*Journal of Industrial and Systems Engineering*
Vol. 3, No. 4, pp 257-274
Winter 2010

# Using Neural Networks with Limited Data to Estimate Manufacturing Cost

**Gary R. Weckman[1*], Helmut W. Paschold[2], John D. Dowler[3], Harry S. Whiting[4], William A. Young[5]**

[1,3,4,5] Department of Industrial and Systems Engineering, Ohio University, Athens, Ohio, USA
[1] weckmang@ohio.edu

[2] School of Public Health Sciences and Professions, Ohio University, Athens, Ohio, USA

**ABSTRACT**

Neural networks were used to estimate the cost of jet engine components, specifically shafts and cases. The neural network process was compared with results produced by the current conventional cost estimation software and linear regression methods. Due to the complex nature of the parts and the limited amount of information available, data expansion techniques such as doubling-data and data-creation were implemented. Sensitivity analysis was used to gain an understanding of the underlying functions used by the neural network when generating the cost estimate. Even with limited data, the neural network is able produced a superior cost estimate in a fraction of the time required by the current cost estimation process. When compared to linear regression, the neural networks produces a 30% higher ***R*** value for shafts and 90% higher ***R*** value for cases. Compared to the current cost estimation method, the neural network produces a cost estimate with a 4.7% higher ***R*** value for shafts and a 5% higher ***R*** value for cases. This significant improvement over linear regression can be attributed to the neural network ability to handle complex data sets with many inputs and few data points.

***Keywords*:** neural network, cost estimation

## 1. INTRODUCTION

Cost estimation integrates specifications, dimensions and other data about an object and calculates cost. It is an important part of product design; for example, with the building of a highway, designing of a new vehicle, or construction of a new factory, a more cost-effective approach or components cost-lowering can be achieved in order to permit options that are more expensive elsewhere in the design. Cost estimation involves taking proposed dimensions, attributes, and other factors, and merges these constraints with knowledge of the past to develop an estimated cost for an object.

A relatively new method application for cost estimating is the use of artificial neural networks or ANNs (Smith and Mason, 1997). One major benefit of using an ANN is its ability to understand and simulate complex functions. One of the current drawbacks to such use is that as the complexity of the function increases, more training data is necessary if the ANN is to be robust enough to

[*] Corresponding Author

produce good results. Neural networks can be used to create more complex functions than older methods such as linear regression, making the amount of data available increasingly important.

Production data, part designs, and part costs from a large aircraft engine manufacturer were used to train and test a neural network. Two different part families, shafts and cases, were evaluated to represent different levels of part complexity. Little has been published on using real world data to train and test an ANN that can produce a cost estimate. The few prior studies demonstrate that ANNs have good potential in their ability to create complex models and provide a superior cost estimate compared to estimation methods such as linear regression (Smith and Mason, 1997).

The cost estimation process is important in the world of design and manufacturing. Cost estimates allow the designer and engineer to be aware of the cost implications for the design decisions they make while still in the design phase. Reliable cost estimates also allow management to make an informed decision as to what products will be profitable and what products should be redesigned. Currently, the large aircraft engine manufacturer uses cost estimation software to estimate part costs at the preliminary design level. This method requires the creation of multiple cost-estimating functions that are very complex and time consuming. Given enough data, an ANN will reconstruct the cost estimation function and produce a better cost estimate in a fraction of the time required in current methods.

## 2. LITERATURE REVIEW

Cost estimation is the evaluation of many factors the most prominent of which are labor, and material (Smith and Mason, 1997). Many methods and procedures have been developed to calculate the cost of a product, each with their own pros and cons. The majority of the models rely on historical data, which do not consistently provide an accurate picture of the current conditions and are not always available. Historical data models often produce results with a low level of accuracy due to these limitations. Statistical models in current use include regression, bottoms-up, parametric, and more recently neural networks (Layer et al., 2002). Regression is “the mathematical nature of the association between two variables” according to the NASA parametric cost estimation handbook 2004. Regression derives an association by using historical data about the part or process to find the best relationship between the cause attributes and the output value (Walpole et al., 2002).

