machine learning & knowledge extraction

MDPI

Article

Better than Trees: Applying Semilattices to Balance the Accuracy and Complexity of Machine Learning Models

Stephen Fox [1,*] and Antonio Ricciardo [2]

[1] VTT Technical Research Centre of Finland, 02150 Espoo, Finland
[2] Algebraic AI, 28036 Madrid, Spain; antonio.ricciardo.math@gmail.com
* Correspondence: stephen.fox@vtt.fi

Abstract: Balancing the accuracy and the complexity of models is a well established and ongoing challenge. Models can be misleading if they are not accurate, but models may be incomprehensible if their accuracy depends upon their being complex. In this paper, semilattices are examined as an option for balancing the accuracy and the complexity of machine learning models. This is done with a type of machine learning that is based on semilattices: algebraic machine learning. Unlike trees, semilattices can include connections between elements that are in different hierarchies. Trees are a subclass of semilattices. Hence, semilattices have higher expressive potential than trees. The explanation provided here encompasses diagrammatic semilattices, algebraic semilattices, and interrelationships between them. Machine learning based on semilattices is explained with the practical example of urban food access landscapes, comprising food deserts, food oases, and food swamps. This explanation describes how to formulate an algebraic machine learning model. Overall, it is argued that semilattices are better for balancing the accuracy and complexity of models than trees, and it is explained how algebraic semilattices can be the basis for machine learning models.

Keywords: agreeable AI; algebraic machine learning; food deserts; food oases; food swamps; semilattices; shared interpretability; trees; world models

Academic Editor: Andreas Holzinger

Received: 19 November 2024
Revised: 19 December 2024
Accepted: 2 January 2025
Published: 7 January 2025

Citation: Fox, S.; Ricciardo, A. Better than Trees: Applying Semilattices to Balance the Accuracy and Complexity of Machine Learning Models. *Mach. Learn. Knowl. Extr.* **2025**, *7*, 5. https://doi.org/10.3390/make7010005

1. Introduction

Machine learning models (MLMs) can be described as models-of-the-world. Over the centuries, and still in 2025, models-of-the-world have had many forms, such as two-dimensional (2D) representations on paper and three-dimensional (3D) representations made from physical materials [1]. As summarised by phrases such as the math is not the territory [2] and the map is not the territory [3], models-of-the-world are partial approximations of things in the real world. Although models can be useful [4,5], formulating accurate approximations of some things in the real world can be very difficult [6], and models that are too complex may be practically useless [7]. Hence, models need to balance accuracy and complexity [8–10]. Yet, balancing the accuracy and the complexity of a model can be a formidable challenge, even for much older technologies than MLMs such as 2D diagrams on paper [11] and 3D physical models [12]. Balancing the accuracy and the complexity of machine learning models (MLMs) is an ongoing challenge [13–15].

In the remaining five sections of this paper, semilattices are explained as an option for balancing the accuracy and the complexity of machine learning models. Unlike trees, semilattices can include connections between elements that are in different hierarchies [16,17]. An explanation is provided here with the practical example of urban food access landscapes [18,19]. In Section 2, the accuracy and complexity of semilattices is compared to

trees. In Section 3, interrelationships between diagrammatic semilattices and algebraic semilattices are explained. In Section 4, steps are described in the formulation of a machine learning model based on semilattices. The steps are made using algebraic machine learning [20–22]. In Section 5, an MLM based on semilattices is described and its outputs are presented in diagrammatic semilattices. In conclusion, diagrammatic semilattices and algebraic semilattices are related to challenges in MLM explainability, transparency, interpretability, and shared interpretability [23] in Section 6.

A general review of algebraic machine learning is provided in [24]. By contrast, in this paper, specific details are provided of how to formulate an algebraic machine learning model comprising diagrammatic semilattices and algebraic semilattices. Overall, in this paper, it is argued that semilattices are better for balancing the accuracy and complexity of models than trees, and it is explained how algebraic semilattices can provide the basis for machine learning models through algebraic machine learning.

2. Comparing Trees and Semilattices

Trees are a subclass of semilattices. Hence, semilattices have higher expressive potential than trees. Semilattice-based and tree-based models can be described by Hasse diagrams, which are named after their originator, the mathematician Helmut Hasse (1898–1979). A Hesse diagram is formed by a set of points for the elements of a model and links between the elements of a model. Formulating a model can begin with arranging information in a hierarchy or order, denoted with $\leq$, $\geq$, which is based on one element including all information of another.

For example, as illustrated by the tree-based model in Figure 1, a restaurant includes all the information of its menu, its menu includes all the information of its meal options, and its meal options include all the information about ingredients. There are four levels of elements (restaurant, menu, meal option, ingredients), which are arranged hierarchically one on top of the other.

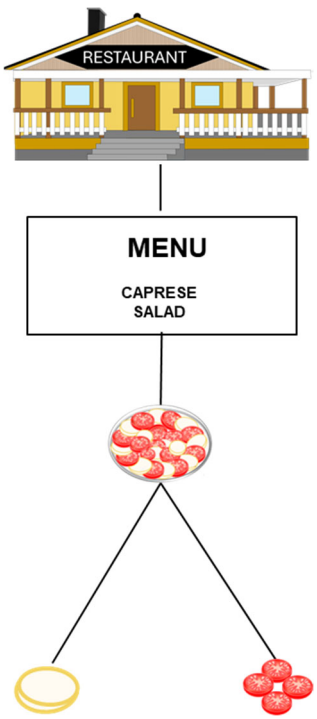

Figure 1. A tree-based model for a restaurant, its menu, one meal option and its ingredients.

The tree-based model shown in Figure 1 is simple because it only comprises a few elements. However, a tree-based model can easily become complex when a model encompasses more elements. Consider, for example, if the restaurant shown in Figure 1 increases its offerings to include the following three meal options: a caprese salad, a cheeseburger, and a basic burger. As shown in Figure 2 below, in a tree-based model, there is the need to repeat the following ingredients: three instances of tomato slices; two instances of lettuce; two instances of bread buns; and two instances of patties. A definition of complexity is the amount of information needed to describe a system [25], and the need for repetition of information in a tree-based model increases the amount of information needed to describe a system. In this example, the system is one restaurant with one menu for three meal options comprising six types of ingredients. However, as shown in Figure 2, there are 11 instances of ingredients when using a tree-based model. Hence, there are five redundant instances of ingredients. By contrast, in the semilattice-based model shown in Figure 2, there is only one instance of each of the six types of ingredient used. Thus, the semilattice-based model is more accurate and less complex than the tree-based model.

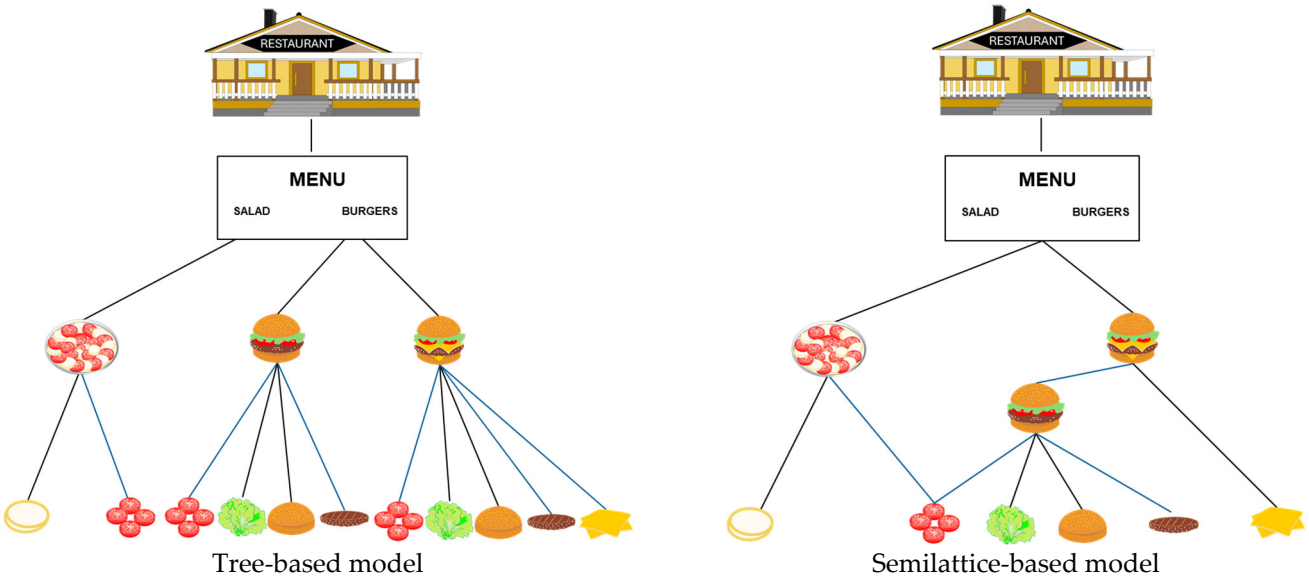

Figure 2. A tree-based model compared to a semilattice-based model for a restaurant, its menu, its meal options and their ingredients.

