On Similarity Methods in Machine Learning

Mieczyslaw M. Kokar
Department of Electrical and Computer Engineering
Northeastern University
360 Huntington Avenue
Boston, MA 02115, USA

Abstract

This paper reviews a number of uses of similarity theory in the context of machine learning. First, it shows how similarity theory can be used to uncover the fact that some of the relevant variables are missing in a given model. Then, it shows how the same idea can give hints on which variables should be considered as relevant. This is followed by an idea of how dimensional analysis can help in finding physical laws based upon measurements of a physical phenomenon. Finally, the paper introduces the notion of "critical hypersurface" and shows how this notion can be utilized in monitoring time-varying dynamical systems.

1 Introduction

Similarity methods have been proven to be very useful in many applications. The main power of these methods is their predictive ability, i.e., the ability to compute the value of a variable characterizing a given system without even knowing the complete system model. However, while in order to predict we do not need to know the model's equations, we still need to know at least a complete set of the variables that characterize the system. The completeness requirement is crucial since only a complete set of variables gives us the similarity invariants that, in turn, give us the prediction ability. Unfortunately, the search for a complete set of variables is an ill defined task - the search space is too large and thus it is not possible to carry out such a task in any reasonable time. To do this, one would need to identify a "suspect" variable, build an experimental setting, and then test whether this variable influences the system under consideration or not. Even worse, some variables are constant in typical situations; but how can we tell for sure whether our system will stay within the typical regime of operation? Fortunately, similarity theory may provide us with some useful hints on which of the variables should or should not be considered as candidates for the inclusion into the complete set of variables. The idea is to first use a variable as a component in a similarity invariant (similarity number) and then test whether this invariant influences the system or not. Such tests are not straight forward and sometimes are not easy to realize, but in many situations they are possible. The greatest advantage of such an approach is that missing variables can be identified

and "learned" by a computer program (machine learning) during the normal operation of the system. By learning a variable we mean identifying its dimensions with the precise interpretation of the meaning of the variable left to the human designers or operators of the system. In some specific situations even this step can be carried out automatically.

This paper overviews author's work on the application of similarity theory in learning complete sets of variables, model equations that relate such complete sets of variables (physical laws), and learning when to switch from one system model to another in dynamical systems. This approach was first implemented in the system called COPER and later in its intelligent control extension called COPER/IC. COPER/IC was able to handle tasks that are not possible to achieve without the use of the similarity theory.

Section 2 introduces the basic approach to the use of dimensional analysis and similarity theory. Section 3 is devoted to the learning of complete sets of dimensional variables. Section 4 describes an approach to generating functional forms of physical laws. Section 5 connects dimensional analysis and similarity theory with the domain of qualitative simulation. In particular, the notion of critical hypersurface is introduced. In Section 6, this notion is applied to monitoring dynamical systems that change their structure over time. Finally, Section 7 provides conclusions.

2 The Use Patterns of Similarity Theory

The basic theorem of similarity theory, the π-Theorem, has two premises and one conclusion. The premises are:

- there exists a function $Y = F(X_1, \ldots, X_n)$ that relates the variables of interest,
- the function F is invariant with respect to the transformations of systems of units.

The conclusion of the π-theorem is then that such a function can be represented in dimensionless form $\pi_y = f(\pi_1, \ldots, \pi_r)$, where the dimensionless variables $\pi_y, \pi_1, \ldots, \pi_r$ are constructed out of the variables $Y, X_1, \ldots, X_n$ according to the rules of dimensional analysis (e.g., [1]).

Most often this theorem is used in the "forward" direction, i.e., the premises are assumed to be true and thus the dimensionless function f is used to describe the phenomenon of interest, rather than the original function F. Another way of looking at this theorem is to use it in the "backwards" direction, i.e., use its *contrapositive* form. If an implication $A \Longrightarrow B$ is true, then by the rule of contrapositive, the implication $\neg B \Longrightarrow \neg A$ is true, too. By the same rule, if the conclusion of the π-Theorem can be falsified then the premises of the theorem are not true.