Linear regression attempts to fit a straight line to the data set and minimize the sum of the squared error. In many instances, the data set contains more than one input that affects the result; in this case, multiple linear regression can be used. The accuracy for cost estimates using regression analysis varies greatly, depending on the complexity of the function being predicted and the amount of data available. Generally, regression has lower accuracy than more complex models, but the process is relatively simple and fast (NASA, 2004). Regression is commonly used as a basis for comparison of new cost estimation techniques (Smith and Mason, 1997).

The bottoms-up method of cost estimation is the process of breaking a part or process down into lower levels of detail, estimating the cost associated with the lower level of detail, and summing the estimates to provide a cost estimate for the part at the top level (NASA, 2004). For this method, information about each of the lower levels is needed. This requires the user to have product and process knowledge to produce the cost estimate (Divelbiss, 2005). This type of cost estimate provides a high level of transparency, allowing the user to see what is driving the cost estimate. The disadvantages of the bottoms-up method include significant cost, time, and the need for user detailed knowledge of cost relationships (NASA, 2004).

Parametric cost estimating uses cost estimating relationships, CERs, to produce a cost estimate. The CERs are created using historical data and an equation that describes the relationship between the inputs and the output (NASA, 2004). The use of parametric cost estimating procedures defines a cost estimate equation using only a portion of the attributes that describe the part. Once defined, CERs quickly produce a relatively accurate cost estimate, with little knowledge being required about the part or processes producing the part. A simple example of parametric cost estimation is estimating the cost of a pizza based on the number of slices it contains. If it were known that a slice of pizza costs $1.50, and a large pizza contains twelve slices, it would have an estimated cost of $18.00.

The major drawback to using CERs is that the process of definition requires a considerable amount of effort and time (NASA, 2004). Other disadvantages include a lack of transparency of the cost driving attributes, reliance on historical data, and an inability to produce an accurate cost estimate when the new data does not fall in the range used to create the cost estimate relationships.

Neural network models have been designed and used for a number of projects similar to the CER process. In the financial sector, an ANN had been evaluated for credit scoring models (West, 2000), with ANN results found to be only up to 3% more accurate than traditional quantitative models. West et al. (2005) later reported that ensembles of neural network predictors were more accurate in credit scoring models than a single best multilayer perceptron neural network. An ANN was found to provide greater accuracy than traditional simple methods for electricity price forecasting (Yamin et al., 2004). The neural network was found to be a useful tool in construction cost estimating with good short term results (Kim et al., 2004) and average cost estimation for building structural accuracy of 93% (Günaydin and Doğan, 2004). Recognizing that over 70% of a product's life cycle cost is committed in the early design stage, an ANN was successfully used to apply information from life cycle cost models (Seo et al., 2002). The ANN appeared to have an advantage over parametric models were the CER form is not known or well defined (Cavalieri et al., 2004).

### 2.1. Artificial Neural Networks

Neural networks are statistical models of real world systems which are built by tuning a set of parameters. These parameters, known as weights, describe a model which forms a mapping from a set of given values known as inputs to an associated set of values the outputs (Swingler, 1996).

A neural network is created with one or more levels of hidden nodes to model a system. There are multiple connexions from the inputs to the nodes on the hidden layers to the output, which all have a weight and error term, which are adjusted through the training process. Since there is a tremendous amount of interconnectedness between all nodes from the inputs to the output, a neural network can expose relationships that were previously unsuspected before the analysis. During the training process, the network assigns weights to the nodes to achieve the best relationship between the training input and output values. The neural network runs through the process many times adjusting weights to minimize the error. Once trained, the network has a model with the weights that provided the best results to calculate the estimate for a part that was not in the training data set. An ANN creates an optimal model that is general in nature and produces a small mean square error for data that was not in the training set (Prechelt, 1998).

