INDUCED DEPENDENCIES IN RELATIONAL DATABASES†

John F. Roddick[1] ***and Thomas J. Richards***[2]
Department of Computer Science and Computer Engineering
La Trobe University
Bundoora, VIC 3083, Australia

Abstract

With the growth of database complexity and size it is becoming increasingly necessary to automate the analysis of data. Part of this automation is the development of inductive algorithms to search for characterising rules. The paper introduces the formal concept of an induced dependency as a descriptor for the relationships which characterise the data within a database. This is done within the framework of known functional dependencies held in the schema. Two distinct ways of weakening the completeness of an induction are developed, in terms of which a two dimensional matrix of types of induced dependency is defined.

1. Introduction

The automated extraction of rules and trends from large quantities of data is becoming increasingly necessary as the number, size and complexity of databases increases. These rules provide insight into the structure and content of the data which might not be immediately apparent. The general problem can be summarised as the generation of descriptors and algorithms to automate the analysis of large volumes of data.

The specification of functional dependencies in a relational database indicates relationships about the structure of a world of interest [1, 2]. Inductive inference allows the development of rules or characterisations from examples [3, 4]. Our conjecture is that the induction of functional dependencies would allow the structure of a database to be suggested from the inspection of its data.

Han, Cai and Cercone [5, 6] distinguish between derived rules in two ways. Firstly whether the rule is qualitative or quantitative (ie. whether a measure of conformity to the rule is given) and secondly whether the rule is characteristic (ie. it applies to all data identified by some criteria) or classificatory (ie. the rule enables discrimination between identifiable sets of data).

This paper presents a formalisation of the concept of induced dependencies. It proposes a definition of a *full induced dependency* and some subclasses based on weaker criteria. It will be shown that the distinctions made by Han *et al*. can be properly expressed within the proposed formalism.

2. Induced Dependencies

2.1. Initial Discussion

The concept is first introduced by means of an example. Consider a relational schema populated as follows:

STU	Name	StudentId	School
	BROWN	3	Comp.Sci.
	GREEN	5	Maths
	WHITE	6	Elec.Eng.
	BLACK	7	Maths

---

† Published as J.F. Roddick and T.J. Richards, "Induced dependencies in relational databases", in Proc. Second International Computer Science Conference. Kowloon, Hong Kong, pp. 379-385, 1992.

1 Current Address, School of Computer and Information Science, The Levels Campus, University of South Australia, The Levels, SA 5095, Email: roddick@cis.unisa.edu.au

2 Email: tom@latcs1.lat.oz.au

ENR	Stud-No	Subj-No	Year	Grade
	3	DB1	1990	Pass
	3	OS2	1989	Pass
	5	DB1	1990	Fail
	6	DB1	1990	Pass
	6	PH3	1990	Pass
	7	OS1	1990	Fail
	7	IDB	1991	Fail

LEC	Subject	Year	Lecturer
	DB1	1990	JFR
	DB2	1989	JDP
	IDB	1991	JFR
	OS1	1990	TJR
	OS2	1989	JDP
	PH3	1990	TJR

with known Functional Dependencies (FDs):

STU.StudentId → STU.(Name, School)	F1
ENR.(Stud-No, Subj-No, Year) → ENR.Grade	F2
LEC.(Subject, Year) → LEC.Lecturer	F3
ENR.Stud-No → STU.StudentId	F4
ENR.(Subj-No, Year) → LEC.(Subject, Year)	F5

Induced dependencies can be loosely defined as functional dependencies that are consistent with the data currently held in the database but have not been defined within the database schema. Note that induced dependencies are not necessarily true; they are merely satisfied or appear to be true given the data. Thus we are able to suggest the following Induced Dependencies (IDs):

STU.Name →$^{-}$ STU.StudentId	I1
ENR.Stud-No →$^{-}$ ENR.Grade	I2
LEC.Subject →$^{-}$ LEC.Lecturer	I3
STU.School →$^{-}$ ENR.Grade	I4

Rule I4 shows that induced dependencies, like functional dependencies, may exist across relations and may be read as *"Given only a student's school it is possible to determine the results of subjects taken by that student"*, or put another way, *"Either all students in a school pass all subjects or they all fail all subjects"*. For a cross-relation induced dependency the relations are treated as if they have been joined on their foreign keys, thus the defined functional dependencies are an integral part of the definition of IDs, (see also [7] who investigates the satisfaction of functional dependencies over sets of relations). This follows intuitively from the desire to make use of any known structure of the data. Without the use of these known relationships there is no way to associate tuples in different relations by inspection of the data alone, especially in cases where attribute names and domains differ[3].