3. Interrelationships Between Diagrammatic Semilattices and Algebraic Semilattices

In this section, beginning with Figure 3 below, interrelationships between diagrammatic semilattices and algebraic semilattices are explained through practical examples.

A Hasse diagram, more specifically the ordered set it depicts, can be viewed as a semilattice: an algebraic structure whose elements correspond to the points of the diagram endowed with a binary operation, denoted by $\odot$, that, given two elements, say x and y, return a third element $z = x \odot y$, with z the smallest one being greater than both x and y.

For example, we have mozzarella $\odot$ tomato = caprese, which tells us, with as little information as possible, that caprese is the element containing both tomato and mozzarella. This relationship is highlighted in green in Figure 3 below. As highlighted in Figure 4 below, since tomato $\leq$ basic burger, we obtain the following: basic burger $\odot$ tomato = basic burger.

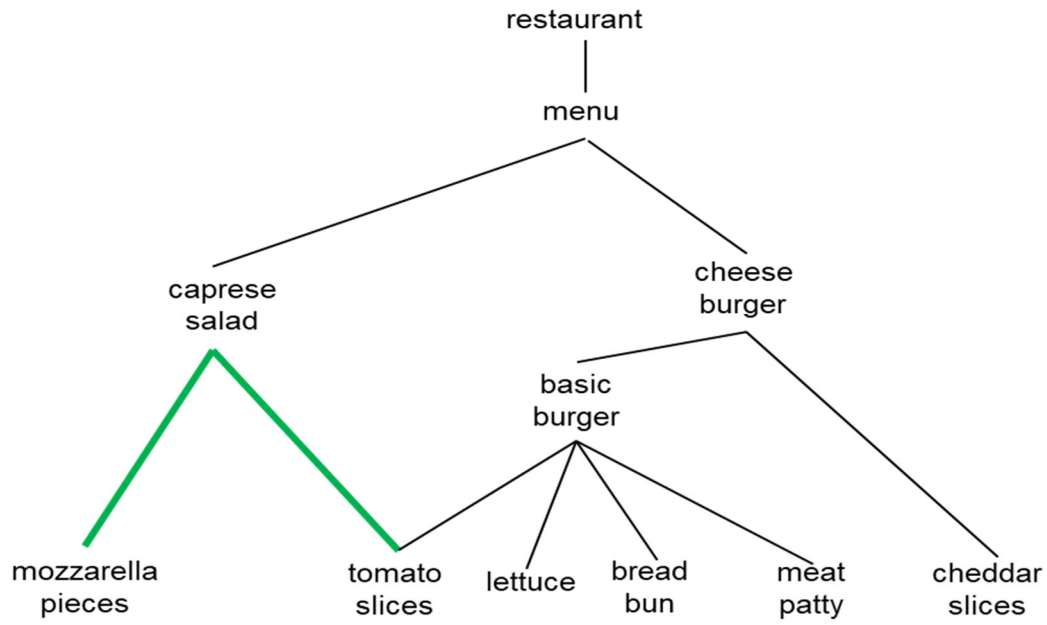

Figure 3. mozzarella ⊙ tomato = caprese.

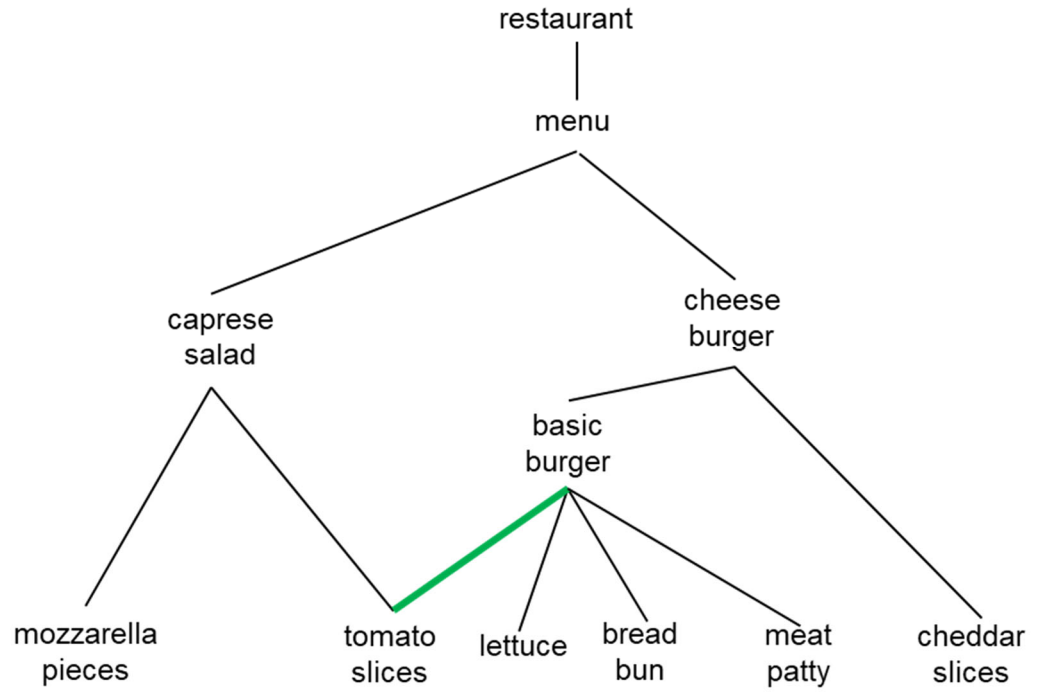

Figure 4. basic burger ⊙ tomato = basic burger.

Starting from the semilattice structure, which we can label M (for model), via its operation ⊙, we can recover the ordered set representation (hence, its Hasse diagram) using the following equation:

$$x \leq y \Longleftrightarrow x \odot y = y$$

For example, if M contains a tomato and a basic burger, and in M we have basic burger ⊙ tomato = basic burger, the order of M will automatically satisfy tomato ≤ basic burger, as highlighted in Figure 4 above. Informally speaking, a semilattice is just a collection of objects (i.e., pieces of information) that can only be summed (joined). A known example of semilattice operation is given by set union, ⊙ = ∪, operating on sets, e.g., {x} ⊙ {y} = {x} ∪ {y} = {x, y}, although a semilattice operation can be "more general" than set union.

The fundamental properties that ⊙ must satisfy in order to be a semilattice operation are simple to describe, with intuitive examples as listed below.

- Joining the information of menu with itself does not bring anything new: menu ⊙ menu = menu. It is idempotent.
- Joining tomato ⊙ mozzarella = caprese is the same as the join mozzarella ⊙ tomato = caprese. It is commutative.
- Operating multiple joins, e.g., (bun ⊙ patty) ⊙ cheese = cheeseburger, does not depend on the order—it is the same as bun ⊙ (patty ⊙ cheese) = cheeseburger. It is associative.

Apart from idempotency, commutativity and associativity characterise the sum between integers +, explaining why ⊙ is referred to as the idempotent sum. We omit idempotent since we are not going to consider the sum between integers, leaving us with no risk of ambiguity. Summing up, formally, we obtain the following:

A semilattice (M,⊙) is a set, M = {x, y, . . .}, with one binary operation ⊙, called the idempotent summation or just the sum, as listed below.

- Idempotent: x ⊙ x = x
- Commutative: x ⊙ y = y ⊙ x
- Associative: x ⊙ (y ⊙ z) = (x ⊙ y) ⊙ z

Without losing generality, we suppose that M is generated by a special set of elements C = {c1, c2, . . .} ⊂ M, which we call constants. They are called constants because they are always present, i.e., constantly present, in the model. For example, in Figure 5, we can take C as equal to all elements except the menu.

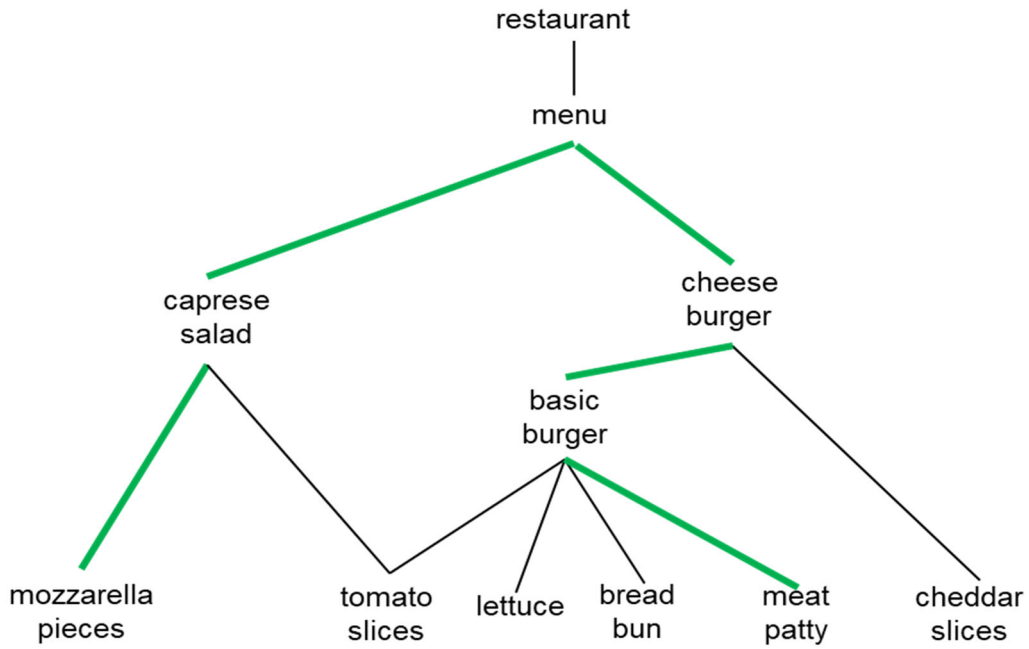

Figure 5. mozzarella ⊙ patty = menu.

