Identifying the Relative Importance of Predictive Variables in Artificial Neural Networks Based on Data Produced through a Discrete Event Simulation of a Manufacturing Environment

R. Pires dos Santos[a]* and D. L. Dean[b] and J. M. Weaver[c] and Y. Hovanski[d] and

[a]*Manufacturing Engineering and Technology, Brigham Young University, Provo, United States; [b]Manufacturing Engineering and Technology, Brigham Young University, Provo, United States; [c]Manufacturing Engineering and Technology, Brigham Young University, Provo, United States; [d]Information Systems, Brigham Young University, Provo, United States.*

[a]*483 Belmont PL Unit 168, Provo, UT 84606. Phone: +1 (801) 228-8274. E-mail: rebecca.piress@hotmail.com

[d]786 N Eldon Tanner Building, Provo, UT 84604. Phone: +1 (801) 830-8677. E-mail: doug_dean@byu.edu

[c]265 Crabtree Technology Building, Provo, UT 84604. Phone: +1 (801) 422-1778. E-mail: Jason_Weaver@byu.edu

[d]265 Crabtree Technology Building, Provo, UT 84604. Phone: +1 (801) 422-7858. E-mail: Yuri.hovanski@byu.edu

Rebecca Pires dos Santos

Rebecca Santos received her Master of Science in Technology with an emphasis in Manufacturing Engineering from Brigham Young University and her Bachelor of Science in Industrial Engineering from Universidade Federal de Pernambuco located in the state of Pernambuco, Brazil. She is interested in studying the application of data science in a manufacturing environment. Contact information: 483 Belmont PL Unit 168, Provo, UT 84606. Phone: +1 (801) 228-8274. E-mail: rebecca.piress@hotmail.com

Douglas L. Dean

Douglas Dean is an Associate Professor of IS at Brigham Young University, Utah. He received his PhD in MIS from the University of Arizona. He has published articles in Management Science, MIS Quarterly, Journal of MIS, Journal of the AIS, IEEE Transactions, Electronic Commerce Research, Expert Systems with Applications, and others. His research interests include knowledge sharing, data mining methods, scientometrics, and collaborative tools and methods. Contact information: 786 N Eldon Tanner Building, Provo, UT 84604. Phone: +1 (801) 830-8677. E-mail: doug_dean@byu.edu

Jason Michael Weaver

Jason Weaver is an Assistant Professor of Manufacturing Engineering at the Ira A. Fulton College of Engineering & Technology at Brigham Young University. He is a Systems Engineer with broad experience in mechanical engineering, nuclear weapon safety, product design, and reverse engineering. His research involves Design for Manufacturing, Additive Manufacturing, Systems Engineering, and Design Theory and Quality. Contact information: 265 Crabtree Technology Building, Provo, UT 84604. Phone: +1 (801) 422-1778. E-mail: Jason_Weaver@byu.edu

Yuri Hovanski

Yuri Hovanski is an Associate Professor of Manufacturing Engineering at the Ira A. Fulton College of Engineering & Technology at Brigham Young University. He is a specialist in Friction Stir Technologies. Contact information: 265 Crabtree Technology Building, Provo, UT 84604. Phone: +1 (801) 422-7858. E-mail: Yuri.hovanski@byu.edu

Identifying the Relative Importance of Predictive Variables in Artificial Neural Networks Based on Data Produced through a Discrete Event Simulation of a Manufacturing Environment

This research used a discrete event simulation to create data on a shipment receiving process instead of using historical records on the process. The simulation was used to created records with different inputs and operating conditions and the resulting overall elapsed time for the overall process. The resulting records were used to create a set of predictive artificial neural network (ANN) models that predicted elapsed time based on the process characteristics. Then the connection weight approach was used to determine the relative importance of the input variables. The connection weight approach was applied in three different steps: 1) on all input variables to identify predictive and nonpredictive inputs, 2) on all predictive inputs, and 3) after removal of a dominating predictive input. This produced a clearer picture of the relative importance of input variables on the outcome variable than applying the connection weight approach once.

Keywords: discrete event simulation; artificial neural networks; connection weight approach; data mining.

Introduction

Predictive analytics methods and discrete event simulation (DES) are two important methods that can provide important insights and find hidden patterns in data. Predictive analytics and DES have different, but complementary, aims.

Predictive analytics is an area of study that develops methods to predict outcomes from information that has predictive value. Predictive analytics encompasses a variety of statistical techniques from data mining, predictive modelling, and machine learning that analyze current and historical facts to make predictions about future or otherwise unknown events [1].

Predictive analytics methods have been created to find predictive relationships between known sets of predictor variables and known outcomes. These methods can be applied to dynamic processes if valid information on operating conditions and the corresponding results are available. However, collecting detailed operating histories that contain operating conditions and outcomes can be time consuming and expensive. When historical process data are not available, a simulation model can be created to represent the process and generate data in a faster and less costly manner than recording an operating history.

Another benefit of DES models is that they can be used to capture valid information about dynamic processes like shipping or manufacturing. Such processes are complex because there are many possible inputs and operating conditions that determine the outcome of the overall process. DES models are created to represent the relationships among discrete events that make up an overall system, such as a factory or a production line. DES models represent dynamic systems that are comprised of a sequence of related, interdependent events.

DES tools can be used to generate a variety of operating scenarios, where each scenario includes specific input variables and other operating conditions. As each scenario is run through the DES model, the model determines the corresponding output value, such as the total time to complete the overall process. Each scenario can be simulated one or many times. In this way, a variety of operating conditions can be simulated, and the related outcomes can be recorded. The output of the DES process is a set of records where each record contains the inputs' values and the corresponding output value. This data can then be used in the predictive modeling process.

In this study, we created a simulation model based on the shipment receiving process. We then used DES to generate data on a raw material receiving process. We then evaluated multiple different predictive modeling algorithms, including KNN, gradient boosting, artificial neural networks (ANNs), and several others to determine which provides the most accurate predictions of overall elapsed time. We found that ANNs produced the most accurate predictions. We then used the connection-weight approach to determine the relative importance of the input variables in relation to overall elapsed time.