So what could be wrong with the premises of the π-Theorem? One suspect could be the second premise, i.e., the invariance with respect to the transformations of systems of units. This premise has been investigated by many researchers, e.g., [16, 17, 18, 15, 20, 10] and there does not seem to be a uniform agreement on why physical laws should be invariant with respect to this group. Nevertheless, most authors attempt to justify such a premise rather than to falsify it. In this paper we will assume that this premise holds. Consequently, we will focus on the first premise, i.e., the existence of a functional dependency among the variables, as specified above.

Table 1: Predicted and measured values of h (complete knowledge)

$V_0[m/s]$	$t[s]$	$g[m/s^2]$	$h[m]$	$h[m]$
0.981	1	9.81	5.886	*
1.962	2	9.81	23.544	23.544
2.943	3	9.81	52.974	52.974
3.924	4	9.81	94.176	94.176
4.905	5	9.81	147.150	147.150
5.886	6	9.81	211.896	211.896
6.867	7	9.81	288.414	288.414
7.848	8	9.81	376.704	376.704

3 Determining a Complete Sets of Variables

In this section we show how we can take advantage of the π-Theorem in the process of learning a complete set of relevant variables. We do this by giving a simple example that shows the steps in such a learning process. A more general treatment of this topic has been presented elsewhere (cf. [4, 6]).

Consider the following formula describing a well known physical law:

$$h = V_0 \cdot t + \frac{g \cdot t^2}{2}, \tag{1}$$

in which the variables are: h - height, V_0 - initial velocity, t - time, g - acceleration due to gravity. But, suppose we don't know this formula and all we do know is that there is a functional dependency of h on V_0, g and t,

$$h = F(V_0, g, t).$$

Applying the π-Theorem we have the dimensionless form

$$\frac{h}{V_0 \cdot t} = f(\frac{g \cdot t}{V_0}). \tag{2}$$

Now suppose we can use the power of the π-Theorem to double check our assumptions. Towards this aim we can design and perform a series of controlled experiments. We design our experiments in such a way that the invariant $\pi_1 = \frac{h}{V_0 \cdot t}$ remains constant (see Table 1). The value of h in this table was computed using the formula given by Equation 1. But this value could be measured directly, although in such a case the result would carry some measurement error.

Since in this case the value of the invariant π_1 was constant, we could calculate the value of h using the π-Theorem. For this, we would need to perform just one measurement, say for the first row, calculate the value $f(\pi_1)$, and then use this value to calculate h according to the formula

$$h = f(\pi_1) \cdot V_0 \cdot t. \tag{3}$$

Table 2: Predicted and measured values of h (lack of knowledge about g)

$V_0[m/s]$	$t[s]$	$g[m/s^2]$	$h[m]$	$h[m]$
7.848	1	9.81	12.753	*
6.867	2	9.81	33.354	22.318
5.886	3	9.81	61.803	28.694
4.905	4	9.81	98.100	31.883
3.924	5	9.81	142.245	31.883
2.943	6	9.81	194.238	28.694
1.962	7	9.81	254.079	22.318
0.981	8	9.81	321.768	12.753

We show this calculated value in Table 1 in column marked as $\bar{h}$. As we can see from the table, the measured and the calculated values are exactly the same.

Now suppose we do not have a complete knowledge of the variables involved, i.e., assume that we are not aware of the relevance of the acceleration g. In this case we could possibly design an experiment shown in Table 2. Since in this case we are not constrained to keep π_1 constant, we are free to vary the variables V_0 and t at will. In Table 2 we used the same values of V_0 as in Table 1 except they are in reverse order. Applying the same process as before, we can calculate the values of h based upon just one experiment (first row). In this case the function f is reduced to a constant, although calculated in exactly the same way as before. Similarly, the values of $\bar{h}$ are computed as before. As we can see, now the calculated values are in great disagreement with the measurements. This should not be a surprise, since in our experiments the value of π_1 was not kept constant, due to our (intentional) ignorance, i.e., as a result of the assumption that g (and consequently π_1) was not relevant.