The generic element of M, say hamburger ∈ M, is either a constant, which it is, or can be written as a (possibly non-unique) sum of constants. It is also the case that hamburger = bread bun ⊙ patty = bread bun ⊙ tomato. It should be noticed that we could take menu out of C since we have menu = burgers ⊙ caprese. Of course, for readability purposes, it is is easier to have menu instead of burgers ⊙ caprese.

Remark 1. *This example illustrates that the choice of constants is not unique, some can be dropped as we just saw for menu. The set C can be tailored to the model, depending on the application, in order to increase the potential for ease of interpretability.*

Overall, diagrammatic representation has the potential to facilitate human understanding, while algebraic description allows for efficient machine computation when the technical tools described in the next section are used.

4. Machine Learning Models Based on Semilattices

The technical formulations described in the following section are derived from [20–22]. The first step is to embed a real-world problem as a semilattice model. More specifically, the input is given by the set of constants C and the set of relations R between the constants and their sums. The output is the semilattice model, M. This is built by starting from the constants and satisfying the relations in R (as well as their consequences). For example, let C = {mozzarella, tomato, caprese, menu, restaurant}. Then, the set of relations R is as follows:

$$\text{menu} \leq \text{restaurant caprese} \leq \text{menu}$$

$$\text{mozzarella} \leq \text{caprese tomato} \leq \text{caprese}$$

The resulting model is M represented by Figure 1, shown above in Section 2.

One advantage of working directly with the information, without any layer of encoding, is that in the model we build, information is preserved and deductions in said model can be linked back to the original constituents of the model. For example, M can be the model in Figure 2 of a restaurant with two different kinds of burgers: a cheeseburger and a basic burger. As shown in Figure 6, the elements of this model can be renamed $\{m_1, m_2, \ldots, m_n\}$. However, as we need to model more general food access problems than what happens in one particular restaurant, we need to "zoom out" to the urban food access landscape summarized in Figure 7.

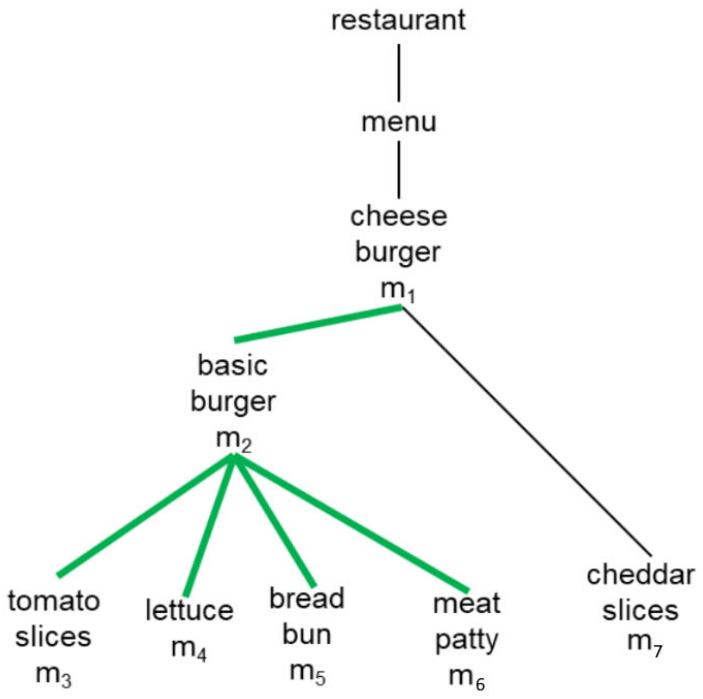

Figure 6. Restaurant offering two types of burger.

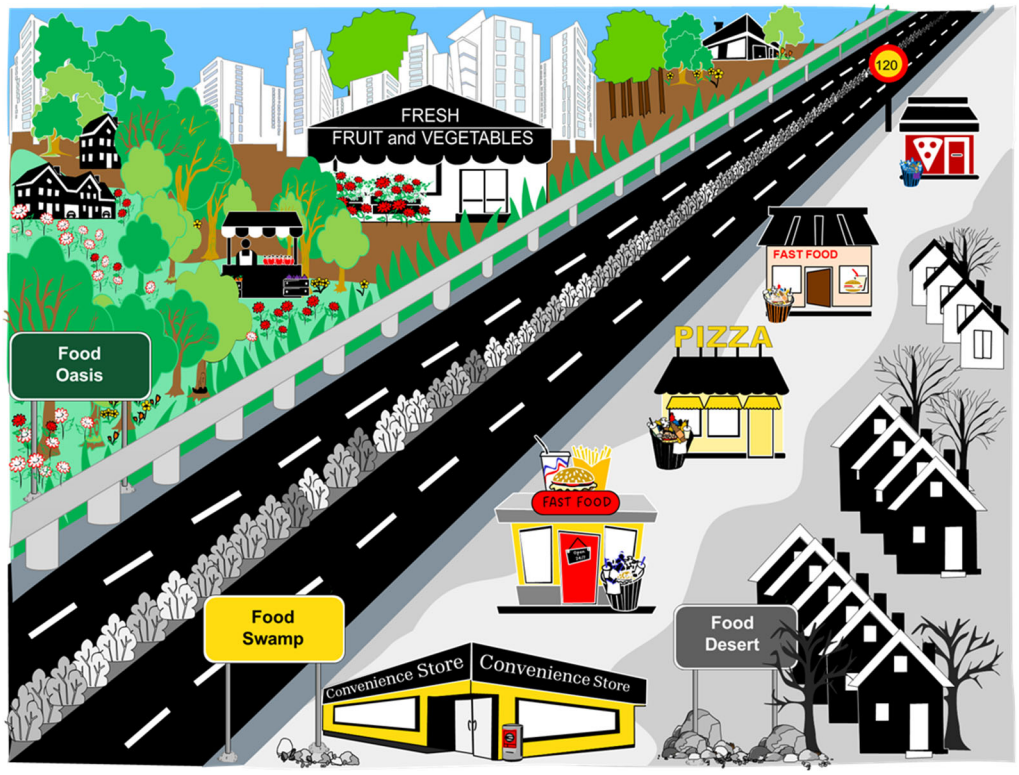

Figure 7. Three areas in an urban food access landscape: food desert, food swamp, and food oasis.

We start building our model step by step. We start with a few constants, and then expand the model. Take the set of constants $C = \{Tr, F_d\}$, with Tr representing a census tract, and F_d the property of being a food desert. We set $F_d \leq Tr$ if Tr is a food desert. Our model will be one of the following options shown below in Figure 8.

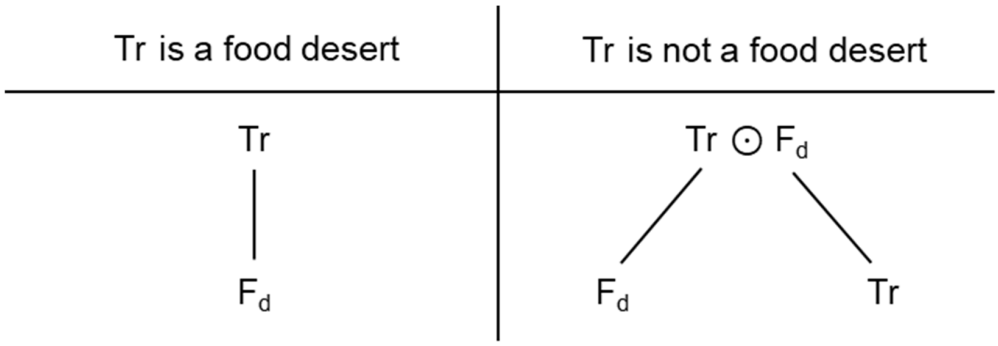

Figure 8. Model options.

Thus, the links and points of the diagram change depending on the properties of the census tract, as does our model as a consequence. The previous model only contained information about a specific census tract Tr and the property of a food desert. In order to build a model for a whole region and to add more properties to the one being analysed, it is reasonable to consider several census tracts, $\{Tr_1, \ldots, Tr_N\}$, and also consider the properties of being a food swamp and a food oasis, F_s and F_o respectively. We denote by M_G the global model over the constants $\{Tr_1, \ldots, Tr_N\} \cup \{F_d, F_s, F_o\}$, with relations given in the following manner:

$F_d \leq Tr_i$ if Tr_i is a food desert;
$F_s \leq Tr_i$ if Tr_i is a food swamp;
$F_o \leq Tr_i$ if Tr_i is a food oasis.

