$ cannot be applied because *the resulting conclusion* $\neg C$ *would have an error bound of* 0.03*,* $(\neg C)_{0.03}$ *which is above the designated threshold* $\epsilon_1 = 0.02$.

For a threshold parameter $\epsilon_2 = 0.03$*, there are two statistical default extensions:* Π^1*, which is the same as described above, and* Π^2*, where* $Crop(\Pi^2)$ *contains*

$$A, B, A \wedge B, \neg C.$$

The default rule that could not be applied before is now applicable with respect to ϵ_2*, giving rise to the second extension* Π^2*.*[2]

Example 2 *Now let* $\Delta_s^2 = \langle W, S_2 \rangle$ *be a statistical default theory, where* $W = \emptyset$ *and* S_2 *contains six s-defaults:*

$$S_2 = \left\{ \frac{: \neg B, C}{C} 0.00, \frac{: C}{C} 0.02, \frac{C : B}{B} 0.01, \frac{: \neg B}{\neg B} 0.03, \frac{: \neg B, A}{A} 0.01, \frac{: \neg A}{\neg A} 0.01 \right\}$$

For an error-bound parameter $\epsilon_1 = 0.02$*, there is no statistical default extension, since while both* $\frac{:\neg B, C}{C}0.00$*,* $\frac{:C}{C}0.02$ *yield* C *only the bounded sentence* $\langle C, 0.00 \rangle$ *from* $\frac{:\neg B, C}{C}0.00$ *may be substituted for the antecedent of* $\frac{C:B}{B}0.01$ *which in turn is applicable in extensions consistent with* B*. But* $\frac{:\neg B, C}{C}0.00$ *is applicable only in extensions consistent with* $\neg B$*.*

For an error-bound parameter $\epsilon_2 = 0.03$*, there are three extensions. We will continue the convention of example 1 of distinguishing them by focusing on the literals of each extension; this will also serve our purposes in the remainder of the paper. However, because this example highlights the role that error-bounds play in constructing extensions we will display the extensions first in uncropped form, then in cropped form.*

$\Pi_1 \supseteq \{\langle C, 0.00 \rangle, \langle C, 0.02 \rangle, \langle \neg B, 0.01 \rangle, \langle A, 0.01 \rangle\}$
$\Pi_2 \supseteq \{\langle C, 0.00 \rangle, \langle C, 0.02 \rangle, \langle \neg B, 0.01 \rangle, \langle \neg A, 0.01 \rangle\}$
$\Pi_3 \supseteq \{\langle C, 0.02 \rangle, \langle B, 0.01 \rangle, \langle \neg A, 0.01 \rangle\}$

And the three corresponding cropped extensions are:

$Crop(\Pi_1) \supseteq \{C, B, A\}$
$Crop(\Pi_2) \supseteq \{C, \neg B, \neg A\}$
$Crop(\Pi_3) \supseteq \{C, B, \neg A\}$

We may think of each of these sets of literals as *signatures* of their corresponding statistical default extensions. In what remains we propose an implementation of statistical default logic that computes the signatures of each extension of a statistical default theory.

[2]The complete cropped extensions Π^1, when $\epsilon = 0.02$, Π^1 and Π^2, when $\epsilon = 0.03$, are as follows: $\Pi^1_{\epsilon=0.02} = \{A, B, A \wedge B, C\}$; $\Pi^1_{\epsilon=0.03} = \{A, B, A \wedge B, C, A \wedge C, B \wedge C\}$; $\Pi^2_{\epsilon=0.03} = \{A, B, A \wedge B, \neg C\}$.

4 Computing Statistical Default Extensions

In this section we describe an embedding of an important subset of statistical default theories into stable model semantics [Gelfond and Lifschitz 1988]. This embedding is designed to compute the signatures of each statistical default extension. Resorting to the available engines for computing Stable Model and Answer Set engines [Niemelä and Simons 1996; Eiter, *et. al.* 1998], we indirectly provide an efficient implementation of statistical default logic. We start by recalling the Stable Model semantics of Gelfond and Lifschitz [Gelfond and Lifschitz 1988].

A (normal) logic program is a set of rules[3] of the form:

$$h \ :- \ a_1, \dots, a_m, not\ a_{m+1}, \dots, not\ a_n$$

where h, and $a_i (0 \leq i \leq n)$ are atoms of a given first-order language. Atom H is the head of the rule, whilst $a_1, \dots, a_m, not\ a_{m+1}, \dots, not\ a_n$ is the body. We say that $not\ a_j$ is a default negated atom. A fact is a rule with an empty body and is succinctly represented by h. A rule with free variables stands for all its ground instances.