This study aims to confirm the following hypotheses:

(1) The connection weight approach applied to ANNs can be used to rank independent variables of a DES model according to their importance.

(2) Manipulation of the most important variables ranked by the connection weight approach in a DES model can lead to improvement in business performance.

Literature Review

DES Coupled with Data Mining

Recent literature has expanded on different applications of DES, such as soybean transportation analysis [2], modeling human behavior [3], predicting crowding scores in an emergency department [4], and many others [5, 6]. In these complex systems, it becomes difficult to make predictions, analyze current states, and propose improvements without the use of analytical tools such as DES.

However, there are cases where DES alone is not enough. Sometimes it is beneficial to use other tools to inform simulation model inputs or extract information from a simulation model for analysis in other tools. Accordingly, researchers have developed new approaches combining DES with other tools, such as data mining algorithms. The current research aims to build upon the ideas created by Better, Glover [7] and Brady and Yellig [8]. Brady and Yellig [8] used DES to generate data that could be evaluated by simulation optimization algorithms to assess the importance of inputs in relation to the overall simulation results. Better, Glover [7] extended this work by suggesting that DES should generate data as an input to data mining tools, which could be used to determine relevant input attributes and rules that could be used to improve the simulation results. Some examples where data mining has been coupled with DES include decision trees used to support DES output analysis [9], determination of association rules among DES parameters [10], classification rules used to dynamically to optimize a DES model [7], and correlation scores used to determine relationships between simulation constructs in order to develop simulation optimization scenarios [8].

ANN Interpreted by the Connection Weight Approach

Neural networks often perform well when compared to other machine-learning based predictive algorithms. The ANN algorithm has a number of advantages. The algorithm is able to learn from linear and non-linear relationships in the data [11, 12]. It can also measure and incorporate both direct effects and interaction effects among variables into predictive models [13]. Historically, the multilayer feed-forward perceptron (MLP) with backward propagation is the most widely used NN typology [14]. One reason for MLPs' success is that several research groups [15, 16] have mathematically demonstrated that an MLP NN with a single hidden layer is a universal function approximator. Plus, some research has shown that neural networks can still perform quite well when MLP ANNs are created with different numbers of neurons in the single hidden layer [17].

However, ANN models can be complex, and it can be difficult to determine the relative impact of input variables for a number of reasons. First, when ANNs are created, random values are used to initialize the beginning values for the weights on the links between neurons. Training then continues from the point. Thus, it is common for the relative importance of inputs to differ considerably across models [17]. Also, different methods of determining the relative importance give somewhat different results [18].

Researchers have developed different methods to help with this problem. Some methods created with this purpose in mind include Garson's algorithm [19, 20], connection weight approach [21], partial derivatives [22, 23], input perturbation [24], sensitivity analysis [25, 26], and others [18, 21, 27]. The present research uses the connection weight approach for three reasons: Olden and Jackson [21] demonstrated it is more accurate than the Garson's algorithm [15], it usually performs as well or better than other approaches [17, 18], and because it is derived directly from the weights of the links in the neural network.

In the connection weight approach, the relative contributions of the input variables on the output of the ANN model is based on the weights on the links from input neurons to hidden neurons and from hidden neurons to the output neurons. The steps created by Olden and Jackson [21] are as follows:

(1) Create multiple ANNs using the original data with different initial random weights. Select the neural network with the best predictive performance as measured by R^2 and RMSE. For example, create 20 ANN models and select the most predictive one.

(2) Record the connection weights from the links between the neurons:

   (a) for links from input nodes to hidden neurons.

   (b) for links from hidden neurons to the output node.

(3) Calculate the product of the input-to-hidden-neuron weight and the hidden neuron-output weight for each input-to-output connection. Calculate the importance score for each input, which is the sum of all contributions to the output node made by a given input through all hidden nodes. Figure 1 shows an example of these calculations.

(4) Go back to step one and repeat the process until enough ANNs have been evaluated to allow for a reasonable distribution of importance scores. Following the example of Olden, Joy [18] and De Oña and Garrido [17], we found that calculating the relative weight of the inputs on 50 "best" models (chosen in step 1) was sufficient to get an adequate distribution of results.

(5) Summarize the relative importance score for the input variables across the multiple models.

Input-Hidden Connection Weights

Y / X	Hidden A	Hidden B	Hidden C
Input 1	0.97	0.82	1.56
Input 2	-0.52	-0.91	0.48
Input 3	-0.69	0.84	1.28
Input 4	1.02	0.24	-0.37

X

Hidden-Output Connection Weights

	Hidden A	Hidden B	Hidden C
Output	1.31	-0.39	0.39

Connection Weight Products Sum to Importance

Y / X	Hidden A	Hidden B	Hidden C	∑ = Importance
Input 1	1.27	-0.32	0.61	1.56
Input 2	-0.68	0.35	0.19	-0.14
Input 3	-0.90	-0.33	0.50	-0.73
Input 4	1.34	-0.09	-0.14	1.10

$$Normalized = \frac{x_i - \min(x)}{\max(x) - \min(x)}$$

$$\text{e.g., } Normalized_{Input4} = \frac{1.10-(-0.14)}{1.56-(-0.14)} = 0.80$$

Normalized Importance Scores

	Importance	Normalized	Rank
Input 1	1.56	1.00	1
Input 2	-0.14	0.26	2
Input 3	-0.73	0.00	3
Input 4	1.10	0.80	4

Figure 1. Connection weight calculation

Methodology

In order to test the hypotheses of this research, we performed an experiment that included the following five steps: 1) create and validate a simulation model; 2) using the data from the simulation model, create predictive models using several types of predictive algorithms to determine which type of algorithm produces the best predictions (we found that ANNs performed the best out of all the algorithms we tested); 3) create 50 high-performing ANN models; 4) calculate the relative importance of each input variable for each model using the connection weight approach; and 5) summarize the results.