## 2.2. Neural Network Design

When designing an ANN, it must be determined experimentally how many hidden nodes are required, and what learning algorithm should be used in training. The design of the network has a large impact on its performance (Schenker and Agarwal, 1996). The number of nodes directly affects the network's ability to learn and generalize the desired function. If there are too many nodes, the network will memorize the input data and provide poor generalization (Prechelt, 1998). If there are too few nodes the network will not be able to fully learn and will provide poor results for the training and testing data.

The number of hidden nodes used is typically decided through a process of trial and error, and has a large impact on the network's performance. Some research has shown that the difference between 4 and 16 hidden nodes affected the error value by only three percent (Shlub and Versand, 1999). The low difference could be the result of estimating a simple function. As the function complexity increases, the network requires more hidden nodes to achieve a good result.

## 2.3. Multilayer Perceptrons

In a typical multilayer perceptrons (MLP) neural network, each node is connected to all the nodes in the previous layer as well as all the nodes in the next layer (Figure 1). Weights are assigned during training to each arc between the nodes. The hidden layer is comprised of the nodes chosen in the design phase. Each node takes the input values, associated weights, and runs them through the chosen function (Principe et al., 1999). The function chosen affects how and how well the network is able to learn (2004). The node then uses a transfer function to produce a weight-associated output. The hidden node values and weights are run through the output node (layer) algorithm, and a final output value is calculated.

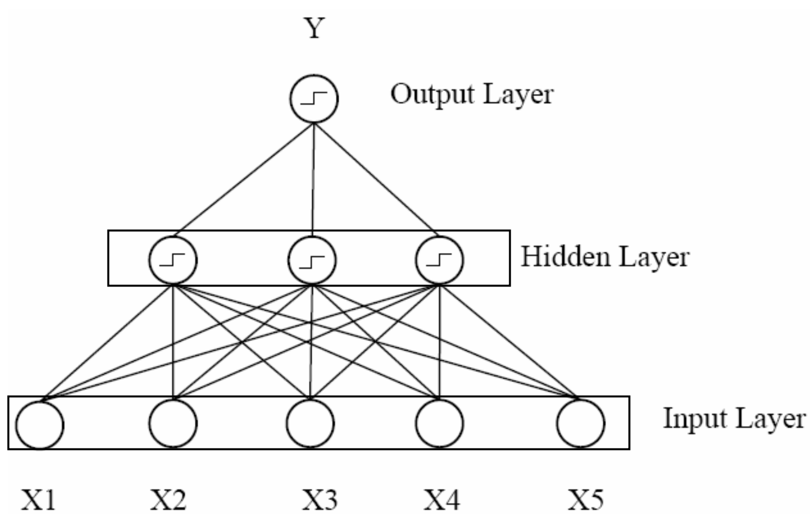


Figure 1 General Neural Network Layout (Seo et al., 2002)

MLP neural networks are commonly used in regression and classification problems. They are capable of modeling many functions but require a large amount of time, epochs, and nodes (Fine, 1999).

## 2.4. Cross Validation

A possible learning ability limitation of a neural network is over learning, or memorization. Memorization provides good results for the test data, but when faced with a part that was not in the test data set, provides poor results (Yuval, 2000). The network is only able to recognize parts exactly as if it has seen before. Any variation in the attributes puts a large amount of variation into the network, and can produce estimates with a large amount of error. The process of cross validation removes the risk of the network memorizing the data (Smith and Mason, 1997).

Cross validation uses its own data set to monitor the neural network's ability to produce generalized cost estimates. For all output-analysis iterations, the results for the cross validation set are also analyzed. Figure 2 shows a typical error graph for a neural network using cross validation. The network continues to learn as the error for the cross validation data set and the training data set continue to decrease. Once the error for the cross validation data set starts to increase, the training stops and the weight values that provided the lowest error for the cross validation data set are considered optimal (Swingler, 1996).

