An Artificial Intelligence System Suggests Arbitrariness of Death Penalty

Stamos T. Karamouzis* and Dee Wood Harper*

Abstract

The arguments against the death penalty in the United States have centered on due process and fairness. Since the death penalty is so rarely rendered and subsequently applied, it appears on the surface to be arbitrary. Considering the potential utility of determining whether or not a death row inmate is actually executed along with the promising behavior of Artificial Neural Networks (ANNs) as classifiers led us into the development, training, and testing of an ANN as a tool for predicting death penalty outcomes. For our ANN we reconstructed the profiles of 1,366 death row inmates by utilizing variables that are independent of the substantive characteristics of the crime for which they have been convicted. The ANN’s successful performance in predicting executions has serious implications concerning the fairness of the justice system.

1 Introduction

The death penalty has an ancient history but in the modern world the United States is the only western democracy that maintains it. Historically, there has always been some disparity between the authorization of executions and actual executions. The highest rate of execution in
AN ARTIFICIAL INTELLIGENCE SYSTEM SUGGESTS ARBITRARINESS OF DEATH PENALTY

the United States occurred in 1938 when there were about 2.01 executions per 100 homicides for the states with the death penalty. Even for capital murder the rate was less that 10 percent [1]. Between June 1967 and January 1977 no one was executed in the United States. In 1972 the U.S. Supreme Court in Furman v. Georgia found evidence of “arbitrary and discriminatory” sentencing that was in violation of the Eighth Amendment which prohibits “cruel and unusual punishment”. In Gregg v. Georgia (1976) the Supreme Court decided that if capital trials were restructured providing a sentencing phase with appropriate guidelines for jurors, death sentences could be applied fairly. The moratorium ended with the execution of Gary Gilmore in Utah by firing squad in 1977. The 900th post Gregg execution was carried out in the United States on March 3rd 2004.

Barbarity aside, the arguments against the death penalty in the United States have centered on due process and fairness. Since the death penalty is so rarely rendered and subsequently applied, it appears, prima facie, to be arbitrary. When the death sentence is rendered, poor and non-whites disproportionately receive it (There is also the issue of innocent persons being given an irreversible punishment) [2]. Our research focuses on what happens once a sentence is imposed. What are the characteristics of cases that will determine whether or not the defendant actually receives death?

Realizing the elusive task of identifying the variables that account for death penalty outcomes and ultimately predicting death penalty outcomes holds an enormous potential utility for specifying the post sentencing variables that account for the death or non-death outcome. Research evidence that further specifies the post death conviction process can assist in determining how fair or unfair the process is and, perhaps, can be used as an abolition argument. The task of prediction can be thought as partitioning prisoners under death sentence into two classes: the inmates whose death sentences were removed (non-executed) and the inmates who were executed.

Partitioning of a data set in classes is a very common problem in information processing. We find it in quality control, financial forecasting, laboratory research, targeted marketing, bankruptcy prediction, optical character recognition, etc. Artificial Neural Networks (ANNs) have been applied in these areas because they are excellent functional mappers (these problems can be formulated as finding a good input-output map) [3].

Considering the potential utility of predicting execution outcomes for prisoners under a sentence of death along with the promising behavior of multilayer perceptrons as classifiers led us into the investigation of ANNs as a tool for predicting death penalty outcomes. This article presents a test of the utility of ANNs and argues that the results pose a serious challenge to the fairness of the administration of the death penalty.
2 Methodology

In achieving our goal for predicting death penalty outcomes (i.e. determining whether or not a death row inmate is actually executed) we developed, trained, and tested an Artificial Neural Network (ANN) of the feed forward type, normally called multilayer perceptron. An ANN is a multiprocessor computing system that resembles the way biological nervous systems process information. The main characteristic of such a computing system is the number of highly interconnected processing elements (neurons) working together to solve specific problems without being programmed with step-by-step instructions. Instead ANNs are capable of learning on their own or by example through a learning process that involves adjustments to the connections that exist between the neurons.

2.1 Subjects (data)

The subjects (data) for the present study represented prisoners under a sentence of death during the 28-year period (1973-2000 inclusive) [4]. This data collection is available from the Interuniversity Consortium of Political and Social Research and is updated annually by the U.S. Department of Justice. Based on the following parameters a 19-parameter profile was created for each inmate.

1. Inmate identification number
2. State
3. Sex
4. Race
5. Hispanic origin
6. Year of birth
7. Third most serious capital offence
8. Second most serious capital offence
9. First most serious capital offence
10. Marital status at time of first imprisonment for capital offense
11. Highest year of education completed at time of first imprisonment for capital offense
12. Legal status at time of capital offense
13. Prior felony conviction(s)
14. Year of arrest for capital offense
15. Month of conviction for capital offense
16. Year of conviction for capital offense
17. Month of sentence for capital offense
18. Year of sentence for capital offense
19. Outcome (execution/non-execution)

In total 1,366 profiles were constructed. Half of them represented executed inmates and the other half non-executed. Randomly, 1,000 profiles from the
total population were used for training the neural network (training set), 66 for cross-validation, and the remaining 300 for testing (testing set).