If we reorder the tracts so that the first n_d corresponds to food deserts, the next n_s to food swamps, and the last ones to food oases, we obtain the diagram shown in Figure 9 for our M_G:

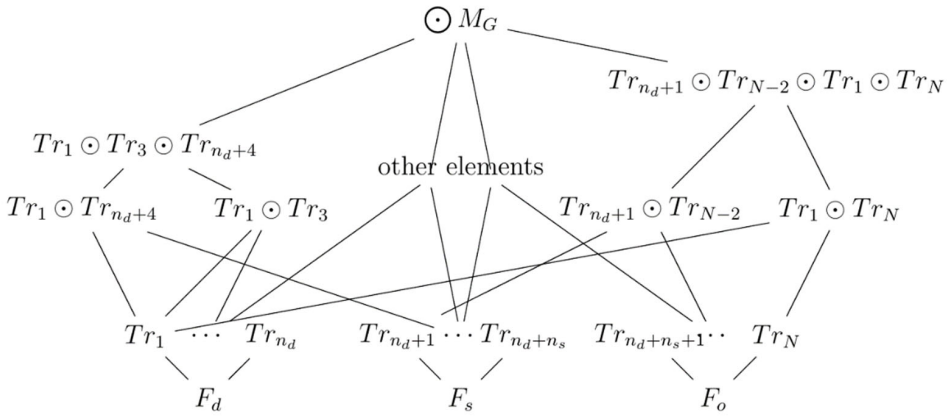

Figure 9. Global model M_G.

The global model M_G has many elements corresponding to some of the possible sums, which can be interpreted as the joint census tract, with top element $\odot M_G$ simply given by the sum of all the constants. Notice that the model has around 2^N elements. Since the number of elements that constitute a model can grow quickly, using either diagrammatic or algebraic semilattices out-of-the-box is computationally problematic, since both representations need us to list every element. To solve this problem, we next discuss briefly an alternative representation for semilattices that, by translating said structure in terms of sets with union, through what we call an atomisation, allows us to efficiently store and manipulate models. Consider M as a model over a set of constants C. For ease of representation, we can think of M as the model of Figure 1. An atom, denoted with Greek letters ϕ, ψ, ..., over C is simply (an object indexed by) a set of constants $U(\phi) \subset C$. Set $\phi < c$ if and only if c is a constant in $U(\phi)$. Examples of atoms are as follows:

$$\phi_{restaurant} < \{\text{restaurant}\}$$

$$\phi_{restaurant,menu} < \{\text{restaurant}, \text{menu}\}$$

Notice that we write ϕ_U to explicitly set $U = U(\phi)$.

Intuitively, we aim to add atoms to the model M consistently with the order of M. For example, in M we know that menu $\leq$ restaurant. If we wish to add an atom ϕ to the picture satisfying $\phi <$ menu, since $\leq$ is an order relation, it must be $\phi <$ menu $\leq$ restaurant.

Hence, as summarised below in Figure 10, in order for ϕ to be compatible, it needs to be $\phi = \phi_{restaurant,menu}$, while ϕ_{menu} contradicts the order of M.

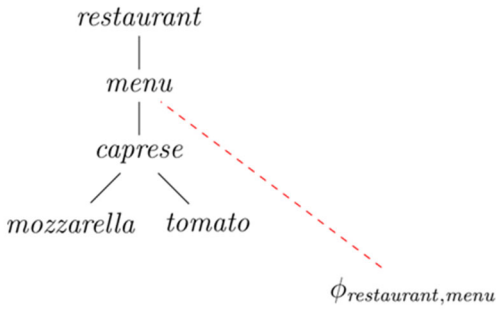

Figure 10. $\phi_{restaurant,menu}$.

Let M be a semilattice. An atomisation for M is a set A = {ϕ1, ϕ2, . . . } of compatible atoms characterising the order of M. The theory developed in [20–22] guarantees us, among other facts that we discuss later, the existence and uniqueness of atomisations.

Theorem 1. *Let M be a semilattice. Then, M has a unique atomisation A with a minimal number of atoms.*

One reason for using atomisations is that they translate a given semilattice to sets with unions. We show this explicitly using M, the model shown in Figure 11. Let A be the atomisation of M given by A = {$\phi_{restaurant}$, $\phi_{restaurant,menu}$}. If we associate to the category of restaurant the atoms lower than restaurant, denoted as L(restaurant), and similarly for menu, then we have menu $\leq$ restaurant $\Longleftrightarrow$ L(menu) $\subseteq$ L(restaurant). As expected {$\phi_{restaurant,menu}$} $\subseteq$ {$\phi_{restaurant}$, $\phi_{restaurant,menu}$}.

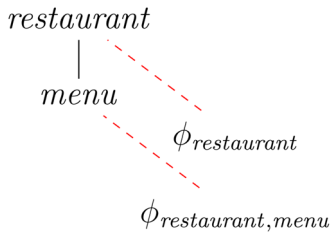

Figure 11. A = {$\phi_{restaurant}$, $\phi_{restaurant,menu}$}.

Equivalently, we have L(menu $\odot$ restaurant) = L(menu) $\cup$ L(restaurant). The previous example is not a coincidence.

Theorem 2. *Let (M,$\odot$) be a semilattice with atomisation A. Then*

$$x \leq y \Longleftrightarrow L(x) \subseteq L(y)$$

$$L(x \odot y) = L(x) \cup L(y).$$

Moreover, the mapping M $\ni$ x $\longleftrightarrow$ L(x) $\subseteq$ A is an equivalence (isomorphism) between semilattices and sets (of atoms).

Remark 2. *We remark that atomisations allow us to compress information about semilattices, while keeping track of constants and their relations. For example, in Figure 9, we have different results for Tr $\odot$ F_d; this is effectively encoded with the atomisation shown in Figure 12 below.*

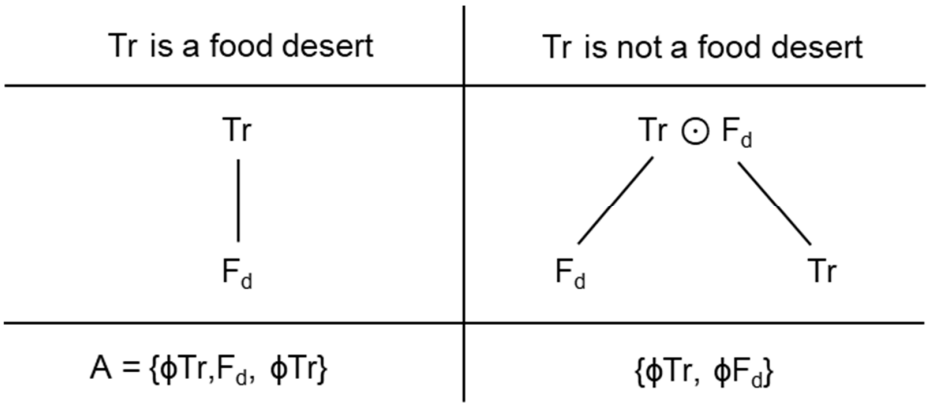

Figure 12. Atomisation of basic model.

The ability of atomisations to effectively compress information is even more evident when considering the model shown in Figure 9. In this case, the 2^N elements of the global model M_G and their order is represented by the N atoms ϕTr_i plus three atoms, each below exactly one of the constants: F_d, F_s, and F_o. For example, we have $\phi_{F_s,Tr_1,Tr_2,\ldots,Tr_{n_s}}$. The model M_G with its atomisation is shown in Figure 13.

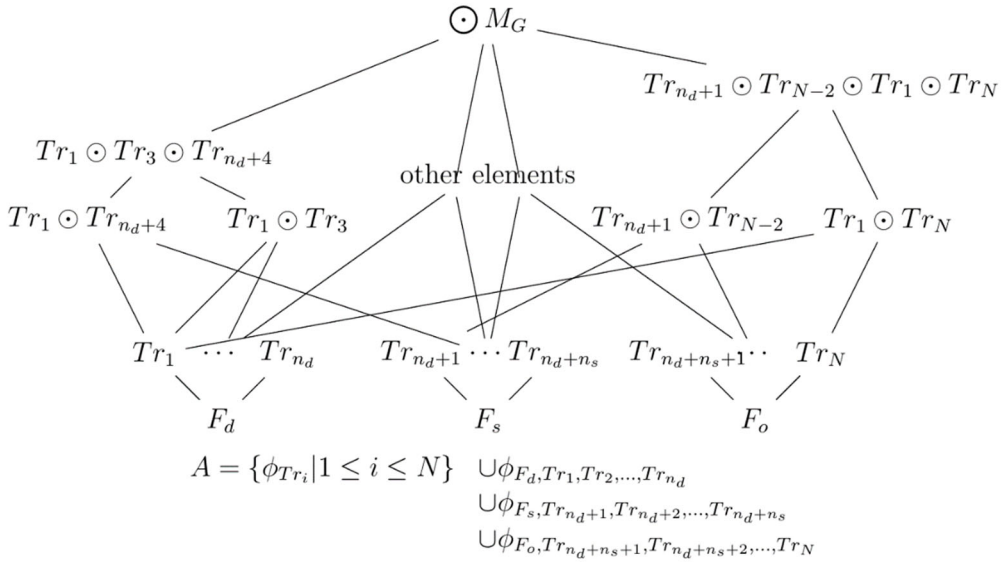

Figure 13. Atomisation of M_G.