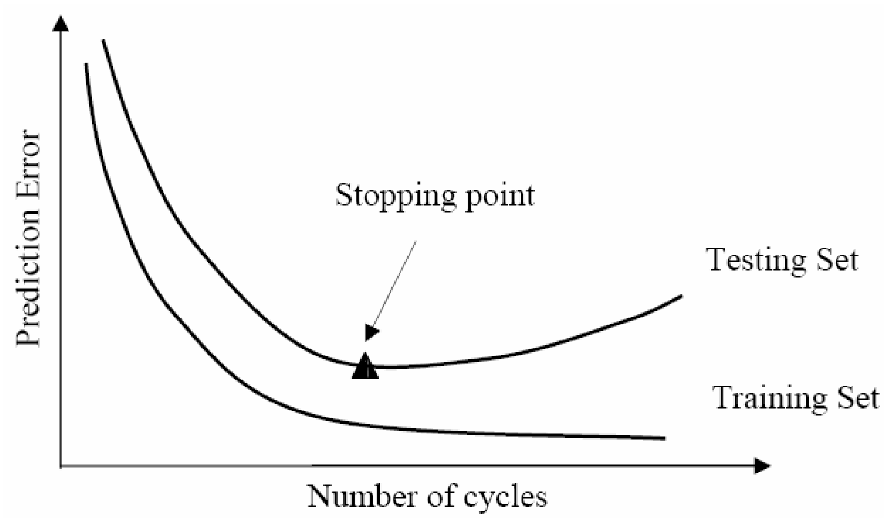


Figure 2 Typical error graph for a neural network using cross validation.

The training data set error value may continue to decrease once the cross validation error starts to rise. At this point, the network is over-fitting and memorizing the training data. If cross validation is not used, then a testing data set is critical to prove that memorization and over-fitting is not occurring (Kim and Han, 2003).

## 2.5. Neural Networks versus Regression

The idea of using neural networks to estimate cost is relatively new and little work has been published showing their abilities (Zhang and Fuh, 1998). The research conducted to date has shown neural networks to be superior to regression, but most of the results were flawed in some way (Smith and Mason, 1997).

Many research papers proved neural networks superior to regression but failed to use either cross validation or a testing set. When cross validation or a testing set is not used, the squared error was shown to more than double when compared to a generalized network on a testing set of data not used to train the network (Shlub and Versand, 1999).

When cross validation or a testing set was used, the results were sometimes artificially inflated. In an estimate using neural networks for the cost of packaging products, researchers were able to get their average error under 10% (Zhang and Fuh, 1998). In order to accomplish this they removed all parts from their testing set that had an error over 15%, claiming that they must contain a feature or attribute that is not covered in the training set. This may be true, but by removing those parts, they are eliminating poor results from their testing set. A better method would have been data randomization to achieve a more representative training data set.

The limited research available shows that neural networks have the ability to be a good cost estimation tool. Very little current neural network research uses real world data for testing, preferring the use of created data instead. Research reported a 55% and 5% decrease in error when compared to regression (Shlub and Versand, 1999; Smith and Mason, 1997). The research that provided a 55% decrease did not use cross validation or a test data set, so the results may be inflated by over-learning.

The largest hurdle to the widespread use of neural networks for cost estimation is the black box nature of neural networks and their inability to duplicate results (Smith and Mason, 1997). Due to the initial weights being randomly assigned, it can be very difficult to recreate a network. This makes it difficult to persuade individuals who do not have a broad understanding of the technology that the neural network is producing a reliable cost estimate. The advantages of using neural networks to estimate cost include accuracy, time, and the ability for a person with very little knowledge about the part or neural networks to produce a good cost estimate once the network has been trained (Zhang & Fuh, 1998).

## 3. METHODOLOGY

### 3.1. Data

Proprietary data obtained from a large aircraft engine manufacturer's part drawings and cost databases was given to one of the paper authors in his role as a contract employee and master degree candidate. Data describing two jet engine part families, shafts, and cases were used to train and test the neural network. Shafts and cases were chosen because they provided two different part complexity levels.