Case Study Description

This study was based on a real problem faced by a manufacturer located in Brazil. The company wanted to be more efficient in their raw materials receiving process. Currently they face fluctuations in the arrival of trucks delivering raw materials, which causes either long lines of trucks waiting to unload or a shortage of raw materials. They want to be able to better predict the total time a truck stays in the system and identify the main factors that impact the total time.

The company's raw material receiving process description is as follows. First, the truck arrives at the entrance location where paperwork is done. Then the truck waits for its turn to have its sample collected at the mill hopper location. After it is collected, the sample goes to the laboratory where it will be analyzed, and the truck awaits the analysis results. When the analyses are finished, if the raw material is accepted, the truck will wait for its turn to unload its material at the mill hopper location. After unloading, the truck is free to go. If the material is rejected, the truck is not allowed to unload.

All variables used to build ANNs are listed below:

- *IsGroupA*, *IsGroupB*, and *IsGroupC*: Dummy variables representing different groups of material that arrive in the manufacturing.
- *IncludesWeekend*: A binary variable that indicates whether or not the truck had to wait to unload during the weekend.
- *IncludesNight*: A binary variable that indicates whether or not the truck stayed overnight in order to be unloaded.
- *Shortage*: A binary variable with a value of one when a shortage caused the manufacturing process to stop during the time the truck was in line.
- *UnloadQuantity*: This variable represents the weight, in thousand pounds, of the material in the truck.
- *WaitedToUnload*: A binary variable that indicates whether or not the quantity loaded in the truck exceeds the silo's free capacity at the time the truck arrives.
- *WasPriority*: Binary variable that indicates whether there is currently a shortage of a material being carried by the truck. If so, the truck gets priority in the queue.
- *TimeEntrance*: Time to complete the paper work at the entrance.
- *TimeAnalysis*: Time taken by the lab to analyze a sample of the material in the truck.
- *TimeCollection*: Time taken at the mill hopper to collect a sample of the truck material.
- *TimeUnload*: Time taken at the mill hopper to unload a truck.
- *TrucksInLine*: This variable represents the number of trucks waiting to unload their material at the moment a specific truck arrives.
- *TotalTime*: This is the response variable. It measures the total time the truck stayed in the system.

Experiment

First, a simulation model of the system studied was created using ProModel® software. The model was then verified and validated to ensure that it was a correct representation of the real system. Then, the simulation was conducted to calculate total elapsed time based on different inputs and process conditions. 4934 records were created by the simulation. Next, the data generated by the simulation was used to create the ANNs.

We varied the number of hidden neurons in a single hidden layer from three to twelve. We tested a combination of TanH and linear transfer functions and found that using all TanH transfer functions performed the best. For each number of neurons, we created twenty ANN models and had the data mining software pick the most predictive one. We also tried a number of configurations with two hidden layers. We found that using a single layer with twelve neurons produced the best predictive results.

Fifty high-performing ANN models were created so that we could summarize the results to account for the variability injected into the ANN creation process. The software uses the back-propagation algorithm with one hidden layer.
When the overall input contributions or importance scores were calculated, it was observed that some values were positive while others were negative.

In order to make comparison between variables possible, absolute values for the importance scores were calculated in this research. It is important to note that the results shown in this research will not indicate whether a variable will increase or decrease the output value, but rather, how important the variable is to the dependent variable that will be predicted.

In the connection weight approach, after the importance scores are calculated, they are given an ordinal number as their rank. Through this research, it was possible to observe that some variables have importance scores that are very similar, thus making them difficult to differentiate. Consequently, when ranks are ordinal they determine whether one variable is more meaningful than the other, but they do not specify by how much. The present research used a normalized rank instead of an ordinal rank. This was done by normalizing the importance scores and using this normalized number as their rank, as shown in Figure 1. This made it possible to not only define an ordinal rank of input variables, but also to determine how much they differ in proportion to the input variable that has the highest relative importance.

Results

The dataset used in this research consisted of 4934 records. We used a 60:40 ratio for training and test partitions. The results shown are based on an average of fifty ANN models.

ANN Compared to Other Algorithms

Before ANN was picked as the algorithm used in this research, it was compared to other data mining algorithms in order to see which had the best prediction capabilities based on the R^2 and RMSE results. This comparison is shown in Table 1. For the current dataset produced by the simulation model, the ANN algorithm had the best prediction performance, having the highest R^2 and lowest RMSE in the test dataset.

	Training Dataset			Test Dataset		
	MAPE	RMSE	R^2	MAPE	RMSE	R^2
Linear Regression	84.5	1174	0.700	88.0	1158	0.640
Random Forest	37.6	871	0.835	43.6	974	0.745
KNN (Equal weights)	36.8	960	0.799	40	1011	0.725
Gradient Boosting	42.6	892	0.827	48.1	1010	0.726
Regression Trees	29.2	805	0.859	39.3	1113	0.667
Artificial Neural Network	35.6	869	0.835	40.1	930	0.770

Table 1. Data Mining Algorithms Prediction Results

Connection Weight Approach

To determine how well the connection weight approach could differentiate input variables that did or did not contain useful predictive information, we conducted a preliminary analysis to determine if each input variable contained predictive information. To determine this, we created models that included all input variables. Then we removed one input variable at a time to see if the predictive power of the models diminished, as reflected by lowered R-square values and higher RMSE values. Out of the 14 input variables, the following five could be removed without reducing predictive quality: TimeAnalysis, TimeCollection, UnloadQuantity, TimeEntrance, and TimeUnload.

We then did a sequence of three rounds of tests using different combinations of input variables. In each round, we created 50 high-performing ANN models. To find each high-performing model, we created 20 ANN models and selected the most predictive one as measured by R-squared and RMSE. Thus, for each round, we created 1000 ANN models and selected the 50 most predictive ANNs. We then used the connection weight approach to determine the relative importance of the input variables.

Round 1: Relative Importance when All Input Variables are Included

In the first round of testing, we included all 14 input variables, including nine that contained predictive information and five that did not. Table 2 and Figure 2 show the results of the first round of testing. Table 2 shows the relative importance of all input variables.