Notice that, in the M_G of the example shown in Figure 13, the only information is a given tract being either a food desert, swamp, or oasis (i.e., F_d, F_s, F_o). Hence, this basic model does not carry much information. This changes when we consider other data for a given census tract. Let Tr_i be a constant representing a census tract. We consider several attributes for Tr_i; for example, we can include the average household income. We add constants $\{I_l, I_a, I_h\}$ to represent low, average, and high household incomes, respectively. Since they refer to quantities, we can suppose them to be ordered, i.e., $I_l < I_a < I_h$. Finally, we set Tr_i to be greater than the relative constant, depending on Tr_i being on a given range, e.g., if Tr_i has an average income, we set $Tr_i > I_a$. In this case, we would obtain, considering only Tr_i, the model shown below in Figure 14. The atomisation for which is as follows: $A = \{\phi Tr_i, \phi I_h, \phi Tr_i, I_h, I_a, \phi Tr_i, I_h, Ia, I_l\}$.

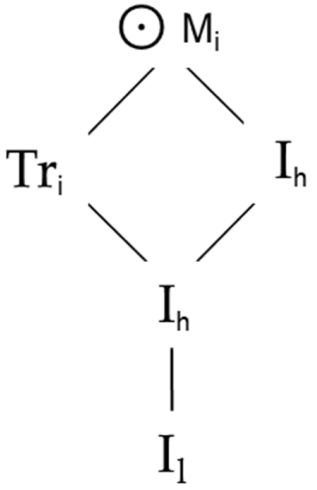

Figure 14. M_2, diagram, and atomisation.

When considering more properties, such as average car availability, education degree of population, etc., the atomisation will become more complex, entailing the relations between tracts, food classifications, and other attributes. We show via an example how this applies to two tracts: Tr_1 and Tr_2. Let M2 be the model with constants for tracts Tr_1 and Tr_2, $\{F_d, F_s, F_o\}$ for the food environment classes, $\{I_l, I_a, I_h\}$ representing income levels, and C representing car availability. Let us suppose that the data we have give us the following model:

$$F_d \leq Tr_1, I_l \leq Tr_1, C \leq Tr_1$$

$$F_d \leq Tr_2, I_l \leq Tr_2, C \not\leq Tr2$$

Thus, the difference between the two tracts is car availability. Consequently, the atomisation for M2 contains, in addition to those with only one constant, the following atoms:

$$\{\phi F_d Tr_1 Tr_2, \phi I_l Tr_1 Tr_2, \phi C Tr_1\}$$

Hence, the atomisation reflects the relations between the data, while compressing the information on the semilattice and allowing us to analyse the semilattice by directly using the atoms.

The model M2, although it perfectly represents the data, is still not able to generalise from such data. In setting $F_d \leq Tr_1$, we made F_d an intrinsic property of the tract, regardless of the income and car availability present in said tract. Since we want to be able to predict F_d from the tract properties, i.e., I_l and C, and since $I_l \odot C \leq Tr_1$, a solution is to make F_d a "property" of the configuration of the tract rather than of the tract itself. In other words, we set

$$F_d \leq I_l \odot C \leq Tr_1 \tag{1}$$

By the transitive property, we still obtain $F_d \leq Tr_1$, but now our model is more refined. Suppose a new tract Tr3 is introduced, with the same income and car availability of Tr1, since we made F_d a consequence of $I_l \odot C$ we automatically obtain $F_d \leq Tr_3$, by $F_d \leq I_l \odot C$ and $I_l \odot C \leq Tr_1$. Hence, Tr_3 inherits the F_d property as a consequence of looking the same as Tr_1, which is perfectly accurate for such a small model. Denoting new model as M3, we see a consequent change in the atomisation:

$$\{\phi F_d I_l Tr_1 Tr_2 Tr_3\ ,\ \phi I_l Tr_1 Tr_2 Tr_3\ ,\ \phi C Tr_1 Tr_3\}$$

The atomisation now encodes the interdependence between F_d and I_l. Since all the food deserts have the property I_l and Tr_2 have only the property I_l, we obtain $F_d \leq I_l$, and accordingly

$$\{\phi F_d I_l Tr_1 Tr_2 Tr_3\} \subseteq \{\phi F_d I_l Tr_1 Tr_2 Tr_3\ ,\ \phi I_l Tr_1 Tr_2 Tr_3\}$$

The process of replacing the income values with a few ranges is a simplification which, while useful for representing and interpreting data, decreases the accuracy of the model. In reality, we do not need to do so; for example, we can add a constant for each value we have for the income of each tract, in order not to lose information, and hence accuracy. Figure 15 shows the general picture of such a model, which we denote as being continuous.

Vice versa, if our model were to become excessively large, then we can switch back to the previous scenario that has ranges. This can be done by setting, for example, I_100 < I_50. As a result, since we already had I_50 < I_100, by definition, all the constants between I_100 and I_50 would collapse to a unique object, interpretable as the range I_[50, 100], which inherits the relations of all its constants, i.e., I_50 < Tr while I_100 < I_50, also yields I_100 < Tr.

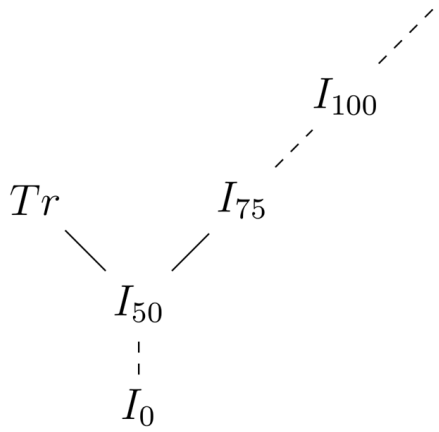

Figure 15. Continuous model.

We end this section by outlining the basics of the implementation that we apply to the food access landscape problem in the next section. As explained in Algorithm 1 below, the main step is to build the model for our data. In other words, we build the set of constants, **C,** containing one constant for each class of food landscape and one for each value attained by each census tract, as well as satisfying the set of relations **R** as Equation (1). This is performed by means of the output atomisation **A**, which completely encodes the model. Subsequently, we use the atomisation A to predict if a test census tract is either a food desert, food swamp or food oasis based on the tract data.

Algorithm 1: M = Model(**R**)

```
C = {F_d, F_s, F_o }, A = { ϕF_d, ϕF_s, ϕF_o}
foreach r = (Tr in class F_x with Property_i = y_i) in R:
    add constants "Property_i = y_i" to C
    set Rel:= F_x < Property_1 = y_1⊙ ...⊙ Property_n = y_n (cf. Equation (1))
    A:= sparseCrossing(Rel, A)
```

More specifically,

1. We initialise the constants, one for each class, and relative atoms.
2. The main cycle: for each tract in the training, pertaining to a given food landscape class and with given properties (also referred to as predictors) we do the following:
3. We add constants for our data;
4. We build the corresponding algebraic relations;
5. We update the atomisation to obtain a model that satisfies the given relation through the sparse crossing algorithm (cf. [20]), which allows us to perform this operation efficiently.

5. Case Study

By following the methods described in the previous sections, and informed by [20–22], we build a model for the food desert landscape in the US in the year 2010, which for the purpose of comparison is informed by [26]. As summarised in Table 1 below, all the necessary information to build our model can be arranged in a table of size 50,212-by-19.

Table 1. Summary of predictors.

Variable	Description	Mean
Observations	Census tracts (2010)	50,212
HH Income	Median household income in the past 12 months (thousand 2010 inflation-adjusted dollars)	54.75 (26.99)
Poverty rate	Tract poverty rate (measured with heterogeneous family-level threshold)	17.13 (12.73)
HH with SNAP	Tract housing units receiving Supplemental Nutrition Assistance Program benefits	219.17 (190.17)
Inequality	Gini Index for the tract (increases in inequality)	0.41 (0.06)
Unemployment	Unemployment rate among population 16 years and over	8.66 (5.61)
Below high school	% of population did not graduate high school	18.58 (14.52)
High school	% of population with high school degree	38.62 (17.05)
Bachelor's or more	% of population with a bachelor's degree or higher	26.63 (18.31)
Property value	Median value of housing unit in thousand dollars	242.37 (192.99)
Public transport	% population using public transportation (excluding taxicab)	2.31 (4.86)
No vehicle	Number of housing units without a vehicle	160.37 (246.28)
Land area	Land area of the tract in square miles	45.83 (622.92)
Population density	Population density per square mile of land area (in thousand)	5.68 (11.78)
Black	African American population (%)	13.50 (22.55)
Hispanic	Hispanic or Latino population (%)	10.95 (12.15)
Asian	Asian population (%)	3.83 (7.52)
White	White population (%)	72.60 (26.39)
Rural population	The percentage of the population living in a rural part of tract (measured by distance from population weighted centroid of a census tract)	17.47 (33.23)
mRFEI	The modified Retail Food Environment Index (%)	11.41 (9.09)

We start with a dataset of 50,212 census tracts, those for which in the given year complete data is available, together with the 18 socio-economic predictors plus 1 for the mRFEI,

which is the indicator used to classify food deserts, swamps and oases, corresponding, respectively, to mRFEI of 0, mRFEI median (9.09) and mRFEI above the median. The reason why the constants Tr_i for the census tracts do not appear in the previous list is that for the purpose of classification, we are more interested in the values attained by the predictors, as it is clear from Equation (1), in contrast to the global model of Figure 9, where we want to explicitly consider all the tracts. Notice, we have constants for US states, as we deem it a relevant feature of the data, each of them having different relevant legislation (taxes and policies). The data used has been retrieved from [27–29]. A short summary of the structuring of the source data is provided in Table 2 below. This shows the starting data entries and the concluding data entries.

