Reflex

On data-driven derivation of discrete mappings between finite spaces

Uwe Beyer Frank Śmieja

Report number: 1995/6

Research Group for Adaptive Systems
SET.AS

On data-driven derivation of discrete mappings between finite spaces

Uwe Beyer Frank Śmieja

German National Research Center for Information Technology (GMD),
Schloß Birlinghoven, 53754 Sankt Augustin,
Germany.

June 1995

Abstract. The guessing of a function behind a discrete mapping between finite spaces using only the information provided by positive examples is shown to be in principle optimally handled by a lookup table combined with a random process. We explore the implications of this result for the construction of intelligent approximators.

Keywords: discrete mapping, approximation, generalization

1 Introduction

The generation of hypotheses about unknown mappings is one of the central aspects of current research in the field of automatic learning. In general a mapping hypothesis is based on a given subset of all possible input/output pairs (examples) belonging to the mapping. In this paper we investigate some properties of the hypothesis space under certain constraints. Given a known distribution of hypotheses and an arbitrarily large but finite space, a number of general theorems can be proven. These theorems give some indications as to the real contributions that can be expected from the use of "sophisticated" or "intelligent" learning methods.

2 Approximation of Binary Mappings

2.1 Definitions and terminology

Definition 1
Let

$$B = \{0, 1\} \tag{1}$$

denote the set of **binary values***.*

Definition 2 (binary mapping)
Let f, defined by

$$\begin{aligned} f^{n,m} : B^n &\rightarrow B^m & (2)\\ \textit{or: } y &= f^{n,m}(x), \quad x \in B^n \; y \in B^m & (3) \end{aligned}$$

with n and m finite, be a **binary mapping** *from an n–dimensional input space to an m–dimensional output space.*

For simplicity we will initially assume that $m = 1$, and as a result will drop the m superscript for the remainder of this section. Let F^n be the set of all functions $B^n \times B$. Then there are in total k^n possible total functions:

$$k^n = |F^n| = 2^{(2^n)} \tag{4}$$

Definition 3 (example)
A tuple $p^n = (x, y)$ where $x \in B^n$, $y \in B$ is

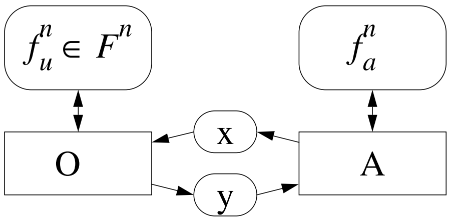

Figure 1: The approximation game

known as an **example** *of an input/output pair with input size n. Denote the set of all examples with input size n by* $E^n = \{p_1^n, \ldots, p_{2^{n+1}}^n\}$.

Definition 4 (valid hypothesis)
The mapping $f \in F^i$ *is a valid hypothesis of a subset* $e^n \subseteq E^n$, *if*

$$ish(f, e^n) \Leftrightarrow \underset{(x,y) \in e^n}{\forall} f(x) = y \tag{5}$$

Definition 5 (valid hypothesis set)
Let $H_{e^n} \subseteq F^n$ *be the set of all valid hypotheses of* $e^n \subseteq E^n$. *Then*

$$\underset{f \in H_{e^n}}{\forall} \quad ish(f, e^n) \tag{6}$$

Definition 6 (example set)
The maximally large subset $u \subset E^n$, *for which*

$$ish(f, u) \tag{7}$$

is called the **example set of a function** $f \in F^n$.

2.2 The Approximation Game

To investigate the properties of approximation of binary mappings, we propose an imaginary "game", illustrated in Figure 1.

The rules of this **approximation game** are as follows:

1. There are two entities in the game: an **oracle** O and an **approximator** A. Both entities agree initially on a value for n, which thereafter remains fixed.

2. O then chooses a particular $f_u^n \in F^n$. The likelihood of choosing any f^n is taken to be $1/k^n$. In other words we assume that the hypothesis distribution is *uniform*. This assumption typifies the first scenario of this game. Later we will consider the effect of having a non-uniform hypothesis distribution. Let U be the example set of f_u^n.

3. A tries to guess O's hypothesis f_u^n. In order to do this A may ask O about the value of f_u^n for any arbitrary $x \in B^n$. In this way a subset of examples $e^n \subseteq U$ is constructed.

4. Each time a new example is acquired A makes a guess f_a^n at f_u^n. Because $|U| = 2^n$ the game must terminate after at most 2^n question–answer sessions.

5. If A guesses f_u^n correctly before it has seen all the 2^n examples it has won.

This scenario typifies the usual method of learning from examples. f_u^n plays the role of an unknown data source to be approximated. The questions posed by A generate its learning set. When A provides a guess it is tested by comparing all values of $f_a^n(x)$ with $f_u^n(x)$ for $x \in B^n$. This is the usual way in which the "generalization" properties of A are tested.

