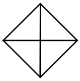


Why

We compare inductors by comparing the relations they produce. We can compare relations by similarity functions.

Definition

Let X and Y be sets. Let $\mathcal{R}$ be the set of relations between X and Y. Let $\mathcal{H} \subset \mathcal{R}$ be a set of hypothesis relations.

Given a similarity function $d : \mathcal{R} \times \mathcal{R} \to \mathsf{R}$ we can consider

A *loss* function is a nonnegative real-valued function on pairs which is zero only on repeated pairs. It need not be symmetric. A loss function is often also called a *cost* or *risk* function.

We use loss functions to judge a prediction in comparison to the recorded postcept. The first argument of the loss function is the predicted postcept and the second is the recorded (observed, true, recorded) postcept.

The *loss of a predictor on a pair* is the result of the loss function on the pair. Similarly, the *loss of a predictor on a dataset* of record pairs is the sum of the losses on the pairs of the dataset. Likewise, the *average loss* of a predictor is the loss of the predictor on the dataset divided by the size of the dataset. The average loss is also known as the *empirical risk* of the predictor on the dataset.

Notation

Let $(a, b) \in A \times B$; $A, B \neq \varnothing$. For a lost function $\ell : B \times B \to \mathsf{R}$ and predictor $f : A \to B$, the loss of f on (a, b) is $\ell(f(a), b)$. Let $s = ((a^1, b^1), \dots, (a^n, b^n))$ be a record sequence. The loss of f on s is $\sum_{k=1}^{n} \ell(f(a^k), b^k)$. The average loss of f on s is $(1/n) \sum_{k=1}^{n} \ell(f(a^k), b^k)$.

Training and test loss

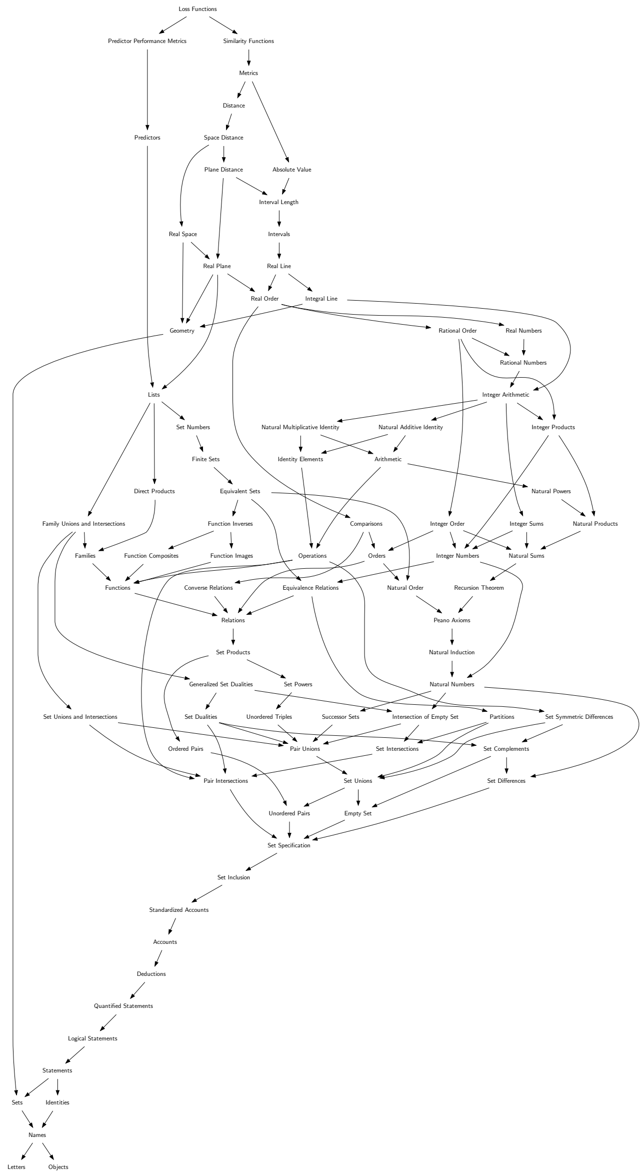
Recall that s is a dataset of records pairs in $A \times B$. We call a predictor $f : A \to B$ an *interpolator* of the dataset s if, for each pair (a^i, b^i) in the dataset, $f(a^i) = b^i$. An interpolator achieves zero loss on the dataset it interpolates.

The rub is that an interpolator may have nonzero loss a record pair which is not contained in the dataset used to construct it. For this reason, it is common to consider two datasets. First, the one used to construct the predictor (the *training dataset*) and second, one used to evaluate the predictor (the *test dataset*, *validation dataset* or *evaluation dataset*).

We judge a predictor by its average loss on the test dataset. We call this the *test loss*, in contrast with the *train loss* obtained on the dataset used to construct the predictor. A predictor whose average test loss is much larger than its average train lost is said to be *overfit* to the train dataset.[1]

Roughly speaking, we judge an inductor by some aggregation metric of the loss of predictors it produces on datasets.

[1] Many authorities will refer to this as the "problem" or "danger" of overfitting.


Loss Functions
Predictor Performance Metrics
Similarity Functions
Metrics
Distance
Predictors
Space Distance
Plane Distance
Absolute Value
Interval Length
Real Space
Intervals
Real Plane
Real Line
Real Order
Integral Line
Geometry
Rational Order
Real Numbers
Rational Numbers
Lists
Integer Arithmetic
Set Numbers
Natural Multiplicative Identity
Natural Additive Identity
Integer Products
Finite Sets
Identity Elements
Arithmetic
Direct Products
Equivalent Sets
Natural Powers
Family Unions and Intersections
Function Inverses
Comparisons
Integer Order
Integer Sums
Natural Products
Families
Function Composites
Function Images
Operations
Orders
Integer Numbers
Natural Sums
Functions
Converse Relations
Equivalence Relations
Natural Order
Recursion Theorem
Relations
Peano Axioms
Set Products
Natural Induction
Generalized Set Dualities
Set Powers
Natural Numbers
Set Unions and Intersections
Set Dualities
Unordered Triples
Successor Sets
Intersection of Empty Set
Partitions
Set Symmetric Differences
Ordered Pairs
Pair Unions
Set Intersections
Set Complements
Pair Intersections
Set Unions
Set Differences
Unordered Pairs
Empty Set
Set Specification
Set Inclusion
Standardized Accounts
Accounts
Deductions
Quantified Statements
Logical Statements
Statements
Sets
Identities
Names
Letters
Objects