2.2. Full Induced Dependencies

The notational conventions adopted in this paper are based on Maier [2]. A relational scheme R is a set of attribute names $\{A_1, \ldots, A_n\}$ where each A_i, $1 \le i \le n$, has a corresponding D_i, called as the domain of A_i. If $D = D_1 \cup D_2 \cup \ldots \cup D_n$ then a relation r on R is the set of mappings or tuples $\{t_1, t_2, \ldots, t_p\} \in r$ from R to D where for each t_j, $1 \le j \le p$, $t_j(A_i)$ must be in D_i, $1 \le i \le n$.

If r is a relation on R, with X and Y subsets of R then the functional dependency $X \rightarrow Y$ holds if and only if for every X-value x, $\pi_Y(\sigma_{X=x}(r))$ has at most one tuple, where π and σ are the *project* and *select* operators as given in [2, p 13 et. seq.].

A definition of the full induced dependency is as follows, (the prefix *full* is used to distinguish it from the weak induced dependency introduced later).

---

[3] In the simple situation it is often possible to assume that similar attributes will have identical names and domains. In practice this is commonly not the case and inducing relationships between relations containing attributes with unlike names should not be prohibited.

Definition - *Full Induced Dependency*

Let R be a set of non-empty relations $r_1, r_2 \ldots r_n$ defined on $R_1, R_2 \ldots R_n$ and let X and Y be a subset of attributes from $R_1, R_2 \ldots R_n$. A full induced dependency between X and Y exists if:

i. The data in R is consistent with the existence of a functional dependency $X \rightarrow Y$,

ii. The closure of functional dependencies defined for R does not imply $X \rightarrow Y$.

Returning to our example earlier, the induced dependencies

`STU.StudentId` $\rightarrow^{-}$ `STU.Name`

`ENR.Stud-No` $\rightarrow^{-}$ `STU.Name`

do not hold as condition ii. requires that IDs be distinct from FDs, both directly and from the transitive closure of FDs. Clearly, IDs could have been defined such that FDs comprised a subset of IDs. In practice however it is felt that IDs and FDs have separate functions and thus condition ii. requires that they be distinct. This also parallels the definition of an inductively strong argument in that it must not be deductively valid [8]. The induced dependency,

`ENR.(Stud-No, Subj-No)` $\rightarrow^{-}$ `ENR.Grade`

holds but is not minimal. Minimality can be enforced by insisting that:

For no proper subset X' of X, does $X' \rightarrow^{-} Y$ hold.

2.3. Value Sensitive Induced Dependencies

The Value Sensitive Induced Dependency (or VSID) extends the concept further. In some cases, only a subset of values in a set of attributes infers values in a second set of attributes or an attribute only infers an identifiable subset of values in a second set of attributes. Consider the case below:

ENR	Stud-No	Subj-No	Year	Grade
	3	DB1	1990	Pass
	3	OS1	1991	Fail
	5	DB1	1990	Pass
	5	OS1	1991	Fail
	5	PG1	1990	Pass
	7	DB2	1991	Fail
	7	HW1	1992	Pass
	7	PG1	1990	Fail
	7	PG2	1991	Fail

While there is no FD or ID from Year to Grade, we can indicate that a Value Sensitive Induced Dependency holds for a selection of the ENR relation as follows:

`ENR.Year[1991]` $\rightarrow^{-}$ `ENR.Grade` I5

This can be read as *"Every subject taken in 1991 had the same grade"*. Similarly restriction can be made on the target of the dependency so that:

`ENR.Year` $\rightarrow^{-}$ `ENR.Grade[Pass]` I6

Formally Value Sensitive Induced Dependencies can be defined as follows:

Definition - *Value Sensitive Induced Dependency*

Let R be a set of non-empty relations $r_1, r_2 \ldots r_n$ defined on $R_1, R_2 \ldots R_n$ and let X and Y be subsets of attributes from $R_1, R_2 \ldots R_n$. Let D_x and D_y be the domains of X and Y resp. and let D'_x be a proper subset of D_x and D'_y be a proper subset of D_y. A value sensitive induced dependency between X and Y exists with respect to D'_x *and* D'_y if the induced dependency $X' \rightarrow^{-} Y'$ holds for $X' = \sigma_{x \in D'_x}(X)$ and $Y' = \sigma_{y \in D'_y}(Y)$.

