Training A Neural Network Through Iterative Optimization
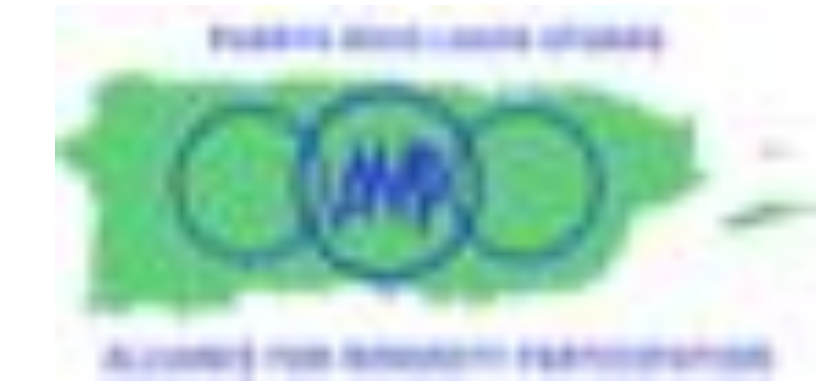
Sebastián Díaz Vera | Juan Ortiz Marrero | Dr. Mauricio Cabreras Ríos

University of Puerto Rico at Mayagüez | The Applied Optimization Group

Training a neural network implies finding its parameters to provide the best approximation quality to known patterns, while still being able to predict reliably when faced with unknown patterns. In other words, a neural network must have a balance between its capabilities to learn and to generalize. With such a problem this research group proposes applying a bicriteria iterative optimization strategy to manipulate the parameters of a simulation model to arrive at the best possible configurations in the presence of several performance measures in conflict and find the parameters in search of such balance.

Keywords: Data Simulation; Neural Network: Iterative Method; Optimization

Introduction

The use of Artificial Neural Networks is very useful to solve problems like a human brain does, basically they are a system of layers with interconnected nodes that works like neurons in our brain. With their connection system, the Neural Network has the ability to learn, generalize and improve, arriving with the best possible estimations. The underlying optimization objective is to create a universal approximator that can estimate whatever input is given, in order to solve multiple and complex problems in the world.

The purpose of this statistical research to improve an already existing optimization strategy in order to incorporate multiple variables, forming complex models that can successfully train and optimize our Neural Network.
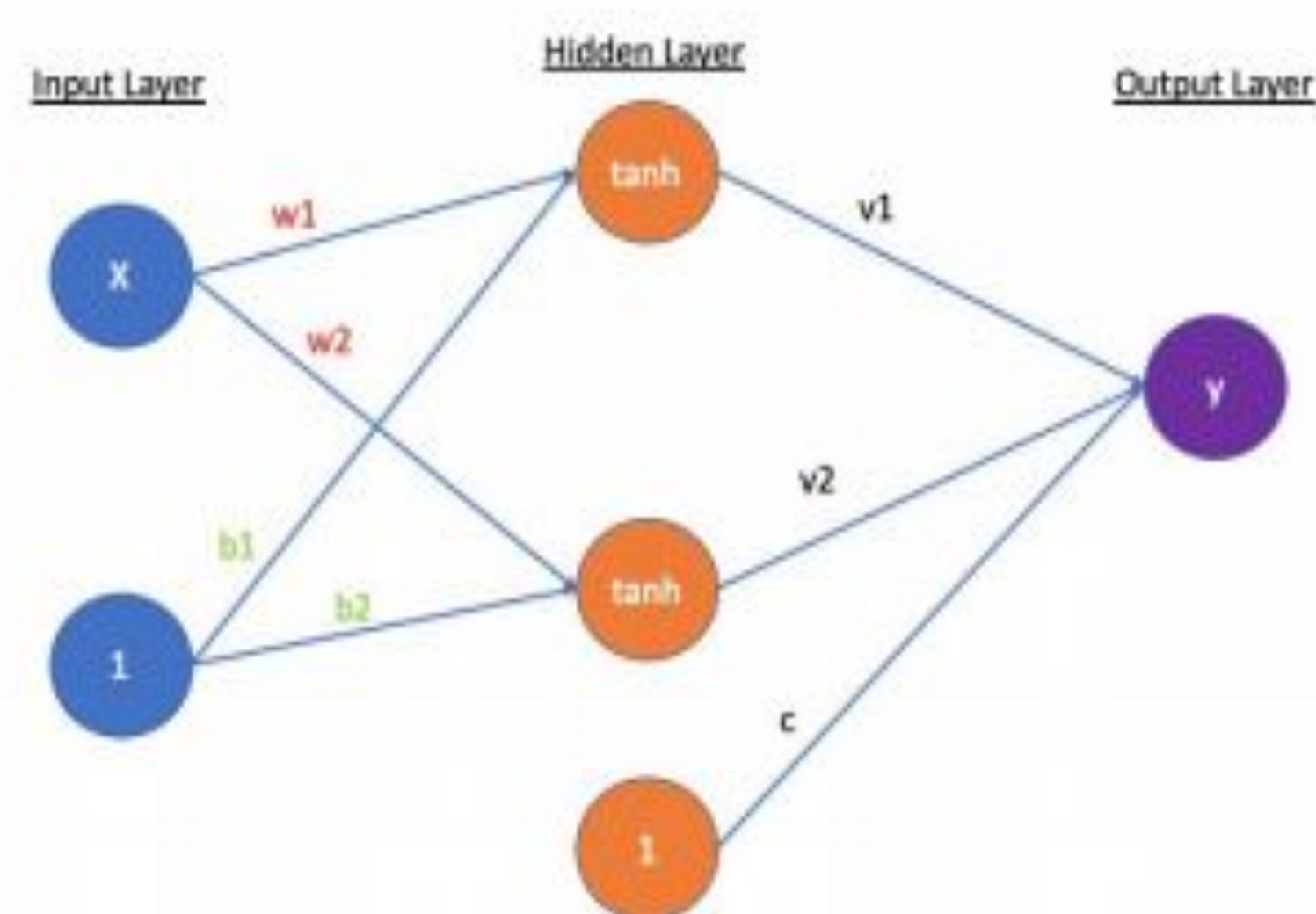