This could happen at the beginning of the process of completing the relevant variables. The next logical step would be to search for some missing variables. Under some assumptions, the π-Theorem can be useful for learning missing variables. After the failure to confirm the completeness of the set of variables, the learner needs to generate new hypotheses. Suppose the learner generates the hypothesis that the missing relevant variable is some $S[m^2]$, i.e, a variable that represents some area. The new assumption leads to the formulation of the problem as

$$h = F(V_0, t, S). \tag{4}$$

In dimensionless form this takes the form

$$\frac{h}{V_0 \cdot t} = f(\frac{S}{V_0^2 \cdot t^2}). \tag{5}$$

Similarly as before, we design an experiment in which $\pi_1 = \frac{S}{V_0^2 \cdot t^2}$ is kept constant, like in Table 3. Following the same procedure as before for calculating $\bar{h}$, we obtain results that are significantly different than the measured values of h. This is a good

Table 3: Predicted and measured values of h (assuming $S[m^2]$ relevant)

$V_0[m/s]$	$t[s]$	$g[m/s^2]$	$h[m]$	$h[m]$
7.848	1	9.81	12.753	*
3.924	2	9.81	27.468	12.753
2.616	3	9.81	51.993	12.753
1.962	4	9.81	86.328	12.753
1.570	5	9.81	130.473	12.753
1.308	6	9.81	184.428	12.753
1.121	7	9.81	248.193	12.753
0.981	8	9.81	321.768	12.753

indication that our assumption about the variable S being relevant and constant was wrong. Consequently we need to consider a different kind of variable. This procedure is based on the assumption that the newly considered variable (in this example, S) is kept constant. Not necessarily this is the right assumption and thus not necessarily this procedure will give us a correct answer. Nevertheless, it seems to be a relatively good heuristic. We used it many times and it always worked well.

This kind of procedure was implemented in the program called COPER. Various aspects of COPER have been described in various papers (cf. [7, 6]). It is a learning system whose basic learning rule for discovering relevant variables is based on the power of the π-Theorem described above. COPER generates dimensions of suspect variables and tests the prediction of the value of the output variable. It performs an exhaustive search in the space of dimensions. The dimensions are generated by varying all exponents in the dimensional formula

$$X = L^{x_1} M^{x_2} T^{x_3}. \tag{6}$$

The exponents x_1, x_2, x_3 take values from an interval of integers, say (-5 .. 5). It is very rare to see physical laws with exponents higher than 5. The space of dimensions is thus a three dimensional cube in the space of integers, or a subset of a cube in the space of rational numbers.

4 Uncovering Functional Formulas

The process described in the previous section leads to a complete set of relevant variables, i.e., the set of variables that we are dealing with from now on includes all of the variables that are the arguments of function and none of those that are not. In the example of the previous section, the knowledge at this point can be summarized by the signature of the function F,

$$h = F(V_0, g, t). \tag{7}$$

The next step is to find that function F. Dimensional analysis provides some help in this process. Instead of searching for a function of three variables, dimensional analysis

reduces the problem to the problem of searching for a function of just one variable. This is because after transforming Equation 7 to dimensionless form, we need to find a function of just one (dimensionless) variable:

$$\pi_h = f(\pi_1). \tag{8}$$

But still, even in this simple example, we need to find the function f. One way is to perform a number of measurements of the values of the function, assume a type of the function (e.g., linear) and then find the necessary coefficients through the process of approximation (e,g., using the Least Mean Squares function fit). Essentially, under some assumptions, we are guaranteed to find such an approximation with a sufficient degree of accuracy. This feature of function approximation is guaranteed by the Weierstrass theorem [2], which says that a function (satisfying some conditions on continuity) can be approximated with any arbitrarily small degree of accuracy. In our example, this would mean representing f by a polynomial. For instance, we could start with a polynomial of first degree (linear function), find the best coefficients, and go to a higher degree, if the degree of fit is not satisfactory. The problem with this approach is that this procedure may lead to a polynomial of a relatively high degree. In such a case the function is rather difficult to interpret. Typically, more complex functions don't generalize too well, since the higher order components of the polynomial may just fit some local or transient disturbances.

Dimensional analysis gives us another way of performing such a search for a function. Instead of searching the space of polynomials, we can search the space of dimensionless numbers (invariants). It is known that the transformation from a dimensional form to a dimensionless form is not unique. Consequently, one can analyze various such transformations from the point of view of the degree of fit and select the transformation that makes this degree the best.