Three comments should be made about the above definition.

a. When D'_X is equal to D_X and D'_Y is equal to D_Y then the Value Sensitive Induced Dependency simply becomes a Full Induced Dependency as the select operations specify no restriction.

b. Restricting the domain of X does not require a corresponding restriction in the domain of Y and visa-versa.

c. A qualitative measure of the weakening of the full induced dependency is the extent to which D'_X and D'_Y are reduced relative to D_X and D_Y.

Value-Sensitive induced dependencies are similar to the horizontal decompositions discussed by Ceri *et al.* and others [9-14]. These references (especially [13]) discuss the fragmentation of relations within the context of distributed databases.

2.4. Weak Induced Dependencies

In some cases, in particular where there is a large amount of data, the absence of an induced dependency does not imply the absence of an interesting correlation of values. Consider the following:

ENR	Stud-No	Subj-No	Year	Grade	
	3	IS1	1990	Fail	
	3	OS1	1990	Fail	
	3	OS3	1991	Pass	
	3	DB2	1991	Fail	x
	5	IS1	1991	Pass	
	5	OS1	1991	Pass	
	5	PG1	1990	Fail	
	6	DB1	1991	Pass	
	6	DB2	1991	Pass	
	6	HW2	1990	Fail	
	6	PG2	1991	Pass	
	7	EN2	1990	Fail	
	7	HW1	1991	Pass	
	7	PG1	1990	Fail	
	7	PH3	1990	Fail	

In only one case the student's grade was not inducible from the year the subject was attempted (as shown by the tuple marked). This can be written:

$$ENR.Year \rightarrow^{--} ENR.Grade \qquad I7$$

Clearly the definition of an *interesting correlation* is context sensitive. Han *et al.* [6] suggest a *t–threshold* to handle noise and exceptions, ie. fewer than some number of exceptions are ignored. Other possibilities such as confidence thresholds, percentages and the number of standard deviations may be equally appropriate. Choice of these is essentially application sensitive. Where quantification is required we can write:

$$ENR.Year \rightarrow^{--}(98\%)\ ENR.Grade \qquad I8$$

Definition - *Weak Induced Dependency*

Let R be a set of non-empty relations $r_1, r_2 \ldots r_n$ defined on $R_1, R_2 \ldots R_n$ and let X and Y be a subset of attributes from $R_1, R_2 \ldots R_n$. A weak induced dependency $X \rightarrow^{--}(k) Y$ exists if for some arbitrary confidence value k', $k' \leq k$, we can write $X \rightarrow^{-} Y$.

Clearly, k can be viewed as a measure of the weakening.

2.5. Weak Value Sensitive Induced Dependencies

The two concepts above can be amalgamated in cases where a subset of attribute values in an attribute almost determines a value in another attribute. For instance, given the example above we could more accurately write:

$$ENR.Year[1990] \rightarrow^{-} ENR.Grade \qquad I9$$

ENR.Year[1991] $\rightarrow^{--}$ ENR.Grade I10

The significance of Weak VSIDs is an area for further research but clearly the development of induction algorithms must avoid their excessive production.

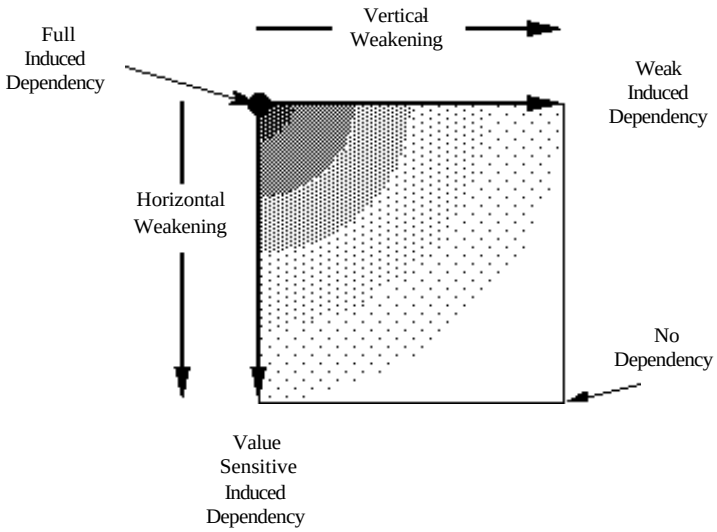
3. Properties of Induced Dependencies

Value sensitive and weak induced dependencies can be considered to differ from full induced dependencies on horizontal or vertical grounds.

Value sensitive induced dependencies can be viewed as a horizontal weakening in that not all tuples adhere to the property and a selection must be performed to create a relation that satisfies full induced dependence, (cf. [12]).

