

A Neural Network Approach to Treatment Optimization

Paul Munro, Ph.D. and Siripun Sanguansintukual

School of Information Sciences, University of Pittsburgh, Pittsburgh PA 15260

ABSTRACT

Typical medical diagnosis applications of neural networks for prediction and classification require training data (observations) that include the "correct" category for a number of patient records. In this paper, we borrow a technique from control systems applications of neural networks. Optimal control parameters of a system are typically not known. Instead, we only know the effect on a remote system. The correct control action drives the remote system optimally. The learning technique requires two networks: one to model the system to be controlled (here, the patient), and one to optimize the treatment (here, the treating physician). The concept was tested with artificially generated noisy data, and gives promising results.

INTRODUCTION

The task of medical treatment is to optimize a mapping from the space of patient states (patient measures, including biometric data and symptoms) to the space of treatments (here, dosages). Standard statistical approaches, including supervised learning with neural networks, require "training data" consisting of many input-output pairs (here, symptom-treatment pairs). However, in many cases, these kind of data are unavailable since the optimal treatment is not known; rather, the dosage is prescribed based on broad guidelines, the weight of the patient. In cases with multiple symptoms and multiple medications, prescriptive tools available to the physician are primitive at best.

An approach to "distal learning" (Williams, 1986; Munro, 1987; Jordan and Rumelhart, 1992) specifies a framework for training a network to perform an *action* (*A*) in response to a *situation* (*S*), where there are no training data indicating "correct" *S-A* pairs. Instead, it is assumed that "indirect" feedback data are available that express some *consequence* (*C*) of certain situation-action combinations. This approach has proven valuable in understanding motor control in humans and in artificial systems (Jordan, 1996).

THE MODEL

Two networks, a *patient model* (*PM*) and a *treatment network* (*TN*), are trained using backpropagation on artificial data that mimics hypothetical patient records (see Figure 1). Each record includes age, gender, an initial symptomatic measure (this could correspond to blood pressure, tumor size, etc.), dosage values of two medications, and a second measure (following treatment). Training occurs in two stages: First, the patient model is trained to predict the new symptom values from patient data (including initial symptoms) and treatment (dosages).

In the second stage, the weight parameters in the *TN* are trained to optimize the output of the patient model according to backpropagated gradients; while the treatment network is trained, the weights in the *PM* are held fixed.

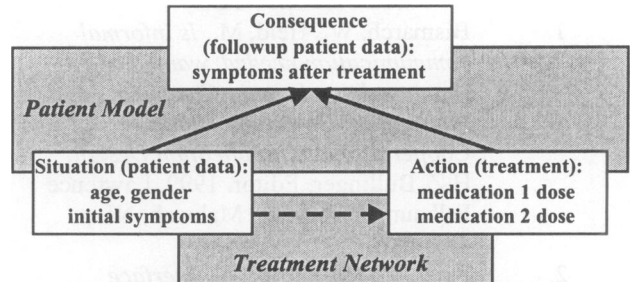


Figure 1. The Distal Learning System. Shaded rectangles represent networks and white rectangles represent fields from the observed data set. The Patient Model (*PM*) is trained to predict followup symptoms after treatment for patient records in the training set. The parameters in the *PM* are then held fixed (solid arrows) while the Treatment Network (*TN*) parameters (dashed arrow) are trained to optimize the predicted symptoms.

GENERATION OF ARTIFICIAL DATA

Each data set consisted of 6000 artificially generated patient records. An age value for data item α , y^α , is drawn from the normal distribution with mean 50 years and standard deviation 8 years. Gender (g^α) is coded by choosing 0 or 1 randomly with probability 0.5. Doses of the two medications (d_1 and d_2) were randomly drawn from the uniform distribution [0.2, 0.8]. It is assumed that the optimal dose of medication 1 is strictly a function of age ($d_1^{opt} = age/100$), and that the optimal value of medication 2 is strictly a function of gender: For females, $d_2^{opt} = 0.3$, and for males, $d_2^{opt} = 0.7$. The effect of the medications falls off according to a Gaussian function from the ideal points. This function is then multiplied by the initial symptom value s_i to give a followup value s_f (see equation below). A random number $\rho(n)$, uniformly drawn from the interval $[-n, n]$, is added to each target value.

$$s_f = s_i - s_i \exp \left[\frac{(d_1 - d_1^{opt})^2}{k_1} + \frac{(d_2 - d_2^{opt})^2}{k_2} \right] + \rho(a), \quad [1].$$

where k_1 and k_2 are constants.