2.2 Architecture

Given the computational capabilities of a multilayer perceptron as a universal pattern classifier a three-layered perceptron was developed. The first layer (input level) comprised of 17 neurons (processing elements) - one for each profile parameter minus the inmate identification and outcome parameters. The second layer (hidden level) comprised of 5 processing elements. The third layer (output level) comprised of 2 neurons - one for denoting execution and the other non-execution. Each neuron (processing element) is fully connected to every neuron in the following layer. Each neuron accumulates input from the neurons in the prior layer and provides output to neurons in the higher layer (figure 1).

2.3 Training

Considering that the desired responses of our system are known our perceptron was trained with error correction learning [5, 6]. Denoting $y_i(n)$ the system’s response at processing element $i$ at iteration $n$, and $d_i(n)$ the desired response then for a given input profile an instantaneous error $e_i(n)$ is defined by

$$e_i(n) = d_i(n) - y_i(n)$$

Based on the principle of gradient descent learning [7] each weight in the network is adapted by correcting the present value of the weight with a term that is proportional to the present input and error at the weight.

For updating the weights in our network we used an improvement to the straight gradient descent principle by using a memory term (the past increment to the weight).

Training was implemented using batch learning, i.e. first we presented all the patterns that describe the inmate profiles, then accumulated the weight updates, and at the end we updated the weights with the average weight update. The update of the weights after we present all patterns constitutes an epoch. Training took place over several epochs. To start the training we used small random values for each weight.

3 Results

After optimizing the network’s structure and training the network within 1000 epochs we tested the network’s predictive power on the training data set (i.e on the same 1,000 profiles used to train it). The mean square error achieved was 0.077 and the network was able to correctly classify 460/488 profiles of non-executed inmates and 448/512 profiles of executed inmates. Table 1 represents the network’s performance when tested with the training data.
When tested with the testing set (300 profiles) it produced a mean square error of 0.07 on non-executed and 0.07 on executed inmates. The network successfully classified 147 out of 158 non-executed inmates (93.0%) and 130 out of 142 executed inmates (91.5%).

4 Conclusion & Discussion

Having in mind importance of predicting death penalty outcomes and considering the classification power of ANN’s we turned into ANN technology for predicting death outcomes. In this article we presented the development, training, and testing of such a network. The network was developed as a three-layered perceptron and was trained using the backpropagation principles. For training and testing various experiments were executed. In these experiments, a sample of 1,366 profiles of death penalty convictions was used. The sample was divided into three sets. The first set of 1,000 profiles was used for training, 66 profiles for cross-validation, and the remaining 300 profiles were used for testing. The predictability rate for the training and test sets was higher than 90%. Comparatively, this is considerably better than reported results in similar domains such as predicting juvenile recidivism rates by employing artificial neural networks [8].

What we have demonstrated here is that ANN technology can predict death penalty outcomes at better that 90%. From a practical point of view this is impressive. However, given that the variables employed in the study, have no direct bearing on the judicial process raises serious questions concerning the fairness of the justice system.

Death penalty researchers believe that the most crucial variables for determining execution outcomes are whether or not DNA tests were conducted when relevant, and whether or not the defendant received competent representation [9]. Those variables are missing from our
data set because a) there is no available data on DNA testing at this
time and it probably would not be a factor in cases decided before the
test became available, and b) at this time we have no direct measure of
competent representation. Despite of not including those two crucial
variables our ANN yielded an impressive prediction rate solely based on
variables that are independent of the substantive characteristics of the
crimes.

In the future, we plan to expand the repertoire of variables that
describe the inmate profiles, include more profiles in the training set,
and employ sensitivity analysis techniques that will help us identify the
variables with the highest contributory weights in the predictive task. We
believe that this future work will not only help the network to achieve
even higher levels of predictability but will allow domain experts gain
new insights in determining how fair or unfair the process of death
sentencing is.

Table 1. Performance when tested with training data

<table>
<thead>
<tr>
<th>Performance</th>
<th>Non-Executed(1)</th>
<th>Executed(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE</td>
<td>0.077</td>
<td>0.077</td>
</tr>
<tr>
<td>NMSE</td>
<td>0.308</td>
<td>0.309</td>
</tr>
<tr>
<td>MAE</td>
<td>0.161</td>
<td>0.162</td>
</tr>
<tr>
<td>Min Abs Error</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Max Abs Error</td>
<td>0.984</td>
<td>0.975</td>
</tr>
<tr>
<td>r</td>
<td>0.831</td>
<td>0.831</td>
</tr>
<tr>
<td>Percent Correct</td>
<td>94.26</td>
<td>87.50</td>
</tr>
</tbody>
</table>

Table 2. Performance when tested with testing data

<table>
<thead>
<tr>
<th>Performance</th>
<th>Non-Executed(1)</th>
<th>Executed(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE</td>
<td>0.072</td>
<td>0.072</td>
</tr>
<tr>
<td>NMSE</td>
<td>0.291</td>
<td>0.292</td>
</tr>
<tr>
<td>MAE</td>
<td>0.161</td>
<td>0.162</td>
</tr>
<tr>
<td>Min Abs Error</td>
<td>0.001</td>
<td>0.000</td>
</tr>
<tr>
<td>Max Abs Error</td>
<td>1.051</td>
<td>1.050</td>
</tr>
<tr>
<td>r</td>
<td>0.843</td>
<td>0.843</td>
</tr>
<tr>
<td>Percent Correct</td>
<td>93.03</td>
<td>91.54</td>
</tr>
</tbody>
</table>

Where MSE is the mean square error, NMSE is the normalized mean square error, MAE is
the mean absolute error, and r is the correlation coefficient.

...
References