The question then is what is the space of possible dimensionless representations? In general, it is infinite. But in practice, only a few of the representations are significantly different from others. One way to look at this is to think of a transformation to a dimensionless form as being composed of two steps:

1. Select a *dimensional base* (i.e., a maximal set of variables that are *dimensionally independent*) [1].

2. Express the rest of the variables in terms of the base variables.

Only the first step in this procedure is non-unique. This suggests a method - out of all the relevant variables select all possible dimensional bases. This kind of an exhaustive search approach was implemented in COPER [7]. Various tests have proven that this approach was successful. In this paper we give only a simple example explaining the main idea of this approach. An interested reader is referred to other papers (cf. [3, 5, 7]).

For this example, there are three possible choices of a dimensional base: (V_0, t), (V_0, g) and (g, t). In each case, we end up with the formula:

$$\pi_h = f(\pi_1), \tag{9}$$

Table 4: Results of linear fit to the dimensionless formula

Base	π_h	π_1	C_0	C_1	Fit	Formula
V_0, t	$\frac{h \cdot a}{V_0^2}$	$\frac{a \cdot t}{V_0}$	-186294	2135	$4.78 \cdot 10^6$	$h = -186294\frac{V_0^2}{a} + 2135 V_0 \cdot t$
V_0, a	$\frac{h}{V_0 \cdot t}$	$\frac{a \cdot t}{V_0}$	1.0	0.5	0.014	$1.0 V_0 \cdot t + 0.5 a \cdot t^2$
a, t	$\frac{h}{a \cdot t^2}$	$\frac{V_0}{a \cdot t}$	0.5	1.0	$1.84 \cdot 10^{-4}$	$1.0 V_0 \cdot t + 0.5 a \cdot t^2$

although π_h and π_1 are expressed by different formulas. We show the results for the simplest form of f, i.e., linear

$$\pi_h = C_0 + C_1 \cdot \pi_1 \tag{10}$$

The results of this experiment are summarized in Table 4. The first column of this table lists the components of each basis. The two following columns show the formulas for π_h and π_1. This is followed by the values of the coefficients C_0 and C_1 obtained through function fitting for each case. And finally, the last column reveals the mystery - the fit for the two bases, (V_0, g) and (g, t), (shown in the sixth column) is so good due to a very simple reason: the resulting formula, although it is not quite obvious from Equation 10, is exactly the same as the original physical law represented by Equation 1! It seems that this result is rather counterintuitive. The Weierstrass theorem would suggest that we should look for a higher degree polynomial, and yet, we were able to achieve this goal without making this step.

5 Critical Hypersurfaces

The problems discussed in the previous sections are all examples of "quantitative modeling". But in practice, so called *qualitative models* are also used. One way to view qualitative models is as being *abstractions* of quantitative models. In this view, quantitative variables give rise to qualitative variables. Most typically, qualitative variables represents intervals of quantitative variables. For instance, if we deal with flows of fluids, we might be interested whether the flow is "laminar", "turbulent" or "transitional". In other words, we would create a qualitative variable of "flow-type" having those three qualitative values. This kind of variables have been intensively investigated in the area known as *qualitative reasoning* (QR), which is a subarea of artificial intelligence (AI).

A simple idea to determine the semantics of these qualitative values would be to establish some "critical values", or "landmark points" (as it is known in the QR world) of flow velocity and say that below the first landmark the flow is laminar, above the second it is turbulent, and transitive in between. While this kind of approach is acceptable in AI, the similarity theory researchers know that a better approximation of this partition is the Reynolds number, rather than just the velocity itself. This observation was formulated in [8] as a theory of *critical hypersurfaces*. In the case of flows of fluids, two hypersurfaces can be defined by fixing the Reynolds number to $Re_l = 2000$ and 3000. The two values

of Re_l constitute the landmark points. The hypersurfaces are defined by

$$\frac{\rho \cdot v \cdot D}{\eta} = Re_l \tag{11}$$

where ρ denotes the fluid density, v - velocity, D - pipe diameter, and η - viscosity. This observation can be generalized. Note that any hypersurface in the cross product $X_1 \times \ldots \times X_n$ can be represented as