Table 2. Structuring of source data.

Census Tract	Class	State	Income	...	Pop-Dens
1	Desert	Alabama	45.96	...	0.9
2	Oasis	Alaska	86	...	4.3
.	.	.	.	.	.
50,212	Swamp	Wyoming	52.5	...	7

Suppose we have a tract in Alabama which is a food desert, Figure 16 below depicts a simplified, partial diagram for this case.

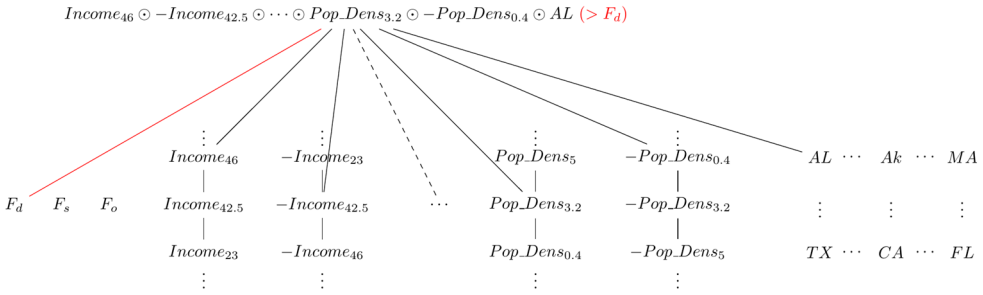

Figure 16. A partial diagram relative to a single food desert.

Using the methods provided by AML, namely, sparse crossing, we are able to derive a model, denoted M, with only 13,000 atoms (we call this set A) out of the much larger set compatible with our relations. More specifically, in choosing a smaller set of atoms we are in fact building a simplified model "focused" on the relations like Equation (1).

Using these atoms, we can predict if a census tract, based on the values of the predictors, is either a desert, swamp, or oasis. In Table 3, we report the accuracy for distinguishing between two classes, excluding a third. For example, between deserts and oases, in a test set excluding swamps. We compare with the accuracy obtained from a straightforward XGB implementation. The overall accuracy from the benchmark study [26] was 72 percent.

Table 3. Accuracy comparison between XGB and AML.

Method	Deserts vs. Oases	Swamps vs. Oases	Deserts vs. Swamps
XGB	68.37%	69.18%	69.56%
AML	69.97%	69.25%	71.51%

The prediction is obtained by comparing the atoms of M that are lower than a given class with those that are lower than the test tract we want to predict. Let us consider the tract Tr represented in Figure 16 and suppose our atomisation is as follows:

$$B = \left\{ \phi_{F_d Income[20,43] PopDens[1,2]}, \phi_{F_s Income[45,50] PopDens[10,13]}, \phi_{F_o Income[51,98] PopDens[3,5]} \right\}$$

where we used the intervals, as Income [20, 43], to abbreviate a set of constants, like {Income21, Income24, . . ., Income42.8}.

It is simple to check that the term corresponding to Tr, which is the sum in Figure 16, has more in common with the atom of B relative to a food desert, rather than to food swamp or food oasis. Consequently, we assign the food desert property to the tract Tr. Clearly, as the number of atoms grows, this prediction becomes more accurate but at the same time becomes more complex. To manage increasing complexity, we select from the 13,000 atoms representing M a smaller subset that contains the atoms that are most relevant with respect to the relations like Equation (1). Thus obtaining a restricted set of around 1300 atoms, which we call A′. There are only 5124 constants appearing in the atoms of A′, and we name this set C′. This means that, inside the global model M, which is able to effectively discriminate between deserts, swamps, and oases, we can find a much smaller model M′, built over the constants of C′ with atomisation A′, which is still a "good" representation of our problem. By good, we mean having high predictive accuracy for this problem while controlling model complexity.

Selecting among those from the restricted set of constants C′, those more frequently appearing in the atomisation A′, we obtain a much smaller sub-model we are actually able to draw. Intuitively, we select the "concepts" of our model that recurrently appear in our classifying model. An enlarged view of some of Figure 17 is provided in Figure 18.

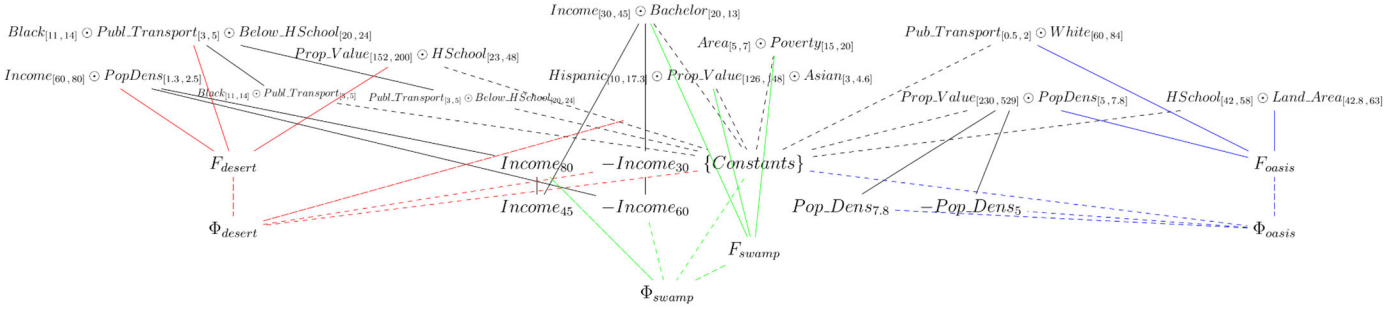

Figure 17. A partial diagram for the restricted model.

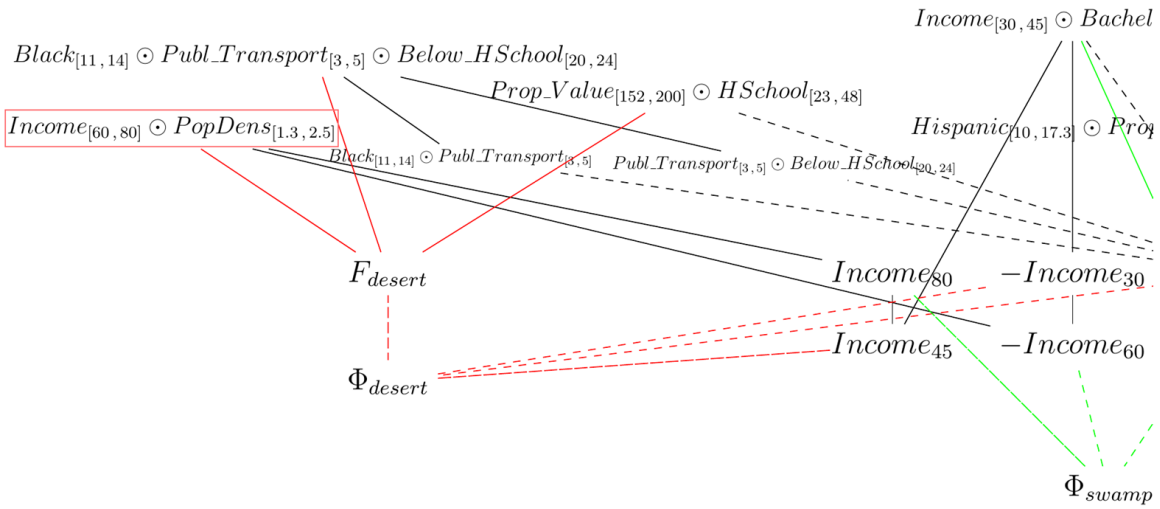

Figure 18. An enlarged detail of the partial diagram for the restricted model.

Figure 17 shows part of the diagram obtained by picking the five most relevant atoms for food deserts, swamps, and oases, denoted with Φ, together with a few selected constants. The diagram shows us some basic criteria for identifying food deserts. For example, a tract Tr that has an income of 70 and a population density of 2 will be classified as a food desert. This is an overly simplified model, as we considered only 15 atoms out of the 1300 of the already-restricted set A′. More refined selection criteria can be used in order to picture the interdependence of a subset of predictors or better analyse the differences between only two of the three classes under investigation.

6. Discussion

6.1. Summary

As the category of semilattices contains that of trees, semilattices have greater representation power than trees. In particular, semilattices enable a system to be represented with fewer vertices than trees, thus providing better compression. The need for fewer vertices is of primary importance because the number of vertices determines the number of edges. The inherent advantages of semilattices are as applicable to machine learning as they are to any other application of semilattices.