It is assumed here that it is in the interest of patients to minimize s_f . The framework could easily be extended to multiple symptoms, where each one has a different optimal value. The data set included 6000 independently generated "patient records", of which N_{Tr} (= 5000) were used for training the *PM* (items above the heavy line in Table 1) and N_{Ts} (= 1000) were used to test performance (items below the line).

Table 1. Record format

Item	Initial State (random)	Treatment (random)	Followup Symptoms (computed)
1	y^1, g^1, s_i^1	(d_1^1, d_2^1)	s_f^1
⋮	⋮	⋮	⋮
α	$y^\alpha, g^\alpha, s_i^\alpha$	(d_1^α, d_2^α)	s_f^α
⋮	⋮	⋮	⋮
N_{Tr}	$y^{N_{Tr}}, g^{N_{Tr}}, s_i^{N_{Tr}}$	$(d_1^{N_{Tr}}, d_2^{N_{Tr}})$	$s_f^{N_{Tr}}$
1	y^1, g^1, s_i^1	(d_1^1, d_2^1)	s_f^1
⋮	⋮	⋮	⋮
β	$y^\beta, g^\beta, s_i^\beta$	(d_1^β, d_2^β)	s_f^β
⋮	⋮	⋮	⋮
N_{Ts}	$y^{N_{Ts}}, g^{N_{Ts}}, s_i^{N_{Ts}}$	$(d_1^{N_{Ts}}, d_2^{N_{Ts}})$	$s_f^{N_{Ts}}$

METHODS

The *PM* is a feed-forward network with 3 inputs (one initial symptom value and two dosage values), 6 hidden units, and a single output unit (followup symptom). It is trained using items from the training set to predict the followup symptom values in the presence of noise, and subsequently tested. The backpropagation training technique is used for the *PM*, with the s_f values as the teacher signal (targets). In the backpropagation procedure, the *PM* weights are iteratively modified by gradient descent to minimize the objective function

$$\sum_{\alpha}^{N_{Tr}} \left(s_f^{\alpha} - r_{PM}(s_i^{\alpha}, d_1^{\alpha}, d_2^{\alpha}) \right)^2, \quad [2]$$

where r_{PM} is the output of the *PM*.

The test data is used to assess the generalization performance of the *PM*; that is, the performance is

measured by the same objective function as measured over the test set.

The *TN* is a feed-forward network with one input (the initial symptom value), 6 hidden units, and two output units (doses). After training the *PM*, the *TN* is trained using a modified version of the backpropagation procedure to optimize the output of the *PM* according to the objective function

$$\sum_{\alpha}^{N_{Tr}} \left(s^* - r_{PM}(s_i^{\alpha}, r_{TN}(s_i^{\alpha})) \right)^2, \quad [3]$$

where s^* is the desired symptom value, and r_{TN} is the output of the *TN* (dosage values).

In training the *TN*, it is important to observe that the *PM* weights are held fixed. Note that if the *PM* weights were allowed to change, the network could produce optimal responses by simply ignoring the input and adjust the output biases such that the predicted symptom was optimal. By fixing the *PM* weights the *TN* is forced to compute dosage values that optimize the symptoms.

RESULTS

The performances of both the *PM* and *TN* are graphically displayed in Figures 2-6. Both networks performed well in terms of their average responses (as shown by linear fits) in response-target plots for each network.

The Patient Model

As illustrated in Figure 2, this network learns the task quite well. The target value is plotted against the target value from the data set for each item in the test set. Ideally, the points should lie on the line $x=y$. The regression line shows that the expected value of the output given a patient state is very close to the ideal line.

The target function computed by the network is illustrated for the space of possible doses in Figure 3, given an input corresponding to a 25-year-old female. The solid curves are contours of the target function, with the optimal value (*) at the center. The dashed curves are contour lines for the function computed by the *PM*. Notice that the contours are close to each other near the optimum. However, far from the minimum point, the *PM* contours deviate from the target function contours, and even exhibit local minima. Because of this, the *TN* is likely to converge to the wrong minimum if the initial weights are not carefully chosen. In every case we analyzed, the local minima were more distant from the origin than the "correct" minimum. Therefore, as long as the initial weights are sufficiently small, the network converges to the right value.