$$F(X_1, \ldots, X_n) = C, \tag{12}$$

where C is a constant (either dimensional or dimensionless). After transforming this to the dimensionless form we have a new equation for the hypersurface

$$f(\pi_1, \ldots, \pi_r) = \pi_c, \tag{13}$$

where π_c is a dimensionless constant. Note that this equation can be satisfied when the dimensionless variables are kept at some specific constant values $\pi_1 = C_1, \ldots, \pi_r = C_r$. This is a much easier problem to solve than to find a hypersurface as described by Equation 12, since we don't need to know the form of the function f (we just can experiment with the physical system and find for which values of the $\pi's$ we have $f() = \pi_c$s), and since we know the formulas for the $\pi's$. But this is not the most general solution. In general, this equation can also be satisfied by varying the $C_j's$. For this we would need to know the form of the function f. However, as was shown in many examples, this partial solution to finding critical hypersurfaces has proved to be useful.

6 Monitoring Dynamical Systems

One area for the applicability of critical hypersurfaces is control of time-varying systems (in control terminology the controlled system is called *plant*). This is quite a challenging goal. The main difficulty of controlling such plants is that we don't know when and under what circumstances the plant switches to a different behavior. Different behaviors are modeled by different models. If the controller uses the knowledge of the model, then it needs to monitor for such changes.

As an example of time-varying system consider a mass-spring system shown in Figure 1. Suppose the control goal is to keep the mass within some bounds of the height. Also assume the level of liquid in the container can change instantly (for instance, the container was damaged and thus the liquid was drained). It may be the case that the desired range of the height includes both configurations: when the mass is immersed in the liquid and when it is above the liquid. The plant can be characterized by two different models.

$$m \cdot \ddot{h}(t) + k \cdot h(t) = f(t) + m \cdot g \tag{14}$$

$$m \cdot \ddot{h}(t) + c \cdot \dot{h}(t) + k \cdot h(t) = f(t) + m \cdot g \tag{15}$$

where m - mass, t - time, h - height, k - spring constant, g - gravity. The first of the equations is for the case when the mass is not in the liquid (no damping) and the second one is for the mass immersed the liquid.

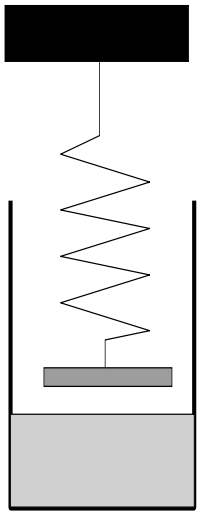

Figure 1: An Example of a Time-Varying System

If the equations for all possible behaviors are known, the designer of the controller can derive one control law for each behavior and, in addition to this, a switching policy so that the controller can monitor for the change of the behavior and switch to another control law whenever a switching policy rule is satisfied. However, not always do we have all such knowledge. In many situations we don't know all the different behaviors and their models. One of the possible approaches to deal with this kind of a problem is to develop a learning controller that can learn models of qualitatively different behaviors, control laws and switching policies. Results of experiments with this kind of approach can be found in [11, 9, 12, 14, 13, 19].

In this approach, we used memory-based learning, i.e., we stored state transition functions of dynamic plants as tuples consisting of *previous state* and *next state*, current input and elapsed time. The set of tuples was partitioned into subsets, one for each qualitatively different behavior. Note that we could not treat all of the tuples as one big set since in such a case the tuples would not represent a functional dependency (since for different behaviors we would have different next states for the same previous states).

The challenges with this approach were: (1) to be able to partition the space into a small number of qualitative behaviors, (2) to reduce the size of the stored database of tuples using similarity theory, (3) to be able to learn the behavior selection policy, (4) to be able to use the learned database for selecting behaviors and controlling the plant. The answers to all of these questions were positive. These results were presented in [11, 9, 12, 14, 13, 19].