Input Variable	Mean Normalized Relative Importance	Contains Predictive Information	Correlation (r) with Outcome Variable
TrucksInLine	1.00	Yes	0.313
IsGroupA	0.49	Yes	-0.108
IsGroupB	0.46	Yes	-0.253
IncludesNight	0.44	Yes	0.520
WaitedToUnload	0.43	Yes	0.607
Shortage	0.25	Yes	0.377
IncludesWeekend	0.23	Yes	0.372
TimeAnalysis	0.21	No	0.062
WasPriority	0.20	Yes	-0.240
IsGroupC	0.18	Yes	0.374
TimeCollection	0.15	No	-0.030
UnloadQuantity	0.14	No	-0.130
TimeEntrance	0.04	No	-0.006
TimeUnload	0.02	No	-0.076

Table 2. Relative Importance, Predictive Variables, and Correlation with Outcome

The four input variables with the lowest relative importance also lacked predictive information. The five input variables with the lowest Pearson correlations, r, with the outcome variable, were also the five that lacked predictive information. Thus, the combination of a low relative importance score and the lowest correlation coefficient corresponded to all five non-predictive input variables. Figure 2 contains a bar chart that reflects the average normalized relative importance score for each input variable and a

Tukey's boxplot to reflect the distributions of the normalized relative importance score for each input variable. The thick lines in the box represent the median, and the thin red lines represent the average.

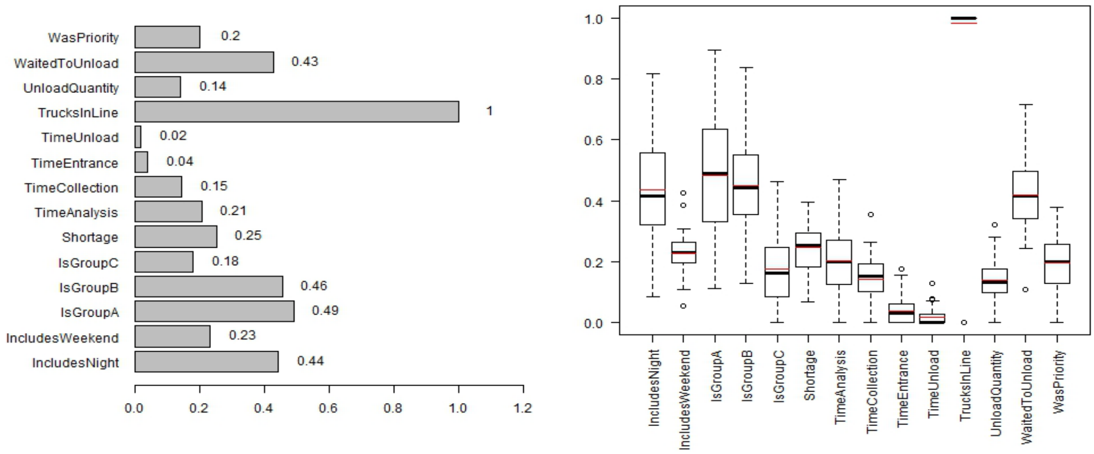


Figure 2. Average Importance Scores Including All Variables and Box Plot of Scores

The variable TrucksInLine had the highest importance score in all models. Thus, its score is represented by one, which is the highest normalized score possible. The next most important variables are IsGroupA, IsGroupB, IncludesNight, and WaitedToUnload. However, they have very similar average scores, making it hard to define which variables are actually the most meaningful to the model.

Round 2: Relative Importance when Only Predictive Variables are Included

Because nonpredictive input variables can cause noise that may confuse the relative importance scores, in the second round, we included only the nine variables that contain predictive information to determine what effect excluding nonpredictive variables would have on the relative importance scores of the remaining input variables. The results are shown in Figure 3.

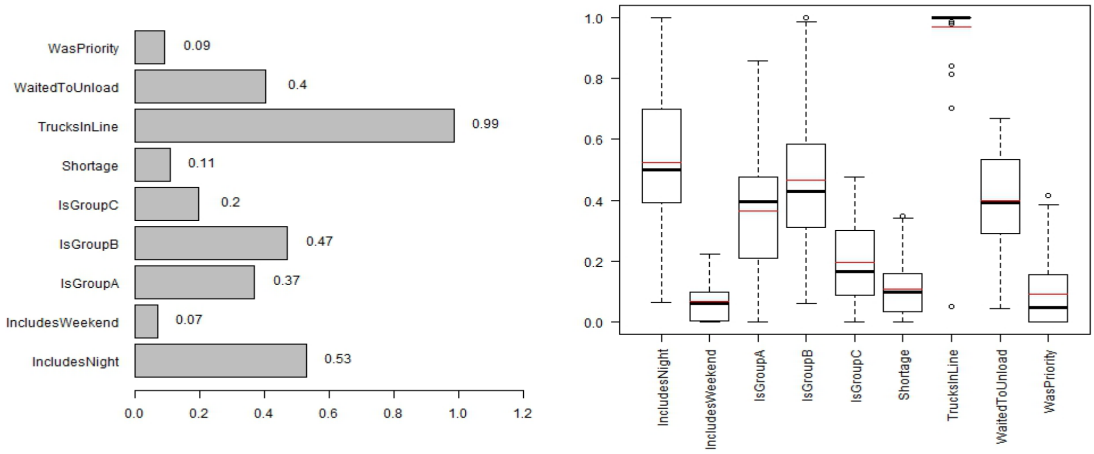


Figure 3. Average Importance Scores Including Meaningful Variables and Box Plot of Scores

As shown in Figure 3, the variable TrucksInLine is still the most important variable in predicting the outcome of the system, followed by IncludesNight, IsGroupB, WaitedToUnload, and IsGroupA. The new scores follow a different order compared to the previous rank. Thus, the removal of variables that did not contribute predictive information to the model clarified the relative importance of the remaining variables.

