International Journal of Approximate Reasoning ••• (••••) •••–•••

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

Analysing inconsistent information using distance-based measures

John Grant [a], Anthony Hunter [b]

[a] Department of Computer Science and UMIACS, University of Maryland, College Park, MD 20742, USA
[b] Department of Computer Science, University College London, London, WC1E 6BT, UK

ARTICLE INFO

Article history:
Received 28 September 2015
Received in revised form 22 February 2016
Accepted 8 April 2016
Available online xxxx

Keywords:
Inconsistency measurement
Inconsistency analysis
Inconsistency management
Inconsistency tolerance
Propositional logic
Distance measures

ABSTRACT

There have been a number of proposals for measuring inconsistency in a knowledgebase (i.e. a set of logical formulae). These include measures that consider the minimally inconsistent subsets of the knowledgebase, and measures that consider the paraconsistent models (3 or 4 valued models) of the knowledgebase. In this paper, we present a new approach that considers the amount by which each formula has to be weakened in order for the knowledgebase to be consistent. This approach is based on ideas of knowledge merging by Konieczny and Pino-Perez. We show that this approach gives us measures that are different from existing measures, that have desirable properties, and that can take the significance of inconsistencies into account. The latter is useful when we want to differentiate between inconsistencies that have minor significance from inconsistencies that have major significance. We also show how our measures are potentially useful in applications such as evaluating violations of integrity constraints in databases and for deciding how to act on inconsistency.

1. Introduction

Understanding the nature of inconsistency is an important topic if we are to develop autonomous systems that are able to behave intelligently with conflicting information. Although the early work of Grant in [10] showed more than 30 years ago that it is possible to compare inconsistent sets of formulae, the great amount of research on measuring inconsistency occurred in the past decade. It turns out that there are different reasonable ways of measuring the inconsistency of a knowledgebase; these measures tend to be incompatible with one another in the sense that one measure assigns a larger inconsistency value to knowledgebase Δ than to Δ' while another does not.

The purpose of this paper is to introduce several inconsistency measures based on model distance. We work in propositional logic and assume that a knowledgebase contains only consistent formulae (i.e. each individual formula is consistent though the set of formulae may be inconsistent). This is a reasonable assumption as portions of conflicting information are typically consistent. However, we note that every inconsistent formula (other than the special case $\bot$) requires a conjunction; such a formula can always be split into consistent fragments. Every consistent formula has at least one model. We think of each model as a point in Euclidean space. The models of a knowledgebase are exactly the intersection of the set of models for each formula. When the knowledgebase is inconsistent, this intersection is empty.

In our method we use distance measures to measure the distances between models (points in space). The idea of our method is to dilate the points representing the models to regions of space in a minimal way so that the intersection of these

E-mail address: anthony.hunter@ucl.ac.uk (A. Hunter).

http://dx.doi.org/10.1016/j.ijar.2016.04.004
0888-613X/

regions is no longer empty. Our various proposals count different aspects of these dilations to come up with measures of inconsistency. Furthermore, this approach lends itself to assigning weights to atoms thereby capturing better the significance of inconsistencies and provides new insight into the nature of inconsistency. For applications, it offers a better account for distances in the significance of parts of the knowledge that may be inconsistent. We illustrate how the new measures are potentially valuable tools for applications by considering violations of integrity constraints in databases.

The plan of this paper is as follows: (Section 2) We present the basic definitions and terminology; (Section 3) We define distance measures; (Section 4) We present the dilation of formulae; (Section 5) We present our definitions for measuring inconsistency using distance measures; (Section 6) We present our definitions for measuring information in inconsistent information; (Section 7) We show how weighting and costing can be used to take into account the significance of the information; (Section 8) We apply our approach to measure violations of integrity constraints; (Section 9) We compare our distance-based measures with several existing measures; (Section 10) We conclude the paper.

This paper is an extended version of [9]. We augment that paper by providing a systematic analysis of distance-based measures in terms of general properties of inconsistency measures, by showing how distance-based measures can be used for measuring information, by providing coverage of significance in terms of cost functions and cost rankings, and by providing a systematic comparison with key proposals for measures of inconsistency.

2. Preliminaries

We assume a propositional language $\mathcal{L}$ of formulae composed of a finite set of atoms $\mathcal{A} = \{a_1, \ldots, a_n\}$ as well as the logical connectives $\wedge$, $\vee$, $\neg$, and the punctuation symbols (and). Instead of subscripts for a we will often use consecutive letters; for instance (a, b, c) instead of (a_1, a_2, a_3). A literal is an atom or a negated atom. We use ϕ and ψ for arbitrary formulae and α and β for literals. The set of atoms used on the composition of a formula ϕ is given by the function $\mathsf{Atoms}(\phi)$. For example, $\mathsf{Atoms}(\neg a \wedge (b \vee (\neg\neg c \wedge d))) = \{a, b, c, d\}$.