If the game is to be evaluated after many repetitions and on a statistical basis, it is possible to prove a number of theorems related to such an evaluation, when one assumes a uniform hypothesis distribution.

Theorem 1 *There is no guarantee of guessing* f_u^n *correctly before* A *has seen the complete set* U *(in which case of course, it would be no guess!). Every correct unseen example suggested by* f_a^n *is a "lucky shot".*

This can be proven using reverse induction over U: Take the case in which A has seen all but one of the examples in U, i.e. a set $e^n \in U$ containing $2^n - 1$ different examples is available for A. In this case A has just to decide whether $f_u^n(U - e^i) = 0$ or 1. However, assuming a uniform distribution of hypotheses F^n chosen by O,

$$P(f_u^n(x) = y) = \frac{1}{2}, \ \forall y \in B \tag{8}$$

must hold. Therefore no secure decision is possible. In words, there are two remaining candidate hypotheses in H_{e^n}, and the likelihood of their being f_u^n is 0.5 for both of them. It follows that for an example set $e^{n\prime} \subset e^n$, the likelihood of guessing f_u^n is smaller since there are more hypotheses to choose from in $H_{e^{n\prime}}$, each with the same probability of being correct.

The following theorem leads on from this:

Theorem 2 *In an approximation game with a uniform hypothesis distribution, there is no A that can guess f_u^n on average better than one drawing from a uniform distribution of the valid hypotheses H_{e^n}.*

As in Theorem 1 this can be proved using reverse induction over U: Having seen a set e^n of $N \leq 2^n$ examples, A can at best randomly select one of the remaining hypotheses in H_{e^n} with a likelihood $\frac{1}{2^{(2^n - N)}}$, i.e. with a uniform distribution.

Since n is fixed throughout the game it is possible to prove another theorem about the possibility of realizing such an approximator:

Theorem 3 *In an approximation game with uniform hypothesis distribution there exists no algorithm for A that can on average learn quicker, or can be evaluated quicker, than a lookup table combined with a uniform random choice mechanism.*

The proof is done by construction. In order to guess at f_u^n, A must perform the following:

1. Initialize a table T with 2^n entries each with the value `unknown` $\notin B$. Given an $x \in B^n$, x can be interpreted as a binary number uniquely indexing the table by T_x.

2. Now both learning and evaluation of the guess f_a^n can be performed. If the learning of an (x, y) tuple is required, the action $T_x := y$ is performed. If the evaluation of f_a^n is required for $x \in B^n$, then A returns

$$\begin{cases} T_x & | \, T_x \neq \texttt{unknown} \\ \text{any } y \in B & | \, \text{otherwise} \end{cases} \tag{9}$$

From Theorem 2 this represents the optimal answer given the current information and a uniform hypothesis distribution.

Assume the time complexity of a table operation such as T_x to be 1, then the learning of an example and the calculation of a single $f_a^n(x)$ requires exactly one operation. This method seems very time effective, because it always provides the optimal answer with a minimal time complexity. However, the algorithm requires a large capacity since 2^n entries are required for storing U.

Even for small n the table may become so large that the method is *de facto* of a transcomputational nature. In a technical realization of such a situation it is unreasonable to expect the system to see all 2^n examples. Assuming then that only $N = |e^n| \ll 2^n$ examples have been seen by A, the table can be replaced by a list in which the members of e^n are stored. If $f_a^n(x) \;\; x \in B^n$ is to be evaluated, then the T_x operation must be replaced by a search through this list. For a bi-directional linked list the learn and evaluation times grow as $O(\log n)$, but space requirements are reduced correspondingly dramatically.

2.3 Discussion of results for binary mappings

As already mentioned above, all proofs rely on the assumption of a uniform distribution of hypotheses, which transmits the simple clear message:

Under the assumption of a uniform hypothesis distribution there is no hope of postulating an "intelligent" method that can, under the rules of the approximation game, outperform a combination of a simple lookup table and random guessing based on a uniform distribution. All efforts to build "intelligent" approximators that can win the game more often make no sense at all.

Furthermore there is absolutely no hope of any kind of generalization, because the known examples give absolutely no indication as to the form of the the unseen examples.

3 Extension to more general mappings

Theorems 1, 2 and 3 can be extended to more general classes of mappings.