Similarly weak induced dependencies weaken the property in a vertical manner in that the target attributes only adhere to the dependency to a given confidence level. Schematically this can be viewed as in Figure 1 with the shading indicating some intuitive measure of usefulness. Note this is different from the vertical fragmentation discussed in, for example [15], where the two vertical fragments of a relation can be outer joined over the primary key to reform the relation.

Figure 1 - Horizontal and Vertical Weakening in Induced Dependencies



The inference axioms for Functional Dependencies as given by Maier [2] do not always hold for Induced Dependencies. In the axioms below X, Y and Z are sets of attributes taken from R. D'_X, D'_Y and D'_Z represent a (not necessarily proper) subset of D_X, D_Y and D_Z which are the cross products of the domains of X, Y and Z respectively.

a. Reflexivity $X \rightarrow^{-} X$ does **not** hold as $X \rightarrow X$ holds and by condition ii for full induced dependencies. The may seem to run contrary to intuition, however induced dependencies form a set distinct from FDs in the same way as deductively valid and inductively strong arguments form distinct sets.

b. Augmentation $X \rightarrow^{-} Y$ and not $Z \rightarrow Y$ implies $XZ \rightarrow^{-} Y$

c. Additivity $X \rightarrow^{-} Y$ and $X \rightarrow^{-} Z$ implies $X \rightarrow^{-} YZ$

furthermore

$X \rightarrow^{-} Y$ and $X \rightarrow Z$ implies $X \rightarrow^{-} YZ$

d. Projectivity $X \rightarrow^{-} YZ$ and not $X \rightarrow Y$ implies X

$\rightarrow^{-} Y$

e. Transitivity $X \rightarrow^{-} Y$ and $Y \rightarrow^{-} Z$ implies $X \rightarrow^{-} Z$

furthermore

$X \rightarrow^{-} Y$ and $Y \rightarrow Z$ implies $X \rightarrow^{-} Z$

and

$X \rightarrow Y$ and $Y \rightarrow^{-} Z$ implies $X \rightarrow^{-} Z$

f. Pseudotransitivity $X \rightarrow^{-} Y$ and $YZ \rightarrow^{-} W$ implies $XZ \rightarrow^{-} W$

furthermore

$X \rightarrow^{-} Y$ and $YZ \rightarrow W$ implies $XZ \rightarrow^{-} W$

and

$X \rightarrow Y$ and $YZ \rightarrow^{-} W$ implies $XZ \rightarrow^{-} W$

All the above also hold for value-sensitive induced dependencies as long as the same domain subset is used wherever an attribute set is specified. For instance with additivity:

$X[D'_X] \rightarrow^{-} Y[D'_Y]$ and $X[D'_X] \rightarrow^{-} Z[D'_Z]$ implies $X[D'_X] \rightarrow^{-} Y[D'_Y]Z[D'_Z]$

but

$X[D'_X] \rightarrow^{-} Y[D'_Y]$ and $X[D''_X] \rightarrow^{-} Z[D'_Z]$ does not imply $X[D'_X] \rightarrow^{-} Y[D'_Y]Z[D'_Z]$

nor $X[D''_X] \rightarrow^{-} Y[D'_Y]Z[D'_Z]$

4. Related Research

As stated earlier, the work of Han *et al*. [6] characterises induction rules in two ways. Firstly, a rule is quantitative when a measure of conformity to the rule is supplied. Full Induced Dependencies are quantitative in that the measure of conformity to the rule is unity. Weak Induced Dependencies are either qualitative or quantitative depending on whether confidence values are specified.

Secondly, a rule is characteristic if it applies to all data identified by some criteria. Singularly, induced dependencies in all forms as presented here are characteristic. In addition, sets of value sensitive induced dependencies (for example, `I9` and `I10` in section 2.5) may be used to discriminate between sets of data and may thus be used for classification. Furthermore, the strength of the closed world assumption (CWA) may be such to allow a VSID to be implicitly classificatory in so far as they could indicate a dependency that does not necessarily hold for the rest of the domain.

Work on the development of database induction algorithms includes Han *et al*. [5, 6] who present an algorithm based on concept tree ascension that returns a generalised relation by aggregating on higher level concepts. Flach [16] investigates a learning algorithm that allows the induction of multi-valued dependencies from relational data. Significantly, Flach relaxes the CWA and allows the induction process to request negative examples as appropriate. While the CWA is often of limited validity, the ability to request confirmation of a candidate dependency is often limited also.

While there is no requirement for a concept hierarchy table (cf. [5]) it could be an effective method of shorthand for complex VSID. For instance, the VSID

```
ENR.Year[1989] Æ- ENR.Grade[PA,CR,DN,HD]                    I11
```

together with the concept hierarchy

```
{PA,CR,DN,HD} # Pass
```

would allow the VSID to be restated as:

```
ENR.Year[1989] →- ENR.Grade[Pass]                           I12
```

Note that concept hierarchies can weaken the induced rule if not all lower level elements of the higher level concept are present in the VSID.