Definition 3 *Let P be a (ground) normal logic program and M a set of ground atoms in the language of P (i.e. a subset of the Herbrand base of P). The reduct P^M is the default negation free program obtained from P by:*

1. *Removing all rules of P having a default negated atom not a in the body such that $a \in M$.*

2. *Removing all occurrences of default negated atoms in the bodies of the remaining rules.*

The set M is a stable model of P iff M is the least Herbrand model of P^M.

The Answer Set Semantics [Gelfond and Lifschitz 1990] generalizes the Stable Model Semantics for the so called extended logic programs. Extended logic programs consist of rules:

$$l \ :- \ l_1, \dots, l_m, not\ l_{m+1}, \dots, not\ l_n$$

where l and l_is are literals, i.e. atoms (say, a) or the explicit negation of atoms (say, $\neg a$). The semantics is given now by special sets of ground literals, the answer sets, extending Definition 3. The reduct operation for extended logic programs is defined similarly, but the fixpoint equation must be changed to take into account that the reduct program is no longer a Horn program. Essentially, it interprets a explicit negated literal $\neg a$ as a new atom, unrelated to a, and the least model is computed as before. A special condition is then added to treat

[3]We use $:-$ instead of $\leftarrow$ in order to respect the syntax used in the existing implementations.

the case of the set of all literals. The reader is referred to [Gelfond and Lifschitz 1990; Marek and Truszczyński 1993] for details.

The relationships of stable model and answer set semantics with default logic are very well understood. See for instance [Marek and Truszczyński 1993] for a full account. In the rest of this section we extend the existing results to statistical default logic in order to compute statistical default extensions via stable model logic programming engines. A first difficulty lies in the impossibility of representation of real numbers. Furthermore, the existing implementations have support only for arithmetic over the natural/integer numbers. The following condition allows the translation of the arithmetic operations over real numbers into corresponding operations over natural numbers:

Definition 4 *Let p be a non-zero natural number. A statistical default theory $\Delta_s = \langle W, S\rangle$ is precision limited by p, if every error bound ϵ in W and S is a rational number $\epsilon = \frac{e}{p}$, for some natural number e such that $0 \le e \le p$.*

We cannot translate arbitrary statistical default theories, due to the difficulties of handling statistical inferences with disjunctive formulae with the proposed embedding. Thus, we restrict ourselves to the following types of theories:

Definition 5 *A literal statistical default theory is a statistical default theory $\Delta_s = \langle W, S\rangle$ such that:*

1. *Every bounded sentence in W is of the form $\langle l, \epsilon\rangle$, where l is a literal.*
2. *Every statistical default in S is of the form*

$$\frac{l_1 \wedge \ldots \wedge l_m : j_1, ..., j_n}{c}\epsilon$$

where $l_1, \ldots, l_m, j_1, \ldots, j_n$ and c are all literals.

Before we proceed, we require the following auxiliary notation. Given a literal $l = a(t_1, \ldots, t_m)$ or $l = \neg a(t_1, \ldots, t_m)$, by $l[e]$ it is meant, respectively, the new atom $a(t_1, \ldots, t_m, e)$ or $neg\_a(t_1, \ldots, t_m, e)$. This function adds a new argument for propagation of error-bounds, and introduces a new predicate name for negated atoms. Similarly, by $crop(l)$ we mean the new atom $crop\_a(t_1, \ldots, t_m)$ or $crop\_neg\_a(t_1, \ldots, t_m)$.

Definition 6 *Consider the literal statistical default theory $\Delta_s = \langle W, S\rangle$ precision limited by p. Construct the logic program $P_s^{\Delta}(error, p)$ as follows, where $error \le p$ is a natural number:*

1. *A bounded sentence $\langle l, \epsilon\rangle$ in W is translated into the fact:*

$$l[0].$$

2. *For every literal l in the language add the rule*

$$crop(l) \ :- \ l[E].$$

3. Every statistical default in S of the form

$$\frac{: j_1, ..., j_n}{c}\epsilon$$

is translated into the rule, where $eps = \epsilon \times p$:

$$c[eps] \ :- \ eps <= error, not\ crop(\neg j_1), \ldots, not\ crop(\neg j_n).$$