As illustrated in Figure 2, semilattices have inherently fewer vertices than trees when modelling the same system, and hence inherently provide a better balance between accuracy and complexity. The more vertices that are in a model, the more complex that model becomes. Consider, for example, Figure 8, which for brevity shows a formulation whereby two vertices for a tract is a food desert and three vertices for a tract is not a food desert. As shown subsequently in Figure 9, the number of vertices can become very large as a system is modelled in more detail. Thus, the development of techniques that yield a reduction in vertices while maintaining the same accuracy is important for balancing accuracy and complexity.

Diagrammatic representation has potential to facilitate human understanding while algebraic description can facilitate efficient machine computation. Diagram—algebra—diagram descriptions can facilitate the explainability, transparency, interpretability, and shared interpretability (ETISI) of models. However, as always, ETISI depends upon minimising the cognitive load of representations [23]. For example, the cognitive load of Figure 9 is higher than the cognitive load of Figure 8. The cognitive load of Figure 2 is smaller than the cognitive loads of Figures 8 and 9. This is because of the work that has been performed to make Figure 2 immediately understandable by using pictures to describe things that people are already familiar with.

Algebraic machine learning (AML) enables the computing of semilattices in durations of time that are short enough to provide timely information to human policymakers. This is because AML uses atoms instead of vertices when computing the models. For example, as shown in Figure 13, we only need N + 3 atoms instead of 2^N vertices. A similar scenario is present in the case study of urban food access landscapes, which otherwise would be intractable. After generating an accurate model with AML for a problem, such as urban food access landscapes, rules can be inferred that policymakers can use to inform their decision-making. For example, the diagram in Figure 18 shows some basic criteria for identifying food deserts, such as tract income and population density.

Unlike other types of machine learning model structures, semilattices are not convoluted. Rather, the simple principles of semilattices are intuitively understandable and can provide the basis for step-by-step traceable modelling. Nonetheless, developing expertise in applying semilattices to machine learning does require practice. Whenever applying AML, practitioners should follow the step-by-step sequence set out in this paper, which starts with a simple diagram and progresses through more detailed diagrams that set out

a sequence of modelling steps and incorporate algebraic descriptions. During its development, this sequence of steps facilitates the evolving comprehension of the modelling as it progresses. Then, when complete, it provides a traceable record, which can facilitate ETISI for others. For example, as illustrated in Figure 2, semilattices incorporate natural and easy-to-understand principles such as the information of higher vertices contains the information of lower vertices. Informed by such principles, the steps set out in this paper, and the methods explained in detail in [20–22], an inexperienced user can begin to address other problems.

6.2. Principal Contributions

The two principal contributions of this paper are to argue that, as summarised in Figure 1 and in Figure 2, semilattices can better enable interpretability than decision trees because semilattices are inherently more accurate and less complex, and, as described in Sections 3–5, algebraic semilattices can enable progression from input diagrammatic semilattices to output diagrammatic semilattices. Through Figures 3–15, the paper includes step-by-step guidance in how to build MLMs with algebraic semilattices, and through Figures 16–18 the paper includes a detailed example of how to apply them to study complex problems by means of algebraic machine learning. The semilattice example in the paper is urban food access landscapes. This example illustrates the need to recognise that MLMs often cannot be sufficient to provide humans with complete explanations. For example, the benchmark machine learning study has a prediction accuracy of 72 percent [26], and the prediction accuracy in the study reported here is not higher.

6.3. Limitations

Prediction accuracy related to urban food access landscapes is lower than, for example, prediction accuracy for road traffic lights, which can be over 95 percent [30]. However, large differences in prediction accuracy can arise from fundamental differences between the things that are being modelled. For example, categories of road traffic lights are much easier to differentiate than categories of urban food access landscapes. Differentiating traffic light categories involves differentiating three colours, which are often organised in configurations that are repeated millions of times throughout the world, such as a red light at the top, an amber light in the middle, and a green light at the bottom. By contrast, differentiating between food deserts, food swamps, and food oases can be a much more difficult problem [31], which can entail disagreements about definitions such as whether an area is a food oasis or is actually a food mirage for many people because it is an area that is inaccessible to them [32]. Especially when MLMs are applied to contribute to explanations of such difficult problems, much human work may still be required such as visiting physical locations in person in order to gain a more complete picture of the situation on the ground. Accordingly, our arguments here are limited to the propositions that semilattices can provide an alternative to decision trees, and that semilattices can contribute to going beyond the currently insufficient contributions from MLMs in explaining complex phenomena. In particular, semilattices can contribute to improving the balance between accuracy and complexity in modelling that involves machine learning. Here, accuracy and complexity are balanced by using algebraic machine learning which, unlike other types of machine learning, enables us to build semilattice models that are represented using atomisations [20–22].

6.4. Directions of Future Research

Improving the explainability, transparency, interpretability, and shared interpretability (ETISI) of machine learning models is an ongoing challenge [23]. This is to be expected because explanation of even well-established constructs that long predate machine learning

continues to be challenging. For example, even explanations of basic mathematics are not yet so effective that every student, everywhere, every time, achieves 10 out of 10 in their school mathematics tests. Rather, some types of basic mathematics, such as long division, can be intrinsically more difficult to explain than others, such as short addition. This can be because of the potential for cognitive load to increase as the number of interacting elements in a subject increases [33]. Furthermore, there is the challenge that there are some phenomena for which there are not yet falsifiable explanations: rather, there are competing unfalsifiable explanations. An example is the ongoing competition between unfalsifiable hypotheses about human consciousness [34]. In addition to hard-to-explain well-established constructs and well-known phenomena without falsifiable explanations, there are new phenomena that are not yet well-defined and that are being described with new constructs. An example is urban food access landscapes, which are described with constructs such as food deserts, food swamps, etc. Whatever type of machine learning modelling is used, there are now more things to be explained: the new phenomena, the new constructs, the machine learning model, and interrelationships between them. Accordingly, it can be expected that explanation will continue to require a lot of hard work.

Less explanation will be required from machine learning models if they are preceded by the application of established techniques such as structural equation modelling, which can provide well-structured initial explanations [35]. Then, these initial explanations can be expressed with analogies, metaphors, and images [36,37]. Figure 7 in Section 4 of this paper provides an example of how images can be used to support explanations of urban food access landscapes that use analogy and metaphor with contrasting similes such as food deserts and food oases. After using analogies, metaphors, and images to reduce the overall burden of explanation upon MLMs, a subsequent challenge is for a MLM to be transparent. Metaphorically, transparency involves a MLM being a "glass box" instead of an opaque "black box". However, transparency does not necessarily improve the interpretability of a MLM by humans because there can be so many things in a MLM that, although it is human-readable, it is not human-intelligible. Consider, for example, this case, where there are 101,225 constants. Humans cannot easily look at 101,225 things simultaneously in detail. Yet, 101,225 constants is a small number of things for humans to try to look at in detail simultaneously compared to the millions of neurons that can be in an artificial neural network involved in deep learning computations [38]. Thus, the interpretability of many types of MLM is not necessarily improved by MLMs being transparent.

By contrast, it has long been argued that machine learning that directly generates decision trees is machine learning that is directly interpretable. However, not all decision trees are equally interpretable, and some decision trees may not be interpretable [39]. Rather, decision trees can entail high cognitive loads, which can lead to individual people's interpretations being based more on their own existing internal world models than on new information within the decision trees [23]. This can prevent shared interpretability, which is needed to achieve consensus in decision-making through what can be described as agreeable AI. Accordingly, a direction for future research is to compare the potential of tree-based machine learning [40] and semilattices to facilitate shared interpretation. Other directions for future research include comparing AML with ensemble models where individual classical models are integrated with machine learning [41] and to models that combine fuzzy sets with machine learning [42]: for example, by extending semilattices and atomisations to the fuzzy set domain as a way to better encode uncertainty. These are appropriate comparators for AML as they are also methods that combine prior knowledge with learning from data.

Author Contributions: Conceptualization, S.F. and A.R.; investigation, S.F. and A.R.; methodology, A.R.; formal analysis, A.R.; writing—original draft preparation, S.F.; writing—review and editing, S.F.; writing—review and editing, A.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by European Union (EU) Horizon 2020 project ALMA grant number 952091.

Data Availability Statement: The data used in this study are from [27–29].

Acknowledgments: The authors thank Fernando Martin-Maroto for technical advice, and Päivi Vahala for graphic design work on Figures 1, 2 and 7.

Conflicts of Interest: S.F. declares no conflicts of interest. A.R. declares his direct involvement as a consultant with the company Algebraic AI, which is the main developer of the AML techniques.