Variables IncludesNight, IsGroupB, WaitedToUnload, and IsGroupA still have similar scores in the second test. However, the model is more sensitive to existing differences, as evidenced by the fact that the differences in the relative scores are higher than in the previous test.

Table 3 provides a comparison between relative importance scores for the first and second rounds. The average importance scores for the variables TrucksInLine, IsGroupB, IsGroupC, and WaitedToUnload are very similar. There are some differences in other variables' scores, the highest being IncludesWeekend with a score of 0.16.

	Round 1: All Input Variables		Round 2: Just Predictive Variables		Difference
Input Variable	Relative Importance Mean (S.D)	Raw Rank	Relative Importance Mean (S.D)	Raw Rank	Relative Importance Mean (S.D)
TrucksInLine	1.00 (0.00)	1	0.99 (0.05)	1	0.01 (0.05)
IsGroupA	0.49 (0.18)	2	0.37 (0.21)	5	0.12 (0.03)
IsGroupB	0.46 (0.15)	3	0.47 (0.24)	3	0.02 (0.09)
IncludesNight	0.44 (0.16)	4	0.53 (0.22)	2	0.09 (0.06)
WaitedToUnload	0.43 (0.11)	5	0.40 (0.15)	4	0.02 (0.04)
Shortage	0.25 (0.07)	6	0.11 (0.09)	7	0.14 (0.02)
IncludesWeekend	0.23 (0.06)	7	0.07 (0.07)	9	0.16 (0.01)
WasPriority	0.20 (0.09)	8	0.09 (0.12)	8	0.11 (0.03)
IsGroupC	0.18 (0.12)	9	0.20 (0.15)	6	0.02 (0.03)

Table 3. Difference in Relative Importance in Rounds 1 and 2

Round 3: Relative Importance when the Most Dominant Variable is Excluded

In the previous two rounds, the variable TrucksInLine was by far the input variable with the highest relative importance. In this round, we removed the TrucksInLine variable to get a clearer picture of the relative value of the remaining input variables.

The first and second tests did not offer many insights on how to rank the variables with very similar scores. In order to better understand how to make a distinction between the variables IncludesNight, IsGroupB, WaitedToUnload, and IsGroupA, a third test was performed. In this test we excluded the variable TrucksInLine from the ANNs. As TrucksInLine was the most influential input variable, it may have dominated the models such that other variables could not differentiate themselves from others with similar scores. The results from the third test are shown in Figure 4.

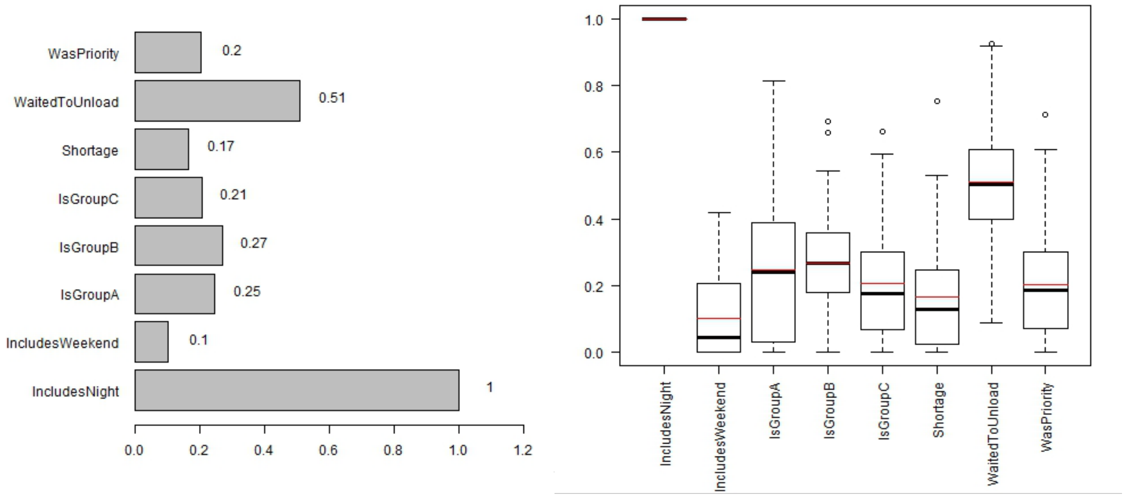


Figure 4. Average Importance Scores Excluding TrucksInLine from Second Test and Box Plot of Scores

With the removal of the dominant predictor, the next best predictor IncludesNight, followed by WaitedToUnload, shows more differentiation from the variables with similar relative importance scores in Round 2. The remaining variables, however, have very similar scores, making it difficult to accurately make a distinction between them.

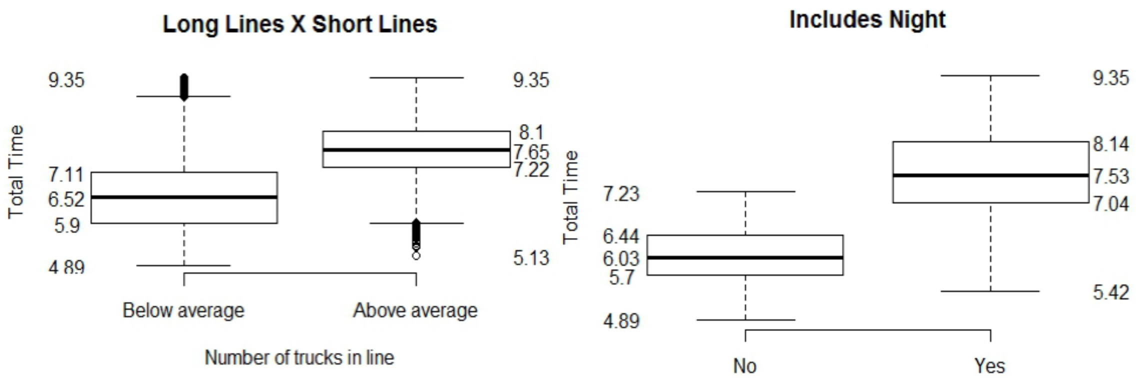
From the results of these three tests, it was possible to observe that the connection weight approach tends to predict the best variable very accurately. When the best variable is taken out of the models, it is possible to see another variable that stands out.