Figure 1: Artificial Neural Network with 7 variables.

Methods

The method starts with a regression model derived from a Neural Network of 2 hidden layers using a hyperbola parabola as an equation. The purpose is to measure the proportion of variance of a dependent variable assigning randomized numbers at each one of the variables of the regression model. From the variables of the hyperbola parabola equation, a design of experiment (DOE) will be created from which a simulation is performed at each design point, will be obtaining 142 points, and an initial incumbent solution is obtained. This incumbent solution is the best data point yet, and the Neural Network can be greatly improved, using this point. The next step is to generate metamodels originating from our initial DOE. In our case, these metamodels are fit regression models that are capable of making predictions. After this point, 100,000 aleatory combinations are performed using Minitab to predict SSE Training, SSE Validation, and identify the minimum value. This minimal value is going to be predicted in the neural network and compared with the first incumbent solution. Lastly, an algorithm will be simulated to identify if the new incumbent is best than the leading incumbent.

1. Initial Design of Experiment (DOE), D_0
2. Linear Transformation
3. Analyzing the responses in the Neural Network in order to find the first incumbent, I_0
4. Generate metamodels from the DOE points
5. Predicted Points: Generating aleatory runs
6. Pareto Analysis: Sorting the responses to find smallest values
7. Simulating all points, D_1
8. Compare the candidates with the current incumbent solution
9. Evaluate the stopping criteria
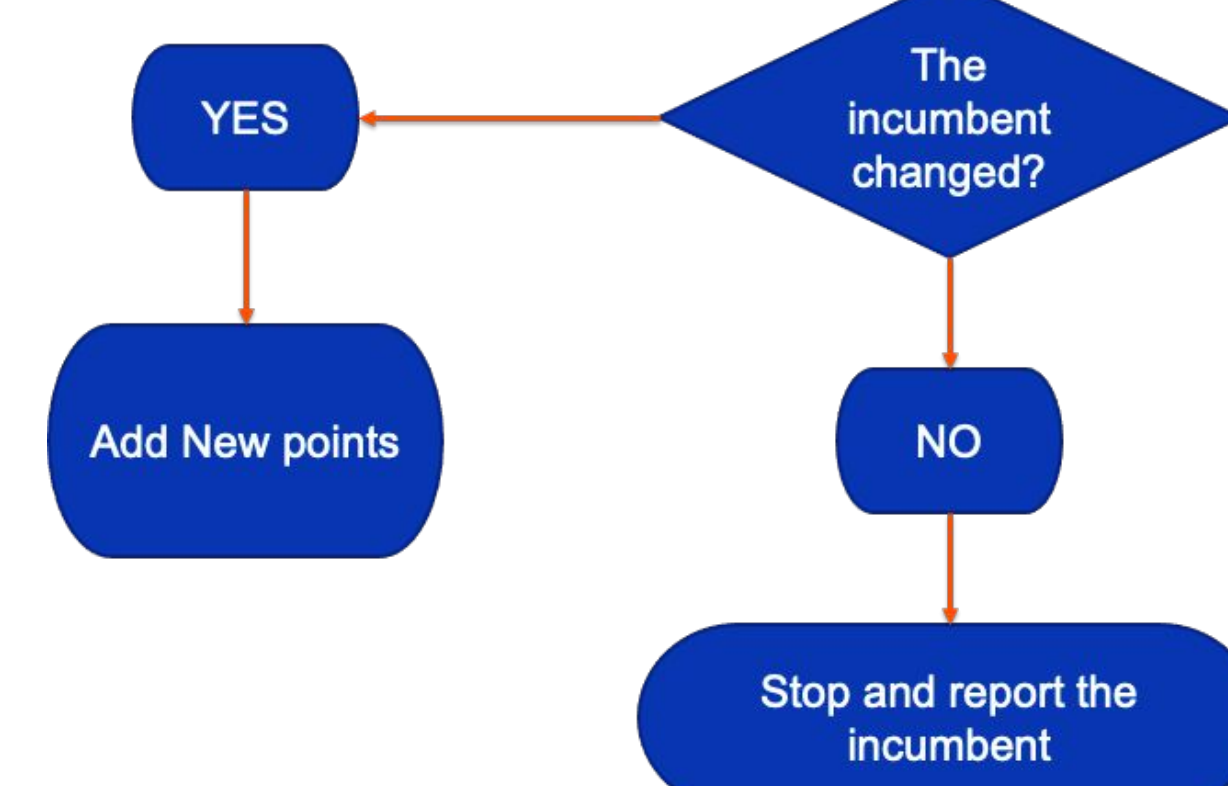10. If the incumbent changes, add points to DOE and repeat method starting from step 4