A **knowledgebase** Δ is a finite set of *consistent* formulae. We let $\vdash$ denote the classical **consequence relation**. Logical equivalence is defined in the usual way: $\Delta \equiv \Delta'$ iff $\Delta \vdash \Delta'$ and $\Delta' \vdash \Delta$. We find it useful to define also a stronger notion of equivalence as follows: knowledgebase Δ is **bijection-equivalent** to knowledgebase Δ', denoted $\Delta \equiv_b \Delta'$ iff there is a bijection $f : \Delta \to \Delta'$ such that for all $\phi \in \Delta$, ϕ is logically equivalent to $f(\phi)$. For example, $\{a, b\}$ is logically equivalent but not bijection-equivalent to $\{a \wedge b\}$. We write $\mathbb{R}^{\geq 0}$ (resp. $\mathbb{R}^{+}$) for the set of nonnegative (resp. positive) real numbers and $\mathcal{K}$ for the set of all knowledgebases.

Given a language $\mathcal{L}$ we can assume an arbitrary sequence for the atoms $\mathcal{A}$, say $(a_1, \ldots, a_n)$. Using this sequence of atoms, a **model** (i.e. an interpretation) is a sequence of 0s and 1s, written $(b_1, \ldots, b_n)$ where 0 means false and 1 means true for the corresponding atom. For $\phi \in \mathcal{L}$, $\mathsf{Models}(\phi)$ denotes the set of interpretations for which ϕ is true using the usual evaluation of formulae in classical logic. For a knowledgebase Δ, $\mathsf{Models}(\Delta)$ denotes the set of interpretations for which every $\phi \in \Delta$ is true. So if $\Delta = \{\phi_1, \ldots, \phi_n\}$, then $\mathsf{Models}(\Delta) = \mathsf{Models}(\phi_1) \cap .. \cap \mathsf{Models}(\phi_n))$. We use $\mathcal{M}_{\mathcal{L}}$ to denote the set of models for the language $\mathcal{L}$ (i.e. $\mathcal{M}_{\mathcal{L}}$ contains the 2^n sequences of $(b_0, \ldots, b_n)$ of 0s and 1s).

Often, instead of writing an interpretation as a sequence of 0s and 1s, we will write it as the (unique) binary number $b_1 \ldots b_n$ (with leading 0s kept for easy readability). For example, if $\mathcal{A} = \{a_1, a_2, a_3\}$ and we assume the sequence of atoms (a_1, a_2, a_3), then $\mathcal{M}_{\mathcal{L}} = \{000, 001, 010, 011, 100, 101, 110, 111\}$. Each interpretation can also be represented by a formula. For example, 101 can be represented by the formula $a_1 \wedge \neg a_2 \wedge a_3$.

We introduce a couple of subsidiary functions to analyse interpretations. For an interpretation m, let $\mathsf{Digit}_i(m)$ return the ith digit of m (e.g. $\mathsf{Digit}_2(1010) = 0$), and let $\mathsf{Atom}_i(m)$ return the atom corresponding to the ith digit (e.g. if we assume the sequence of atoms (a, b, c, d), then $\mathsf{Atom}_2(1010) = b$).

We define the set of minimal inconsistent subsets of Δ, denoted $\mathsf{MI}(\Delta)$, as follows (where for a set of formulae Γ, $\Gamma \vdash \bot$ denotes that Γ is inconsistent, and $\Gamma \nvdash \bot$ denotes that Γ is consistent).

$$\mathsf{MI}(\Delta) = \{\Sigma \mid \Sigma \subseteq \Delta \text{ and } \Sigma \vdash \bot \text{ and for all } \Sigma' \subset \Sigma, \Sigma' \nvdash \bot\}$$

Any formula not involved in an inconsistency of a knowledgebase (i.e. it is not in a minimal inconsistent subset of the knowledgebase) is called a **free formula**. Thus the set of free formulae of a knowledgebase Δ is defined as follows.

$$\mathsf{Free}(\Delta) = \{\alpha \in \Delta \mid \alpha \notin \bigcup \mathsf{MI}(\Delta)\}$$

A more restricted notion than that of a free formula is the following notion: A formula α is a **safe formula** in a knowledgebase Δ when α has no atom in common with any other formula in Δ. Hence, the set of safe formulae is defined as follows. (Note that the usual definition of safe formula requires that α be consistent. We do not need this condition because our definition for a knowledgebase given above requires every formula in a knowledgebase to be consistent.)

$$\mathsf{Safe}(\Delta) = \{\alpha \in \Delta \mid \mathsf{Atoms}(\alpha) \cap \mathsf{Atoms}(\Delta \setminus \{\alpha\}) = \emptyset\}$$

A safe formula cannot be involved in any inconsistency; hence every safe formula is free. However, a formula may be free but not safe: for example, a tautology that contains an atom that is also in another formula.

Please cite this article in press as: J. Grant, A. Hunter, Analysing inconsistent information using distance-based measures, Int. J. Approx. Reason. (2016), http://dx.doi.org/10.1016/j.ijar.2016.04.004