When many variables are included in the model, it is hard to determine an accurate ranking. Through the tests, it is possible to see that the first ranking created was not accurate, as variable IncludesNight was ranked as number four, while in the following tests it was ranked as the second most important variable. This indicates that an iterative process to rank variables might be beneficial, as it will allow variables to emerge unhindered by the score of the most important variable.

Test on Outcome Variable

To test whether input variables with a high relative impact had a stronger impact on the output variable, Total Time, we analyzed each variable. For each input variable, we split records in the data into two groups. One group was made up of records with higher than average values for the input variable. The other group had lower than average values for this variable. Then, we compared these two groups based on Total Time. For example, TrucksInLine had the highest relative impact in the study. Based on this result, we created two additional groups. One contained trucks that arrived with a below average number of trucks in line. The other group contained trucks that arrived with an above average number of trucks in line. We followed this process for the other input variables.

Figure 5 shows this comparison for the four input variables with the highest relative impact. In each of these comparisons, there was a large difference in Total Time between the records in the two groups for each variable.



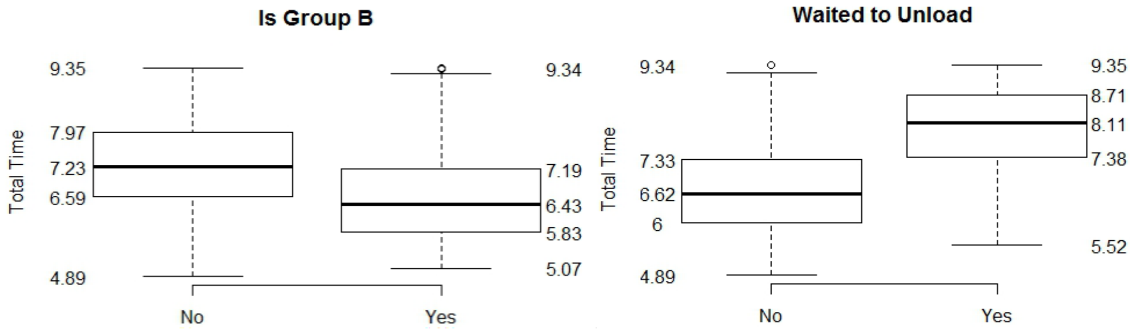
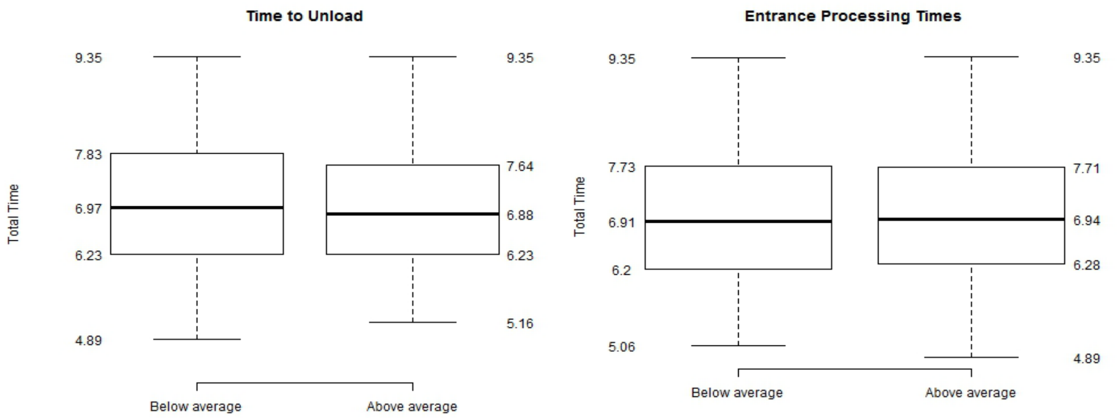


Figure 5. Total Time comparisons for input variables with high relative importance

Figure 6 shows the results of this process for four variables with low relative importance scores. For each of these input variables, the group with low values and the group with high values had similar values for Total Time.



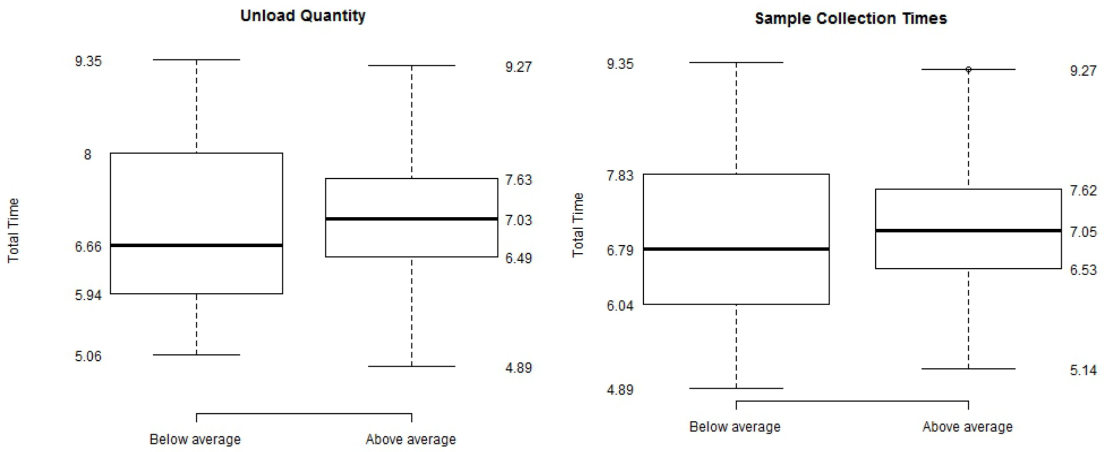


Figure 6. Total Time comparisons for input variables with low relative importance

Discussion

In this research we extended earlier work by combining the results of DES with predictive modeling using ANN. Specifically we used a DES model to generate records that were used by ANN predictive models. These models were then evaluated to determine the relative advantages of inputs by the connection weight approach. This allowed us to calculate and understand the relative importance of these variable without having to take the time to capture a detailed history of the overall raw materials receiving process at the company under study. This informed the input variables that should be focused on to improve the efficiency of the process.