Figure 2: Stopping Criteria

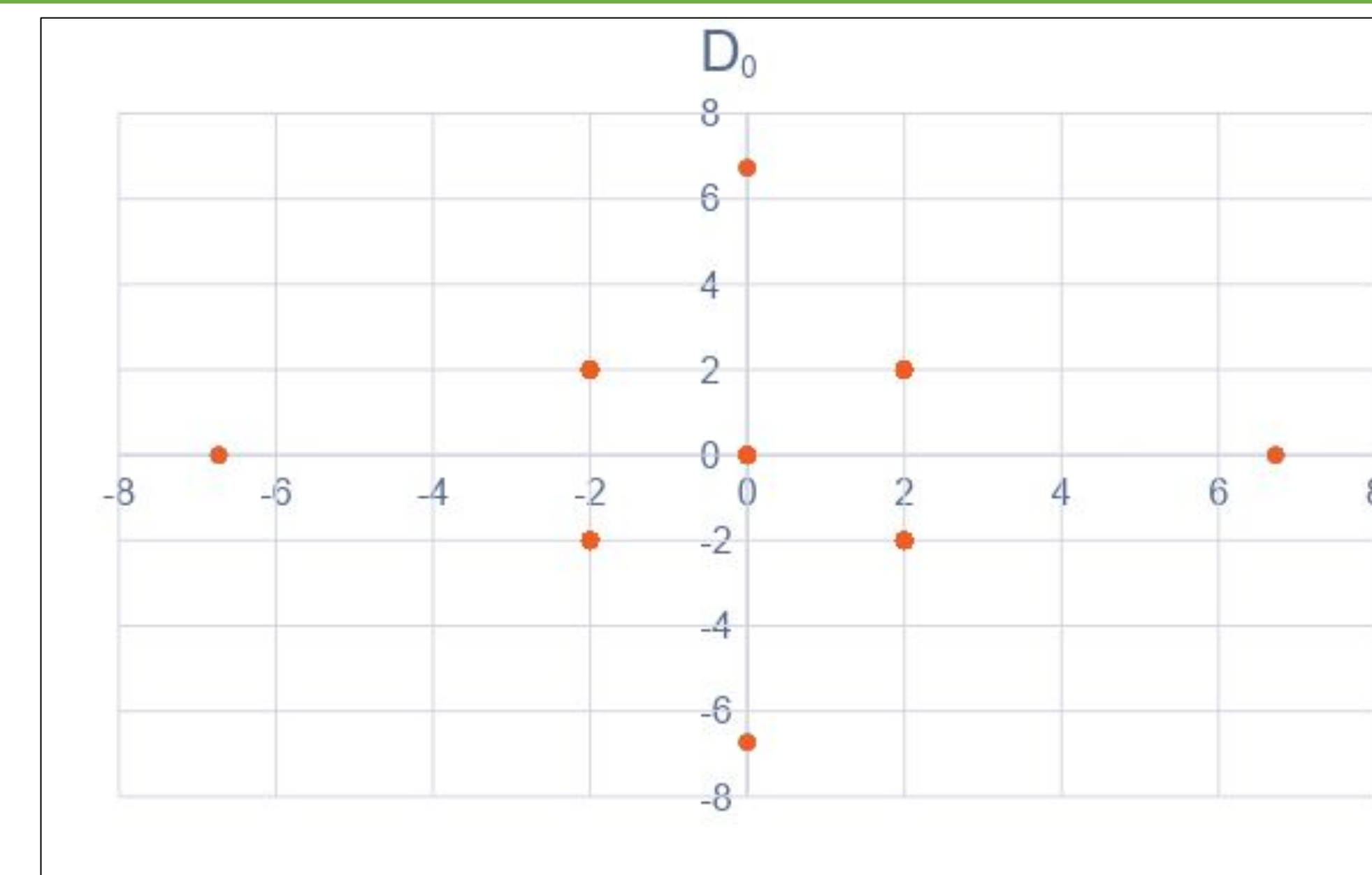## Results

Figure 3: Central Composite Design of 7 continuous factors ranging from -2 to 2.

	SSE Training	SSE Validation	R^2 Training	R^2 Validation
Incumbents, I_0	5.7080	2.9050	-17.95%	-25.39%

Figure 4: Initial incumbent solution.

Figure 5: Graph of our incumbent solution vs new candidate originating from our metamodel.

	SSE Training	SSE Validation
Incumbent, I_0	5.708	2.905
New Candidates	4.114	2.096

Figure 6: Comparison of SSE Training and SSE Validation values between the incumbent 0 and the new candidates.

	R2 Training	R2 Validation
Incumbent, I_0	-17.95%	-25.39%
New Candidates	15.00%	9.51%

Figure 7: Comparison of R^2 Training and R^2 Validation between the incumbent 0 and the new candidates.

Discussion

After studying our regression model and testing different variations, we created our first DOE (**Figure 3**). This design matrix was introduced into our regression model so that we could find our values with minimum error. We were able to find 12 incumbent solutions that had the same R^2 of -17.95% and -25.39%(**Figure 4**). This is our first iteration of the artificial neural network. After following the methodology we found a new incumbent solution with an R^2 of 15.0% and 9.51% (**Figure 7**). As we can analyze in **Figure 5**, after we plot the two incumbent solution we can identify the best solution as the new candidate.

Conclusion

This paper presents the training of a neural network through iterative optimization capable of making predictions with a R^2 of training of -17.9% and R^2 of Validation -25.39%. With our new incumbent solution the value of R^2 training and validation increased to 15% and 9.51%, respectively. In its current state, the neural network is functioning as expected. Even though our first results were positive, there is still room for improvement. We should develop better results as the research advances, understanding that we are still in the developmental phase. Taking these first steps and results into consideration, we can conclude that with this methodology of optimization we can obtain good results capable of solving common or complex problems that face humanity each day. Future work includes identifying new incumbent solutions with the purpose of adding them to the set of available points. The iteration method will be repeated until the criteria requires to stop.

Reference

Cabrera-Ríos, M., Niño-Perez, E. and Méndez-Vázquez, Y. 2015.

"A simulation-optimization strategy to deal simultaneously with tens of decision variables and multiple performance measures in manufacturing. "

Contact Information

Mauricio Cabrera Rios
- Mauricio.cabrera@gmail.com
- Mauricio.cabrera1@upr.edu
Juan Ortiz Marrero
- Juan.ortiz44@upr.edu

Sebastian Diaz
- Sebastian.diaz7@upr.edu
The applied Optimization Group
- Applied.optimization@gmail.com