References

1. Wang, C.; Yap, J.B.; Li, H.; Chua, J.; Abdul-Razak, A.S.; Mohd-Rahim, F.A. Topographical survey engineering education retrofitted by computer-aided 3D-printing. *Comput. Appl. Eng. Educ.* **2018**, *26*, 2116–2130. [CrossRef]
2. Andrews, M. The math is not the territory: Navigating the free energy principle. *Biol. Philos.* **2021**, *36*, 30. [CrossRef]
3. Korzybski, A. Science and Sanity: An Introduction to Non-Aristotelian Systems and General Semantics. The International Non-Aristotelian Library Pub. Co.: Lancaster, PA, USA; New York, NY, USA, 1933; pp. 747–761.
4. Box, G.E.P.; Luceño, A.; del Carmen Paniagua-Quinones, M. *Statistical Control by Monitoring and Adjustment*; John Wiley & Sons: Hoboken, NJ, USA, 2009.
5. Hoel, E.P. When the map is better than the territory. *Entropy* **2017**, *19*, 188. [CrossRef]
6. Sokac, A.M. Seeing a coastline paradox in membrane reservoirs. *Dev. Cell* **2017**, *43*, 541–542. [CrossRef]
7. Levins, R. The strategy of model building in population biology. *Am. Sci.* **1966**, *54*, 421–431.
8. Howell, B.; McIntyre, C.C. Analyzing the tradeoff between electrical complexity and accuracy in patient-specific computational models of deep brain stimulation. *J. Neural Eng.* **2016**, *13*, 036023. [CrossRef]
9. Robinson, S. Exploring the relationship between simulation model accuracy and complexity. *J. Oper. Res. Soc.* **2023**, *74*, 1992–2011. [CrossRef]
10. Fabritius, M.; Miermeister, P.; Kraus, W.; Pott, A. A framework for analyzing the accuracy, complexity, and long-term performance of cable-driven parallel robot models. *Mech. Mach. Theory* **2023**, *185*, 105331. [CrossRef]
11. Rogers, S. The McChrystal Afghanistan PowerPoint Slide: Can You Do Any Better? *The Guardian*. 2010. Available online: https://www.theguardian.com/news/datablog/2010/apr/29/mcchrystal-afghanistan-powerpoint-slide (accessed on 18 June 2024).
12. Elliot, L. The Computer Model That Once Explained the British Economy. The Guardian. 2008. Available online: https://www.theguardian.com/business/2008/may/08/bankofenglandgovernor.economics (accessed on 18 June 2024).
13. Ouyang, N.; Song, Y.; Li, Y.; Wu, F.; Yu, H.; Wang, Y.; Sant, G.; Bauchy, M. Predicting concrete's strength by machine learning: Balance between accuracy and complexity of algorithms. *ACI Mater. J.* **2020**, *117*, 125–133.
14. Ray, S. An analysis of computational complexity and accuracy of two supervised machine learning algorithms—K-nearest neighbor and support vector machine. In *Data Management, Analytics and Innovation: Proceedings of ICDMAI*; Springer: Singapore, 2020; Volume 1, pp. 335–347.
15. Gzar, D.A.; Mahmood, A.M.; Abbas, M.K. A comparative study of regression machine learning algorithms: Tradeoff between accuracy and computational complexity. *Math. Model. Eng. Probl.* **2020**, *9*, 1217–1224. [CrossRef]
16. Friedell, M.F. Organizations as semilattices. *Am. Sociol. Rev.* **1967**, *32*, 46–54. [CrossRef]
17. Huang, J.; Cui, Y.; Chang, H.; Obracht-Prondzyńska, H.; Kamrowska-Zaluska, D.; Li, L. A city is not a tree: A multi-city study on street network and urban life. *Landsc. Urban Plan.* **2022**, *226*, 104469. [CrossRef]
18. De Master, K.T.; Daniels, J. Desert wonderings: Reimagining food access mapping. *Agric. Hum. Values* **2019**, *36*, 241–256. [CrossRef]
19. Yang, M.; Wang, H.; Qiu, F. Neighbourhood food environments revisited: When food deserts meet food swamps. *Can. Geogr./Le Géographe Can.* **2020**, *64*, 135–154. [CrossRef]
20. Martin-Maroto, F.; de Polavieja, G.G. Algebraic machine learning. *arXiv* **2018**, arXiv:1803.05252.
21. Martin-Maroto, F.; de Polavieja, G.G. Finite atomized semilattices. *arXiv* **2021**, arXiv:2102.08050.
22. Martin-Maroto, F.; de Polavieja, G.G. Semantic embeddings in semilattices. *arXiv* **2022**, arXiv:2205.12618.
23. Fox, S.; Rey, V.F. A Cognitive Load Theory (CLT) analysis of machine learning explainability, transparency, interpretability, and shared interpretability. *Mach. Learn. Knowl. Extr.* **2024**, *6*, 1494–1509. [CrossRef]

24. Haidar, I.M.; Sliman, L.; Damaj, I.W.; Haidar, A.M. Legacy Versus Algebraic Machine Learning: A Comparative Study. In *International Congress of Electrical and Computer Engineering*, 2nd ed.; Seyman, M.N., Ed.; ICECENG 2023. EAI/Springer Innovations in Communication and Computing; Springer: Cham, Switzerland, 2024.
25. Modis, T. Links between entropy, complexity, and the technological singularity. *Technol. Forecast. Soc. Chang.* **2022**, *176*, 121457. [CrossRef]
26. Amin, M.D.; Badruddoza, S.; McCluskey, J.J. Predicting access to healthful food retailers with machine learning. *Food Policy* **2021**, *99*, 101985. [CrossRef] [PubMed]
27. U.S. Census Bureau. American Community Survey. 2010. Available online: https://www2.census.gov/acs2010_1yr/ (accessed on 1 June 2023).
28. CDC. Census Tract Level State Maps of the Modified Retail Food Environment Index (mRFEI). 2011. Available online: https://stacks.cdc.gov/view/cdc/151050 (accessed on 3 June 2023).
29. Miller, S.; Melendez, R.; Chenoweth, M. *National Neighborhood Data Archive (NaNDA): Urbanicity by Census Tract, United States, 2010*; Inter-University Consortium for Political and Social Research [Distributor]: Ann Arbor, MI, USA, 2021. [CrossRef]
30. Wang, Q.; Zhang, Q.; Liang, X.; Wang, Y.; Zhou, C.; Mikulovich, V.I. Traffic lights detection and recognition method based on the improved YOLOv4 algorithm. *Sensors* **2022**, *22*, 200. [CrossRef] [PubMed]
31. Reid, N.; Gatrell, J.D.; Ross, P.S. Toward a relational geography of local food systems: Or wicked food problems without quick spatial fixes. In *Local Food Systems in Old Industrial Regions*; Reid, N., Gatrell, J.D., Ross, P.S., Eds.; Routledge: London, UK, 2016; pp. 251–260.
32. Short, A.; Guthman, J.; Raskin, S. Food deserts, oases, or mirages? Small markets and community food security in the San Francisco Bay Area. *J. Plan. Educ. Res.* **2007**, *26*, 352–364. [CrossRef]
33. Sweller, J. Cognitive load theory, learning difficulty, and instructional design. *Learn. Instr.* **1994**, *4*, 295–312. [CrossRef]
34. Lenharo, M. *Consciousness Theory Slammed as 'Pseudoscience'-Sparking Uproar*; Nature; Springer Science and Business Media LLC: Berlin/Heidelberg, Germany, 2023. [CrossRef]
35. Wood, E.K.; Stamos, G.; Mitchell, A.J.; Gonoud, R.; Horgan, A.M.; Nomura, O.; Young, A.; Nigg, J.T.; Gustafsson, H.C.; Sullivan, E.L. The association between food desert severity, socioeconomic status, and metabolic state during pregnancy in a prospective longitudinal cohort. *Sci. Rep.* **2023**, *13*, 7197. [CrossRef]
36. Bailer-Jones, D.M. Models, metaphors and analogies. In *The Blackwell Guide to the Philosophy of Science*; Machamer, P., Silberstein, M., Eds.; Blackwell: Cambridge, UK, 2002; pp. 108–127.
37. English, L.D. *Mathematical Reasoning: Analogies, Metaphors, and Images*; Routledge: London, UK, 2013.
38. Liu, S.; Mocanu, D.C.; Matavalam, A.R.R.; Pei, Y.; Pechenizkiy, M. Sparse evolutionary deep learning with over one million artificial neurons on commodity hardware. *Neural Comput. Appl.* **2021**, *33*, 2589–2604. [CrossRef]
39. Costa, V.G.; Pedreira, C.E. Recent advances in decision trees: An updated survey. *Artif. Intell. Rev.* **2023**, *56*, 4765–4800. [CrossRef]
40. Saha, S.; Roy, J.; Hembram, T.K.; Pradhan, B.; Dikshit, A.; Abdul Maulud, K.N.; Alamri, A.M. Comparison between deep learning and tree-based machine learning approaches for landslide susceptibility mapping. *Water* **2021**, *13*, 2664. [CrossRef]
41. Liu, L.L.; Yin, H.D.; Xiao, T.; Huang, L.; Cheng, Y.M. Dynamic prediction of landslide life expectancy using ensemble system incorporating classical prediction models and machine learning. *Geosci. Front.* **2024**, *15*, 101758. [CrossRef]
42. Eftekhari, M.; Mehrpooya, A.; Saberi-Movahed, F.; Torra, V. *How Fuzzy Concepts Contribute to Machine Learning*; Springer: Cham, Switzerland, 2022.