3.1 Mappings with larger output spaces

First, we can increase to size of the output space to any $m > 1$. For mappings from $B^n \rightarrow B^m$ with $n, m > 1$ Theorems 1, 2, and 3 are still valid and the proofs remain basically the same as for $m = 1$. The superscript n must be changed to "n, m" and all equations must be modified accordingly. We will skip the full execution of this because it is simply a formal procedure. Now $k^{n,m}$ has the value

$$k^{n,m} = (2^m)^{(2^n)} \tag{10}$$

Eqn. (8) of Theorem 1 must be modified to:

$$P(f_u^{n,m}(x) = y) = \frac{1}{2^m} \quad \forall y \in B^m, (x, y) \notin e^{n,m} \tag{11}$$

Eqn. (9) of Theorem 3 must be modified to:

$$\begin{cases} T_x & | \; T_x \neq \texttt{unknown} \\ \text{any } y \in B^m & | \; \text{otherwise} \end{cases} \tag{12}$$

Since Theorems 1, 2, and 3 are valid for all $n \geq 1, m \geq 1$, they are also true for all spaces that can be recoded into a binary representation. This includes intervals of N and Q.

Given a concrete problem instance the fixed upper bounds for n and m must be known, otherwise the set $H_{e^{n,m}}$ of valid hypotheses for a given example set $e^{n,m}$ will be infinite. In the latter case no uniform distribution exists with which one may select hypotheses from $H_{e^{n,m}}$.

3.2 Mappings with non-uniform hypothesis distributions

Before we discuss the case of non-uniform distributions, we should point out the significance of the chosen distribution for the selection of $f_u^{n,m}$.

Given a uniform hypothesis distribution ρ_f, a corresponding uniform distribution ρ_e can be derived that gives the probability that an example $(x, y) \in E^{n,m}$ is in U:

$$\forall_{(x,y) \in E^{n,m}} \; P((x, y) \in U) = \frac{1}{2^{n+m}} \tag{13}$$

Because of the finite size of the input and output space, ρ_e can always be derived from ρ_f, e.g. by enumeration of the example sets for each $f^{n,m}$ weighted by ρ_f. This is also possible if ρ_f is non-uniformly distributed.

Let us now consider an **approximation game with non-uniformly distributed hypothesis space**. In the following derivation, we do not allow cases in which one or more $f^{n,m} \in F^{n,m}$ have zero probability of being chosen:

Constraint 1

$$\forall_{f^{n,m} \in F^{n,m}} \quad P(f_u^{n,m} = f^{n,m}) > 0 \tag{14}$$

If A knows ρ_f, then it has more information in order to base its guess of $f_u^{n,m}$. Theorem 1 still applies because Constraint 1 guarantees the possibility of an induction step required in order to prove it. After having seen a set $e^{n,m} \in U$ of

$2^n - 1$ examples, there is still no way of deciding which of the 2^m remaining hypotheses in $H_{e^{n,m}}$ to choose. Because of ρ_f and Constraint 1, they all have a finite probability of being correct, even if one of the hypotheses is much more likely.

Theorem 2 can be reformulated more generally as:

Theorem 4 *In an approximation game with a known hypothesis distribution ρ_f satisfying Constraint 1, there is no A that can guess $f_u^{n,m}$ with a higher expectation of success than a random guess based on ρ'_f.*

The hypothesis distribution ρ'_f is a *renormalized* version of ρ_f that only contains the elements $H_{e^{n,m}}$, thus:

$$\underset{f \in H_{e^{n,m}}}{\forall} \; P_{\rho'_f}(f = f_u^{n,m}) = \frac{1}{R} P_{\rho_f}(f = f_u^{n,m}) \tag{15}$$

where

$$R := \sum_{g \in H_{e^{n,m}}} P_{\rho_f}(g = f_u^{n,m}) \tag{16}$$

The proof is as used for Theorem 2 by using reverse induction on U.

Theorem 3 is also true in principle but it must be reformulated as:

Theorem 5 *For an approximation game with a known hypothesis distribution ρ_f satisfying Constraint 1, there exists no algorithm for A that can learn quicker, and can evaluate quicker, than a lookup table combined with a random choice mechanism based on an example distribution ρ'_e, if an optimal expectation of winning the game is to be ensured.*

The proof is similar to the proof of Theorem 3, with a subtle twist to the question of evaluation speed. If the expectation value of winning is to be maximized, then any algorithm must either explicitly or implictly use the example distribution ρ'_e in the evaluation phase. As each new example is seen, $2^n - 1$ hypotheses will be discarded, and consequently the example distribution will need to be recalculated each time. This is naturally an expensive process, but the explicit method that is used by the lookup table plus random choice algorithm must be the cheapest of these. The explicit method simply enumerates and combines the probability of each example set from all remaining hypotheses.