4. Every statistical default in S of the form

$$\frac{l_1 \wedge \ldots \wedge l_m : j_1, ..., j_n}{c}\epsilon$$

is translated into the rule:

$$\begin{array}{lll} c[A_m] & :- & l_1[E_1], \ldots, l_m[E_m], \\ & & A_1 = eps + E_1, \ldots, A_m = A_{m-1} + E_m, A_m <= error, \\ & & not\ crop(\neg j_1), \ldots, not\ crop(\neg j_n). \end{array}$$

where $eps = \epsilon \times p$, and $E_1, \ldots, E_m$ and $A_1, \ldots, A_m$ are new free variables.

Complete the program P_s^Δ with the following closure rules, for every combination of atoms a and b in the language:

$$\begin{array}{rll} a[E] & :- & b[E_1], \neg b[E_2], E = E1 + E2, E <= error. \\ \neg a[E] & :- & b[E_1], \neg b[E_2], E = E1 + E2, E <= error. \end{array}$$

For simplicity, we assume that the sum operation, as well as the equality and arithmetic comparison predicates are built-in. Theoretically, this can be captured by an infinite set of ground facts of the form $X = Y + Z$, such that variables are substituted by natural numbers x, y, z obeying the equation; the same applies to facts of the form $X <= Y$, where X and Y are instantiated with two natural numbers $x \leq y$.

The translation is self-explanatory. The first case takes care of the theory W; by design of statistical default logic, it is assumed that the knowledge W is considered to be error free. The rules introduced in the 2nd step implement the crop operation. The translation of statistical defaults is now immediate, where error-bounds are propagated from the bodies to the head of rules, taking into account the global threshold *error* and the error-bound of the default. The justifications are translated into default negations of the complements, as usual in the relationships of default logic with answer set semantics. The last sets of rules encode the explosive behavior of statistical default logic in face of contradiction, which differs from the one of Answer Set Semantics. The major result is the following:

Theorem 7 *Consider a literal statistical default theory $\Delta_s = \langle W, S \rangle$ with error-bound parameter ϵ, and precision limited by p, and let $error = \epsilon \times p$ be a natural number. Then, a set of ground literals $\{l_1, \ldots, l_i, \ldots\}$ is contained in $Crop(\Pi)$, where Π is a statistical default extension Π of Δ_s, iff there is a stable model of program $P_s^\Delta(error, p)$ containing $\{crop(l_1), \ldots, crop(l_i), \ldots\}$.*

By resorting to the known translation of extended logic programming under the answer set semantics into default logic, and the relationship of statistical default logic with Reiter's default logic we obtain:

Corollary 8 *Let P be a extended logic program and construct the statistical default theory $\Delta_P = \langle \emptyset, S \rangle$ by including in S a default*

$$\frac{l_1 \wedge \ldots \wedge l_m : \neg l_{m+1}, ..., \neg l_n}{l} 0.0$$

for each rule

$$l \; :- \; l_1, \ldots, l_m, not \; l_{m+1}, \ldots, not \; l_n$$

in the extended logic program. Then, M is an answer set of P iff Π is a statistical default extension of Δ_P such that $Cn(M) = Crop(\Pi)$, where Cn is the first-order consequences operator.

We conclude by illustrating the embedding:

Example 9 *Consider the theory of Example 1 with error-bound threshold of* 0.03*, and precision limited by* 100*. The translated normal logic program is:*

```
crop_a :- a(_).
crop_b :- b(_).
crop_c :- c(_).
crop_neg_a :- neg_a(_).
crop_neg_b :- neg_b(_).
crop_neg_c :- neg_c(_).

a(1) :- 1 <= 3, not crop_neg_a.
b(1) :- 1 <= 3, not crop_neg_b.

c(A1) :- a(E1), A1 = 1 + E1, A1 <= 3,
         not crop_neg_b, not crop_neg_c.

neg_c(A2) :- a(E1), b(E2),
             A1 = 1 + E1, A2 = A1 + E2, A2 <= 3, not crop_c.

a(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
neg_a(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
a(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
neg_a(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
a(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.
neg_a(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.

b(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
neg_b(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
```

```
b(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
neg_b(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
b(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.
neg_b(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.

c(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
neg_c(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
c(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
neg_c(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
c(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.
neg_c(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.
```

The stable models of the above program are:

```
{a(1), b(1), neg_c(3), crop_a, crop_b, crop_neg_c}

{a(1), b(1), c(2), crop_a, crop_b, crop_c}
```

which correspond exactly to the signature statistical default extensions of Example 1.