7 Conclusions

The goal of this paper is to give an overview of various uses of similarity theory in the context of machine learning. We showed three of such uses. The first one was learning new relevant variables, i.e., variables that should be included in the model of a given physical phenomenon. The main leveraging in this process that was attributed to similarity theory was that we were able to discover the relevance of parameters that were not varied in the learning experiments. The second use of similarity theory was in learning functions that model physical phenomena. The advantage due the use of similarity theory in this case was the reduction of the degree of polynomials. This was achieved by fixing the degree of the polynomial and searching through various dimensional bases rather than jumping to a higher degree polynomial. The third use was in monitoring for qualitative changes of behaviors of physical systems. We described some experiments in which similarity theory was used to reduce the number of different models that need to be used for monitoring and controlling a dynamic system.

References

[1] S. Drobot. On the foundations of dimensional analysis. *Studia Mathematica*, 14:84–89, 1953.

[2] L. W. Johnson and R. D. Riess. *Numerical Analysis*. Addison Wesley, 1982.

[3] M. M. Kokar. Similarity in dimensional systems. *Systems Science*, 2:173–181, 1978.

[4] M. M. Kokar. The use of dimensional analysis for choosing parameters describing a physical phenomenon. *Bulletin de l'Academie Polonaise des Sciences: Serie des Sciences Techniques*, XXVII, No. 5/6:249–254, 1979.

[5] M. M. Kokar. A procedure of identification of laws in empirical sciences. *Systems Science*, 7/1:32–41, 1981.

[6] M. M. Kokar. Determining arguments of invariant functional descriptions. *Machine Learning*, 1:403–422, 1986.

[7] M. M. Kokar. Determining functional formulas through changing representation base. In *Proceedings of AAAI-86, Fifth National Conference on Artificial Intelligence*, pages 455–459. AAAI, 1986.

[8] M. M. Kokar. Critical hypersurfaces and the quantity space. In *Proceedings, AAAI-87, Sixth National Conference on Artificial Intelligence*, pages 616–620. AAAI, 1987.

[9] M. M. Kokar. Accumulating qualitative knowledge. In *Proceedings of the Fourth International Symposium on Intelligent Control*, pages 574–579. IEEE, 1989.

[10] M. M. Kokar. Semantic equivalence in concept discovery. In D. P. Benjamin, editor, *Change of Representation and Inductive Bias*, pages 309–325. Kluwer Academic Publishers, Boston-Dodrecht-London, 1989.

[11] M. M. Kokar, S. N. Keshav, S. Gopalraman, and Y. Lirov. Learning in semantic control. In *Proceedings of the 27-th Conference on Decision and Control*, pages 1812–1817. IEEE, 1988.

[12] M. M. Kokar and J. J. Reeves. Qualitative monitoring of time-variant physical systems. In *Proceedings of the 29-th Conference on Decision and Control*, pages 1504–1508. IEEE, 1990.

[13] M. M. Kokar and S. A. Reveliotis. Integrating qualitative and quantitative methods for model validation and monitoring. In *Proceedings of the 1991 IEEE International Symposium on Intelligent Control*, pages 286–291, 1991.

[14] M. M. Kokar and S. A. Reveliotis. Learning to select a model in a changing world. In *Proceedings of the 8-th International Workshop on Machine Learning*, pages 313–317, 1991.

[15] D. H. Krantz, R. D. Luce, and P. Suppes. *Foundations of Measurement*. Academic Press, 1971.

[16] R. D. Luce. Similar systems and dimensionally invariant laws. *Philosophy of Science*, 6:157–169, 1971.

[17] R. D. Luce. Dimensionally invariant physical laws correspond to meaningful relations. *Philosophy of Science*, 45:1–16, 1978.

[18] L. Narens. A general theory of ratio scalability with remarks about the measurement-theoretic concept of meaningfulness. *Theory and Decison*, 13:1–70, 1981.

[19] S. A. Reveliotis and M. M. Kokar. A framework for on-line learning of plant models and control policies for restructurable control. *IEEE Transactions on Systems, Man and Cybernetics*, 25, no. 11:1502–1512, 1995.

[20] F. S. Roberts. On the theory of meaningfulness in of ordinal comparisons in measurement. *Measurement*, 1, No.2:35–38, 1984.