Eqn. (9) must be reformulated to

$$\begin{cases} T_x & | \; T_x \neq \texttt{unknown} \\ y_{rand} \in B^m & | \; \text{otherwise} \end{cases} \tag{17}$$

where y_{rand} is randomly chosen based on the probability $P_{\rho'_e}(y|x)$.

If ρ_f and (consequently) ρ_e are completely unknown to A, the approximation game makes no sense, because there is absolutely no chance of A behaving in a mathematically motivated way. If every possible hypothesis distribution can be expected, then in particular a uniform one is possible. Therefore it seems unreasonable to expect any quantified statement about which method is expected to win more often. A could simply always assume the hypothesis distribution to be uniform, and on average have more success over many trials in which O uses different distributions, than methods assuming non-uniform distributions.

Thus is seems worthwhile to write down at least the following requirements in an approximation game:

Postulate 1 *The distribution ρ_f must be known, or there must be plausible hints about the form of ρ_f, otherwise there is no chance of doing anything better than using a lookup table containing the known examples.*

In other words:

Postulate 2 *Approximation games without a priori knowledge about hypothesis distributions are meaningless.*

4 Conclusion

If we admit that random choice combined with a lookup table is not the epitome of an "intelligent" algorithm, then we apparently run into a dilemma when it comes to general learning systems. If we assume a uniform hypothesis distribution there is clearly no place for intelligence, given that the input and output spaces are finite. However most problems, especially those implemented on computers, must be represented in a finite way (e.g. using bit coding) to enable an automatic manipulation. Consider the case in which the hypothesis suggested by A is to be used to provide a y for a given x, where (x, y) may be an example that has already been seen. Then for small input spaces the chance that the entry is already stored in the lookup table is generally going to be higher than for large input spaces (the number of possible examples grows as 2^n). So for small n the actual lookup process will tend to be dominant over the random choice process. However, when n is larger random guessing from valid hypothesis will become the dominant process. In any case, this is mathematically the method that will be on average more successful.

In order to resolve this apparent dilemma one may try to remove some constraints:

1. The hypothesis distribution really is unknown
   When this is assumed it seems somewhat unreasonable to expect a mathematically based description of how to generate plausible guesses.

2. Not all $f^{n,m} \in F^{n,m}$ are supported by ρ_f
   If some hypotheses can be discarded right at the beginning, $f_u^{n,m}$ may sometimes be guessed before the whole U has been seen. But this calls for simply a correct bookkeeping for $H_{e^{n,m}}$, and requires no "intelligence".

3. The limits n, m are unknown by definition
   This is the case for mappings between infinite spaces, and means that A can never be sure about n, m. It is another manifestation of case 1, because there is no chance of knowing ρ_f exactly without knowing n, m.

Surprisingly only case 1 remains a plausible area where intelligent behavior might show its uses. The actual distribution may be incompletely known, but one can imagine that certain properties may be known be A (perhaps from O). Then there may be a chance for A to use this knowledge "intelligently" to direct its guesses in a more clever way.

A final observation concerns the form of the approximation game itself. The whole story may look completely different if we reformulate the goal of A to be no longer an *exact* reproduction of $f_u^{n,m}$, but this time really as an approximation. We call this the ϵ **approximation game**:

A should find an $f_a^{n,m}$ that has a quality $Q > \epsilon$ where Q is defined by another entity by comparison with $f_u^{n,m}$ using a given metric.

In this case one can imagine algorithms that enable A to be make a guess before having seen all of U. Seen subjectively, A may appear to behave much more intelligently in this type of game. Nevertheless if the oracle is only used for a certain time to "train" A, and then the resulting hypothesis $f_a^{n,m}$ is to be used in a real system, it remains an inherent property that serious errors may be present in the hypothesis.

Unfortunately it is not possible to use the results of this paper to make positive mathematically based suggestions of how to construct *intelligent* approximators. However some definable cases have been examined where any kind of research in this area is clearly meaningless. We notice that other recent reports [1, 2] partly investigate the same issue, from an automata theory perspective [1] and a probabilistically founded viewpoint [2].

Acknowledgments

Thanks to Gernot Richter for checking the manuscript and Heinz Mühlenbein for bringing the mathematical problem of generalization to our attention. This work is part of the GMD Adaptive Systems REFLEX project, supported by BMBF grant number 01IN111A/4.

References

[1] J. Schmidhuber. Discovering problem solutions with low Kolmogorov complexity and high generalization capability. Technical Report FKI-194-94, Technische Universität München, Munich, Germany, August 1994.

[2] D. Wolpert and W. Macready. No free lunch theorems for search. Technical Report SFI-TR-95-02-010, Santa Fe Institute, Santa Fe, NM, USA, February 1995.