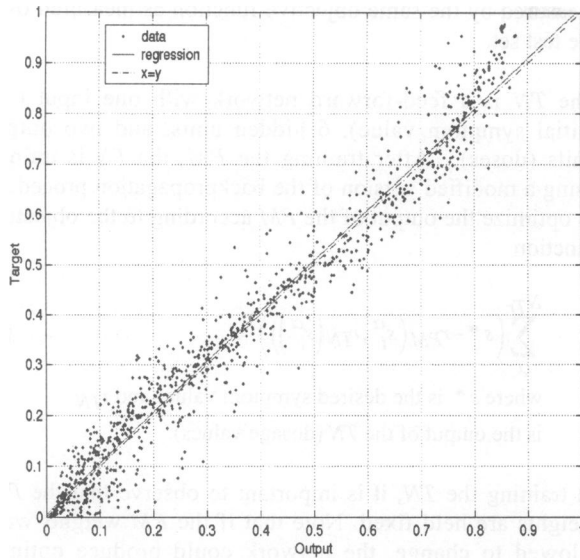


Figure 2. Target values are plotted vs. outputs computed by the *PM* after training for each item in the test set. The ideal line (dashed line) is plotted together with the best linear fit (solid).

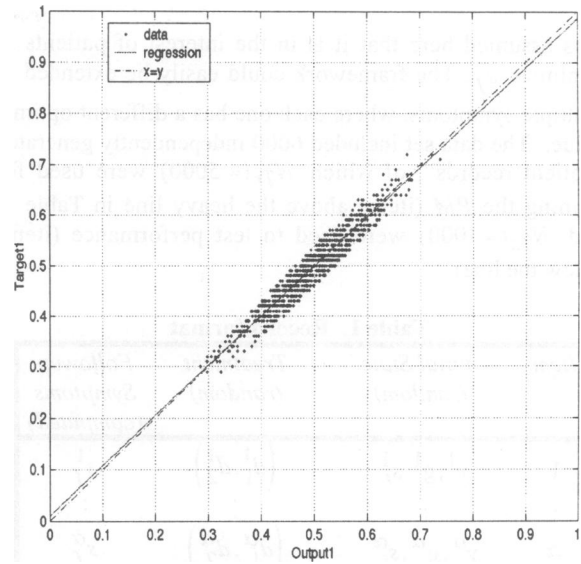


Figure 4. The performance of the *TN* is assessed by plotting Output 1 from the *TN* vs the ideal value (*age/100*) for the test items. As in Figure 3, the best linear fit (solid) is plotted with the ideal line (dashed).

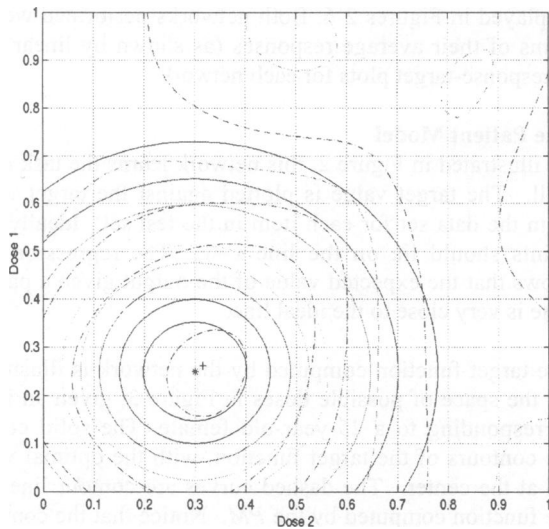


Figure 3. Contours for the noiseless function (Eq. 1, with $n=0$) is plotted as a function of the two dosages (solid curves) for a 25-year-old female. The asterisk shows the minimum of the function (optimal doses). Contours for the function computed by the *PM* are superimposed (dashed curves). The + indicates the dose values to which the *TN* converges.

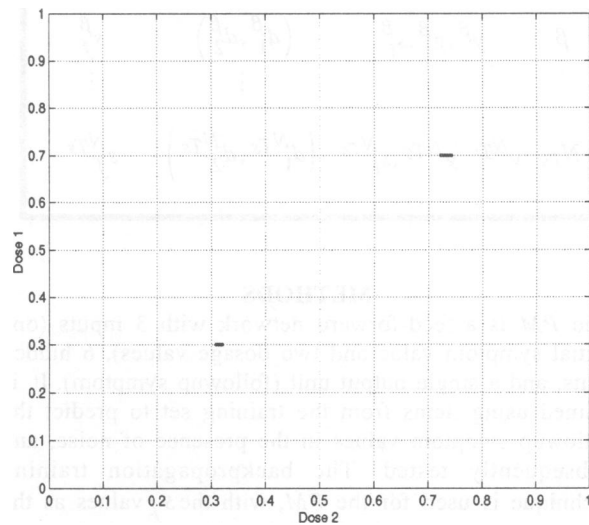


Figure 5. The performance of the *TN* is further assessed by plotting Output 2 from the *TN* vs the ideal value (0.3 for females, and 0.7 for males) for the test items.