Example 10 *Consider the theory of Example 2 with error-bound threshold of* 0.03*, and precision limited by* 100*. The translated logic program is:*

```
crop_a :- a(_).
crop_b :- b(_).
crop_c :- c(_).
crop_neg_a :- neg_a(_).
crop_neg_b :- neg_b(_).
crop_neg_c :- neg_c(_).

a(1) :- 1 <= 3, not crop_b, not crop_neg_a.
neg_a(1) :- 1 <= 3, not crop_a.

b(A1) :- c(E1), A1 = 1 + E1, A1 <= 3, not crop_neg_b.
neg_b(3) :- 3 <= 3, not crop_b.

c(0) :- 0 <= 3, not crop_b, not crop_neg_c.
c(2) :- 2 <= 3, not crop_neg_c.

a(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
neg_a(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
a(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
neg_a(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
a(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.
```

```
neg_a(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.

b(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
neg_b(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
b(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
neg_b(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
b(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.
neg_b(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.

c(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
neg_c(E) :- a(E1), neg_a(E2), E = E1 + E2, E <= 3.
c(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
neg_c(E) :- b(E1), neg_b(E2), E = E1 + E2, E <= 3.
c(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.
neg_c(E) :- c(E1), neg_c(E2), E = E1 + E2, E <= 3.
```

The stable models of the above program are:

```
{neg_a(1), neg_b(3), c(0), c(2), crop_neg_a, crop_neg_b, crop_c}

{neg_a(1), b(3), c(2), crop_neg_a, crop_b, crop_c}

{a(1), neg_b(3), c(0), c(2), crop_a, crop_neg_b, crop_c}
```

which correspond exactly to the signature statistical default extensions of Example 2.

5 Conclusions

In this paper we have presented an embedding of a key fragment of Statistical Default Logic into stable model semantics. The embedding is designed to compute the signature set of literals that uniquely distinguishes each extension on a statistical default theory.

Acknowledgement 11 *This research was supported, in part, by grant SES 990-6128 from the National Science Foundation.*

References

[1] Cramér, H. 1946. *Mathematical Methods of Statistics*, Princeton: Princeton University Press.

[2] Eiter, T., N. Leone, C. Mateis, G. Pfeifer and F. Scarcello. 1998 "The KR system: Progress Report, Comparisons and Benchmarks," *KR '98: Principles of Knowledge Representation and Reasoning*, Cohen, A., L. Schubert and S. Shaprio [eds.]. San Francisco: Morgan Kaufmann.

[3] Gelfond, M. and V. Lifschitz 1988. "The Stable Model Semantics for Logic Programming," *Proceedings of the 5th International Conference on Logic Programming*, [ed.] Kowalski, R. and K. Bowen. Cambridge: MIT Press. pp.1070-1080.

[4] Gelfond, M. and V. Lifschitz 1990. "Logic Programs with Classical Negation," *Proceedings of the 7th International Conference on Logic Programming*, Warren, D. and P. Szeredi [eds.]. Cambridge: MIT Press. pp.579-597.

[5] Kyburg, H. E., Jr. and C. M. Teng. 1999. "Statistical Inference as Default Logic," *International Journal of Pattern Recognition and Artificial Intelligence*, 13(2) : 267-283.

[6] Larsen, R. J. and M. L. Marx. 2001. *An Introduction to Mathematical Statistics*, Upper Saddle River, NJ: Prentice Hall.

[7] Marek and Truszczyński 1993. *Nonmonotonic Logic*, Berlin: Springer-Verlag.

[8] Moore, 1979. *Statistics*, San Francisco: W. H. Freeman Press.

[9] Niemelä, I. and P. Simons. 1996. "Efficient Implementation of the Well-founded and Stable Model Semantics," *Proceedings of the Joint International Conference and Symposium on Logic Programming*, Maher, M. [ed.]. Cambridge: MIT Press.

[10] Reiter, R. 1980. "A Logic for Default Reasoning," *Artificial Intelligence*, 13:81-132.

[11] Wheeler, G. R. 2003. "Statistical Default Logic", *Artificial Intelligence*, to appear.