Despite the high complexity of ANNs, it is possible to interpret them in terms of relative importance. And helpful insights can be extracted from this process, such as a better understanding of the relationships that exist between variables.

It is not enough to understand the relative importance of one ANN model because the random weights used to initiate the weights in ANN models can and do produce so many different predictive models. Because different models produce somewhat different relative importance scores for input variables, it is necessary to produce a non-trivial number of models and to examine the distribution of relative importance. Some ANN models are more predictive than others, so by producing a set of models and selecting the most predictive ones, the most predictive models can be studied.

In addition, we found that different numbers of nodes in the hidden layer produced very similar predictive capabilities. Although there was a best number of nodes (12 in this research), we observed that some models with fewer hidden neurons performed almost as well. Future research could investigate the effect of changing the number of hidden neurons on relative importance calculation results.

Figure 7 is an example of why it is conceptually complex to relate input variables to output variables in ANN models and why relative importance is helpful. First, the influence from each input is typically distributed through multiple neurons in the hidden layer. Second, it is difficult or impossible to create a meaningful conceptual abstraction for each hidden neuron. An input node transfers some of its influence via positive links and some of its influence by negative links. Likewise, the same neuron often transfers positive influence from some inputs and negative weights from other input variables. Third, weights on both the inputs-to-hidden links and on the hidden-output links may be either positive or negative, so resulting product values are sometimes positive and sometimes negative. Finally, when they are summed positive influences and negative influences are summed so that relative importance reflects the net effect of adding multiple positive and multiple negative values.

Input-Hidden Connection Weights

X \ Y	12 Hidden Nodes											
Input	1	2	3	4	5	6	7	8	9	10	11	12
TrucksinLine	4.4	-1.0	2.5	2.2	-1.7	-2.1	1.4	-1.8	-1.9	-2.1	3.0	1.2
IsGroupA	8.0	-1.8	-0.8	2.3	-2.3	-1.8	-0.4	1.0	0.6	4.6	1.8	-2.8
IsGroupB	-5.0	-1.1	0.5	-1.7	-0.5	-2.2	-2.9	-1.1	0.8	3.4	1.2	-2.6
IncludedNight	-1.5	-6.7	-0.6	0.0	0.4	-0.5	-4.5	-2.0	4.5	0.4	-6.3	0.7
WaitedtoUnload	4.7	-5.7	-0.6	0.4	-2.5	3.6	-2.7	-0.7	-2.7	-0.3	-0.6	1.4
Shortage	-2.8	4.3	1.9	-4.0	1.3	-0.3	4.2	5.4	-0.9	-0.5	-1.1	-8.2
IncludesWeekEnd	1.9	-4.1	-0.8	1.0	-0.4	-3.2	-3.6	-2.3	3.0	0.2	-1.0	0.6
WasPriority	0.8	0.2	-0.9	-1.6	-0.8	-0.3	1.7	1.4	-0.3	-0.4	1.3	0.0
IsGroupC	-0.3	1.8	1.9	2.7	2.6	-0.7	0.5	-0.1	0.9	-2.3	1.0	-1.0

X

Hidden-Output Connection Weights

	12 Hidden Nodes											
Output	-552	312	27	-25	168	-872	238	281	-566	1098	1681	1267

Connection Weight Products

X \ Y	Hidden Node												Relative	Normalized	Raw
Input	1	2	3	4	5	6	7	8	9	10	11	12	Importance	Importance	Rank
TrucksinLine	-2418	-303	67	-56	-292	1805	340	-510	1064	-2284	5060	1571	4045	0.94	2
IsGroupA	-4416	-568	-21	-57	-388	1604	-93	284	-317	5040	3093	-3548	614	0.76	4
IsGroupB	2743	-337	14	42	-81	1901	-683	-309	-464	3733	1933	-3307	5187	1.00	1
IncludedNight	817	-2078	-15	1	72	476	-1078	-573	-2530	395	-10540	862	-14191	0.00	9
WaitedtoUnload	-2567	-1766	-16	-11	-422	-3113	-638	-194	1545	-285	-958	1761	-6663	0.39	8
Shortage	1551	1351	50	101	217	262	1009	1515	532	-505	-1782	-10427	-6127	0.42	7
IncludesWeekEnd	-1065	-1264	-22	-24	-60	2764	-847	-635	-1715	209	-1597	798	-3458	0.55	6
WasPriority	-414	53	-23	41	-126	226	397	394	192	-395	2168	51	2564	0.86	3
IsGroupC	166	576	52	-67	437	625	126	-37	-526	-2525	1731	-1204	-646	0.70	5

Figure 7. Example of how Connection Weight Products Sum to Relative Importance for one Neural Network

The fact that both negative and positive relative importance scores can result from the connection weight approach is not a new finding. It was found by Holden et al. (2002) who created the method. However, in that study only four hidden neurons were used. In this research,12 hidden neurons were used, which increases the distribution of influence from the input variables across eight more neurons. The difficulty of conceptualizing so many positive and negative influences and netting them out makes the net effects represented by the relative importance values that much more important from a meaning standpoint.

Our research produced insights about ways the connection weight approach can be used in an iterative fashion to better understand the contribution of input variables.

Past research studies have applied this approach to input variables that were already known to contain predictive information. Our finding that the connection weight approach can be used to identify and eliminate potential input variables that contain little or no predictive information can be applied to future problems. In the problem studied in this paper, input variables with the lowest relative impact also were found to contain no predictive information.

Our finding that a dominating variable masked the potential relative contribution of other input variables can also be applied to future problems. In this research we found that by removing a dominating input variable that the relative contribution of the remaining input variables became clearer.