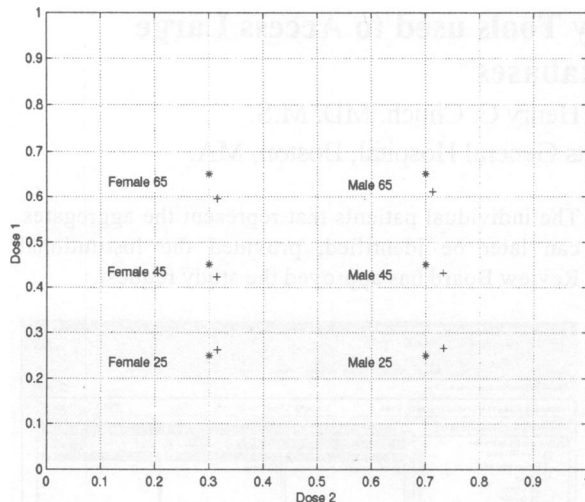


Figure 6. The performance of the *TN* is further assessed by plotting values for Dose 1 and Dose 2 computed by the *TN* (+) (as in Figure 3, but without the contours) for six age-gender combinations. The optimal doses (*) appear for comparison.

The Treatment Network

The *TN* is trained to find values for the treatment that optimize the symptom, by backpropagating deviations between the *PM* output and the optimal values. If the optimum doses for the *PM* network closely approximate the optimum doses for the final symptom function (Eq. 1), then the *TN* will compute doses that are close to the ideal.

The doses prescribed by the *TN* are plotted against the ideal doses for each item in the test set (Figures 4 and 5), showing a close correspondence. The recommended doses for a 30-year-old female woman are marked on the contour map of the final symptom function (Figure 3) with the '+' symbol. The *TN* performs a gradient descent computation of the optimal doses; here, the minimum of the *PM* function. The optimal points for the ideal function (*) and the doses prescribed by the *TN* are shown in Figure 6 for 6 types of patients: males and females of ages 25, 45, and 65. The proximity of the *TN* output to the ideal dosage values is a measure of the performance of the entire system (*PM* and *TN*).

DISCUSSION

This study provides a "proof of concept" for the distal learning approach as a method for treatment optimization. In principle, the system can be extended to account for more complex interactions between treatments, and for multiple symptoms. For example, side effects of medication could be treated as other symptoms that need to be optimized. The relative importance of primary symptomatic measures (tumor size) vs. secondary symptoms (side effects like nausea) can be adjusted by

weighting the terms in the objective functions (Eq. 2 and 3).

Previous studies of dose optimization algorithms have been primarily focused on finding optimal angles and intensities for radiation therapy against cancer (for example, Brahme, 2000; Wu and Zhu, 2001). Generally, these have been based on a biological model of the disease. The technique put forward in this paper builds a purely statistical model, which may be less appealing to patients and physicians. On the other hand, this approach to treatment optimization may prove itself a useful tool for physicians to consult when deciding on dose strengths.

This approach may prove valuable as a technique for providing patients with customized combinations of medications (so-called "cocktails") for treating any of several ailments, including asthma, ADHD, AIDS, various cancers, diabetes, and schizophrenia.

Of course, the ultimate test of this framework would be real data in which different dose combinations have been administered to patients. Both the symptoms and the treatments must be quantitative, and there is a need for at least one followup data point for each patient. Ideally, a clinical trial could be designed for the acquisition of the data.

The performance of the *PM* is enhanced by a uniform exploration of the dosage space. This would raise ethical issues if human subjects were used, since some of the medication levels and combinations may be dangerous. Thus, initial experiments with this technique may be best done on lower animals.

REFERENCES

- Brahme, A. (2000) Development of radiation therapy optimization. *Acta Oncologica* 39: 579-595.
- Jordan, M. & Rumelhart, D. (1992) Supervised learning with a distal teacher. *Cognitive Science* 16:307-354.
- Jordan, M. (1996) Computational aspects of motor control and motor learning. In H. Heuer S. Keele, (Eds.), *Handbook of Perception and Action: Motor Skills*. New York: Academic Press
- Munro, P. (1987) Dual backpropagation: a scheme for self-supervised learning. In: *Proceedings of the 9th Meeting of the Cognitive Science Society*. Princeton, NJ: Erlbaum.
- Williams, R. (1986) Inverting a connectionist network mapping by backpropagation of error.
- Wu X. and Zhu Y. (2001) An optimization method of importance factors and beam weights based on genetic algorithms for radiotherapy treatment planning. *Phys. Med. Biol.* 46:1085-1099.