Why

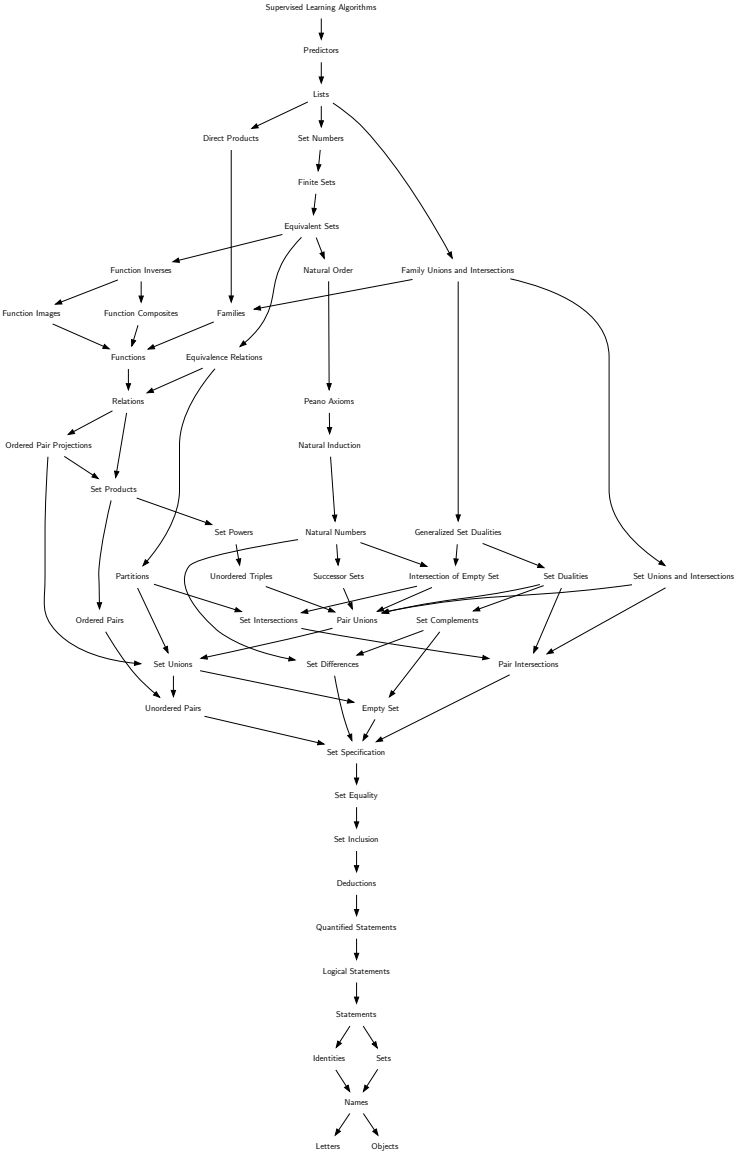
We sometimes use special language for a function inductor, which alludes to its similarities with "learning."

Definition

Let $\mathcal{U}$ and $\mathcal{V}$ be sets and let $\{G_n : (\mathcal{U} \times \mathcal{V})^n \to \mathcal{P}(X \times Y)\}_{n \in \mathbf{N}}$ be a family of functional inductors.

Suppose we have a dataset $(u^1, v^1), \ldots, (u^n, u^n) \in \mathcal{U} \times \mathcal{V}$, and we use our inductor (i.e., learning algorithm) to produce a predictor $f : \mathcal{U} \to \mathcal{Y}$ defined by $f \equiv G_m(a^1, \ldots, a^m)$. Thus, we have a sequence of predictors $f^1, \ldots, f^n$. We can evaluate our learning algorithm on th A predictor can be used to "guess" inputs which do not necessarily appear in the dataset. For this reason, some authors call an inductor (or family of inductors) a *learner* or *learning algorithm*. In accordance with this usage, they refer to the argument of an inductor as the *training data* or *training dataset* or *training set*. As with our terminology dataset, the word "set," however, may mislead since since we are speaking of a sequence.

It is common to refer to the construction a predictor from a dataset a *learning problem*. In this case, the learning problem is said to be *supervised learning*. By supervision, we mean to indicate that we have the outputs corresponding to the inputs. In line with this usage, the outputs are often called *labels* and the labels are said "*supervise*" or "to provide supervision." Since a predictor can be used to *guess* the output of an input, some authors call an inductor (or family of inductors) a *learner* or *learning algorithm* or *supervised learning algorithm* and refer to the argument as the *training dataset*. Often the word "supervised" is included, as in *supervised learning*. This language intends to indicate that inputs are given along with outputs, and these outputs "provide supervision to the algorithm."


Supervised Learning Algorithms
Predictors
Lists
Direct Products
Set Numbers
Finite Sets
Equivalent Sets
Function Inverses
Natural Order
Family Unions and Intersections
Function Images
Function Composites
Families
Functions
Equivalence Relations
Relations
Peano Axioms
Ordered Pair Projections
Natural Induction
Set Products
Set Powers
Natural Numbers
Generalized Set Dualities
Partitions
Unordered Triples
Successor Sets
Intersection of Empty Set
Set Dualities
Set Unions and Intersections
Ordered Pairs
Set Intersections
Pair Unions
Set Complements
Set Unions
Set Differences
Pair Intersections
Unordered Pairs
Empty Set
Set Specification
Set Equality
Set Inclusion
Deductions
Quantified Statements
Logical Statements
Statements
Identities
Sets
Names
Letters
Objects