Conclusion

This paper presents an approach to studying a process that mitigates lack of historical data on the process by using simulation to create useful data. By creating a DES model that reflected the process, a distribution of inputs and other operating conditions were simulated to determine the output that would result for process. The resulting records we input into the predictive modeling process, where ANNs was found to be the most predictive method. We then successfully applied the connection weight approach to a set of high-performing ANN models determine the relative importance of input variables on the process outcome. We also found that applying the connection weight method sequentially can help eliminate non-predictive variables and that by removing a dominant input variable that the relative contributions of the remaining input variables can be characterized more clearly.

References

1. Nyce C, Cpcu A. Predictive analytics white paper. American Institute for CPCU Insurance Institute of America. 2007:9-10.
2. Lopes HdS, Lima RdS, Leal F, et al. Brazilian Soybean Transportation Analysis Through Discrete Event Simulation. 2018.
3. Andrew G, Chris O. Modelling people's behaviour using discrete-event simulation: a review. International Journal of Operations & Production Management. 2018;38(5):1228-1244. doi: doi:10.1108/IJOPM-10-2016-0604.
4. Ahalt V, Argon NT, Ziya S, et al. Comparison of emergency department crowding scores: a discrete-event simulation approach. Health care management science. 2018;21(1):144-155.
5. Liu R, Xie X, Yu K, et al. A survey on simulation optimization for the manufacturing system operation. International Journal of Modelling and Simulation. 2018 2018/04/03;38(2):116-127. doi: 10.1080/02286203.2017.1401418.
6. Rodriguez CM. Evaluation of the DESI interface for discrete event simulation input data management automation. International Journal of Modelling and Simulation. 2015 2015/01/02;35(1):14-19. doi: 10.1080/02286203.2015.1073891.
7. Better M, Glover F, Laguna M. Advances in analytics: Integrating dynamic data mining with simulation optimization. IBM Journal of Research and Development. 2007;51(3.4):477-487. doi: 10.1147/rd.513.0477.
8. Brady TF, Yellig E. Simulation data mining: a new form of computer simulation output. Proceedings of the 37th conference on Winter simulation; Orlando, Florida. 1162762: Winter Simulation Conference; 2005. p. 285-289.
9. Ghasemi S, Ghasemi M, Ghasemi M. Knowledge Discovery in Discrete Event Simulation Output Analysis. In: Pichappan P, Ahmadi H, Ariwa E, editors. Innovative Computing Technology: First International Conference, INCT 2011, Tehran, Iran, December 13-15, 2011. Proceedings. Berlin, Heidelberg: Springer Berlin Heidelberg; 2011. p. 108-120.
10. Kostakis H, Sarigiannidis C, Boutsinas B, et al. Integrating activity-based costing with simulation and data mining. International Journal of Accounting & Information Management. 2008;16(1):25-35.
11. Somers MJ, Casal JC. Using artificial neural networks to model nonlinearity: The case of the job satisfaction—job performance relationship. Organizational Research Methods. 2009;12(3):403-417.
12. Haddouche R, Chetate B, Said Boumedine M. Neural network ARX model for gas conditioning tower. 2018.
13. Tsang M, Cheng D, Liu Y. Detecting statistical interactions from neural network weights. arXiv preprint arXiv:170504977. 2017.
14. Gedeon TD, Wong PM, Harris D, editors. Balancing bias and variance: Network topology and pattern set reduction techniques. International Workshop on Artificial Neural Networks; 1995: Springer.
15. Funahashi K-I. On the approximate realization of continuous mappings by neural networks. Neural networks. 1989;2(3):183-192.
16. Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. Neural networks. 1989;2(5):359-366.

17. De Oña J, Garrido C. Extracting the contribution of independent variables in neural network models: a new approach to handle instability. Neural Computing and Applications. 2014;25(3-4):859-869.
18. Olden JD, Joy MK, Death RG. An accurate comparison of methods for quantifying variable importance in artificial neural networks using simulated data. Ecological Modelling. 2004 11/1/;178(3–4):389-397. doi: http://dx.doi.org/10.1016/j.ecolmodel.2004.03.013.
19. Garson GD. Interpreting Neural-Network Connection Weights. AI Expert. 1991;6:47-51.
20. Goh AT. Back-propagation neural networks for modeling complex systems. Artificial Intelligence in Engineering. 1995;9(3):143-151.
21. Olden JD, Jackson DA. Illuminating the "black box": a randomization approach for understanding variable contributions in artificial neural networks. Ecological Modelling. 2002 8/15/;154(1–2):135-150. doi: http://dx.doi.org/10.1016/S0304-3800(02)00064-9.
22. Dimopoulos I, Chronopoulos J, Chronopoulou-Sereli A, et al. Neural network models to study relationships between lead concentration in grasses and permanent urban descriptors in Athens city (Greece). Ecological Modelling. 1999 8/17/;120(2–3):157-165. doi: https://doi.org/10.1016/S0304-3800(99)00099-X.
23. Dimopoulos Y, Bourret P, Lek S. Use of some sensitivity criteria for choosing networks with good generalization ability [journal article]. Neural Processing Letters. 1995;2(6):1-4. doi: 10.1007/bf02309007.
24. Scardi M, Harding Jr LW. Developing an empirical model of phytoplankton primary production: a neural network case study. Ecological Modelling. 1999 8/17/;120(2–3):213-223. doi: https://doi.org/10.1016/S0304-3800(99)00103-9.
25. Lek S, Belaud A, Baran P, et al. Role of some environmental variables in trout abundance models using neural networks. Aquat Living Resour. 1996;9(1):23-29.
26. Lek S, Delacoste M, Baran P, et al. Application of neural networks to modelling nonlinear relationships in ecology. Ecological Modelling. 1996 1996/09/01;90(1):39-52. doi: http://dx.doi.org/10.1016/0304-3800(95)00142-5.
27. Gevrey M, Dimopoulos I, Lek S. Review and comparison of methods to study the contribution of variables in artificial neural network models. Ecological Modelling. 2003 2/15/;160(3):249-264. doi: http://dx.doi.org/10.1016/S0304-3800(02)00257-0.