5. Conclusions and Further Research

This paper presents a formalisation of the concept of induced dependencies which makes implicit use of any predefined functional dependencies. This formalisation is then compared with existing work.

As well as the machine learning aspects of this work we also foresee conventional database management uses including:

i. the flagging of updates that violate induced dependencies as a warning of unusual data entry activity,
ii. performance improvement possibilities by better utilisation of available indexes, query optimisation and data distribution.

Further work is being undertaken to extend the ideas above to cover temporal databases (defined in a manner similar that used in [17-19] and thus enable temporal trends to be induced. An aim of the project is to also handle schema evolution as investigated in [20, 21].

References

[1] E.F. Codd, "A relational model for large shared data banks". *Commun. ACM*. vol. 13, no. 6, pp. 377-387, 1970.

[2] D. Maier, *The theory of relational databases*. Computer Science Press: Rockville, 1983.

[3] E.M. Gold, "Language identification in the limit". *Inf. Control*. vol. 10, pp. 447-474, 1967.

[4] D. Angluin and C.H. Smith, "Inductive inference: theory and methods". *ACM Comput. Surv*. vol. 15, no. 3, pp. 237-269, 1983.

[5] Y. Cai, N. Cercone and J. Han, "An attribute-oriented approach for learning classification rules from relational databases", in *Proc. Sixth IEEE International Conference on Data Engineering*. Los Angeles, CA, IEEE Computer Science Press, pp. 281-288, 1990.

[6] J. Han, Y. Cai and N. Cercone, "Discovery of quantitative rules from large databases", in *Proc. Fifth International Symposium on Methodologies for Intelligent Systems*. Knoxville, TN, North Holland, pp. 157-165, 1990.

[7] P. Honeyman, "Testing satisfaction of functional dependencies". *J. Assoc. Comput. Mach.* vol. 29, no. 3, pp. 668-677, 1982.

[8] B. Skyrms, *Choice and chance, an introduction to inductive logic*. Third edn., Wadsworth Publ.: Belmont, CA, USA, 1986.

[9] P. De Bra and J. Paredaens, "Horizontal decompositions for handling exceptions to functional dependencies", in *Advances in Database Theory II*, Gallaire, H., Minker, J. and Nicolas, J.-M., (eds.), Plenum Press: New York, pp. 123-144, 1984.

[10] A.L. Furtado, "Horizontal decomposition to improve non-BCNF scheme". *SIGMOD Rec*. vol. 12, no. 1, pp. 26-32, 1981.

[11] J. Paredaens, P. De Bra, M. Gyssens and D. Van Gucht, *The structure of the relational database model*. Vol. 17. Springer-Verlag: Berlin, 1989.

[12] S. Ceri, M. Negri and G. Pelagatti, "Horizontal data partitioning in database design". *SIGMOD Rec.* 1982.

[13] S. Ceri and G. Pelagatti, *Distributed databases: principles and systems*. McGraw-Hill: New York, 1984.

[14] P. De Bra and J. Paredaens, "Horizontal decompositions and their impact on query solving". *SIGMOD Rec.* vol. 13, no. 1, pp. 46-50, 1982.

[15] R. Elmasri and S. Navathe, *Fundamentals of database systems*. Benjamin/Cummings: Redwood City, CA, 1989.

[16] P.A. Flach, "Inductive characterisation of database relations", in *Proc. Fifth International Symposium on Methodologies for Intelligent Systems*. Knoxville, TN, North-Holland, Amsterdam, pp. 371-378, 1990.

[17] S. Jones, P. Mason and R. Stamper, "LEGOL 2.0: a relational specification language for complex rules". *Inf. Syst.* vol. 4, no. 4, pp. 293–305, 1979.

[18] J. Ben–Zvi, "The time relational model". Ph.D. thesis, University of California, Los Angeles., 1982.

[19] R. Snodgrass, "The temporal query language TQUEL". *ACM Trans. Database Syst.* vol. 12, no. 2, pp. 247–298, 1987.

[20] L.E. McKenzie and R.T. Snodgrass, "Schema evolution and the relational algebra". *Inf. Syst*. vol. 15, no. 2, pp. 207-232, 1990.

[21] J.F. Roddick, "Dynamically changing schemas within database models". *Aust. Comput. J.* vol. 23, no. 3, pp. 105-109, 1991.