Why

We discuss inferring (or learning) functions from examples.

Definitions

Suppose $\mathcal{U}$ and $\mathcal{V}$ are two sets. A *predictor* from $\mathcal{U}$ to $\mathcal{V}$ is a function $f : \mathcal{U} \to \mathcal{V}$. We call $\mathcal{U}$ the *inputs*, $\mathcal{V}$ the *outputs*, and $f(u)$ the *prediction* of f on $u \in \mathcal{U}$.

An *inductor* is a function from datasets in $\mathcal{U} \times \mathcal{V}$ to predictors from $\mathcal{U}$ to $\mathcal{V}$. A *learner* (or *learning algorithm*) is a family of inductors whose index set is $\mathbf{N}$, and whose nth term is is an inductor for a dataset of size n.

Notation

Let D be a dataset of size n in $\mathcal{U} \times \mathcal{V}$. Let $g : \mathcal{U} \to \mathcal{V}$, a predictor, which makes prediction $g(u)$ on input $u \in \mathcal{U}$. Let $G_n : (\mathcal{U} \times \mathcal{V})^n \to (\mathcal{U} \times \mathcal{V})$ be an inductor, so that $G_n(D)$ is the predictor which the inductor associates with dataset D. Then $\{G_n : (\mathcal{U} \times \mathcal{V})^n \to \mathcal{V}^{\mathcal{U}}\}_{n \in \mathbf{N}}$ is a learner.

Relations

Functionsare relations, so we might ask if *inferring* relations may be a more general and difficult problem than inferring functions. The following consideration shows that this is *not* the case.

A *relation inductor* is a function from finite datasets in $\mathcal{U} \times \mathcal{V}$ to *relations* on $\mathcal{U} \times \mathcal{V}$. Suppose R is a relation between $\mathcal{U}$ and $\mathcal{V}$. Suppose the function $f : \mathcal{U} \times \mathcal{V} \to \{0, 1\}$ is such that

$$f(u, v) = 1 \quad \text{if and only if} \quad (u, v) \in R$$

Given f we can find R, and given R we can find f. Thus, instead of learning the *relation* R we can think of learning the *function* f. In other words, if we have an inductor for f, we have a *relation* inductor for R.

Consistent and complete datasets

What can a dataset tell us?

Suppose $D = ((u_i, v_i))_{i=1}^n$ be a dataset and $R \subset X \times Y$ a relation. D is *consistent with R* if $(u_i, v_i) \in R$ for all $i = 1, \ldots, n$. D is *consistent* if there exists a relation with which it is consistent. A dataset is always consistent (take $R = \mathcal{U} \times \mathcal{V}$). D is *functionally consistent* if it is consistent with a function; in this case, $x_i = x_j \Rightarrow y_i = y_j$. D is *functionally complete* if $\cup_i \{x_i\} = X$. In this case, the dataset includes every element of the relation.

Other terminology

Other terms for the inputs include *independent variables*, *explanatory variables*, *precepts*, *covariates*, *patterns*, *instances*, or *observations*. Other terms for the outputs include *dependent variables*, *explained variables*, *postcepts*, *targets*, *outcomes*, *labels* or *observational outcomes*. An input-output pair is sometimes called a *record pair*.

Other terms for a learner include *supervised learning algorithm*. Other terms for a predictor include *input-output mapping*, *prediction rule*, *hypothesis*, *concept*, and *classifier*.

Predictors

Lists

Direct Products

Set Numbers

Finite Sets

Equivalent Sets

Family Unions and Intersections

Natural Order

Function Inverses

Families

Function Images

Function Composites

Equivalence Relations

Functions

Peano Axioms

Relations

Natural Induction

Ordered Pair Projections

Set Products

Generalized Set Dualities

Natural Numbers

Set Powers

Set Unions and Intersections

Set Dualities

Intersection of Empty Set

Successor Sets

Unordered Triples

Set Partitions

Set Complements

Pair Unions

Set Intersections

Ordered Pairs

Pair Intersections

Set Differences

Set Unions

Empty Set

Unordered Pairs

Set Specification

Set Equality

Set Inclusion

Deductions

Quantified Statements

Logical Statements

Statements

Identities

Sets

Names

Letters

Objects