The shaft attached to moving components and delivered the rotary power developed in the engine. It has a feature on both ends connected by a straight shaft and can have up to six other features including arm cones, holes, tabs, and splines. Figure 3 shows a cross section example of a generic shaft that is a flange connected to an arm cone, which connects to the main body of the shaft. The average shaft costs a third of what the average case costs.

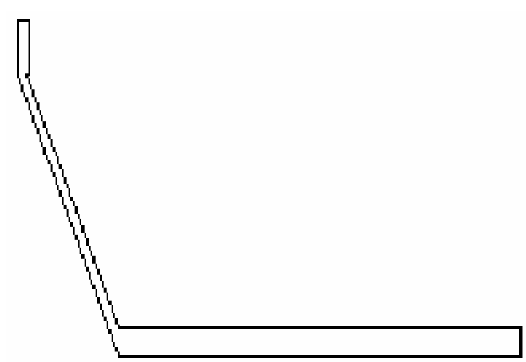

Figure 3 Example Shaft Cross Section.

A case will have a more complex shape, rotated through a circle for enclosing the vane stages along with up to fourteen other features such as hooks, holes, attachment rings, and tabs.
Figure  shows a cross section profile example of a generic case that comprises an outside diameter, or OD flange connected to a series of eight hooks and ending with an OD flange. This particular case also has two attachment rings located on the OD of the part that appear as blocks of material.

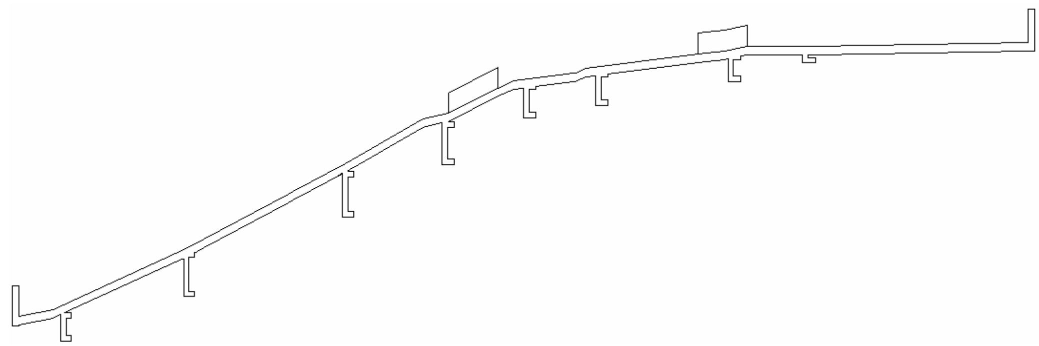

Figure 4 Example of a turbo-machine case cross section.

The case data comprised 132 attributes describing the part, while the shaft data has 33 attributes. Once the symbolic attributes have been converted to numeric, the number of case attributes jumped to 146 and the number of shaft attributes became 71. The attributes describe the location in two-dimensional space and the size of all the features that describe the overall part. Data describing general attributes of the part such as material, manufacturing location, and actual part cost were also used.

Attributes for 60 cases and 37 shafts were used for the training and testing of the neural network. The data was randomly assigned with 70% going to training, 15% to cross validation and 15% to testing. This is a common ratio and was used for all networks. The number of parts was relatively small and the number of attributes relatively large; it was a challenge to create a quality network that was capable of producing an accurate cost estimate.

### 3.2. Data Expansion

Neural networks work best with a large sample size, or many exemplars. The data sets for cost estimation for this problem were small compared to those needs. Because of this limitation, several methods were utilized to expand the data set. Doubling data, doubling data with holdout data, and data creation were used to for testing. Doubling data is frequently used to expand data sets for neural networks when more data is needed, but does leave the difficulty of overtraining on the data since all exemplars might appear in the training, cross validation and testing data. A better method is doubling data with holdout, where a percentage of data is withheld to be used only in the testing stage of the neural network. In this case, the holdout was set at 5%.

Although data creation was tried with this project, the complexity of creating data that could reflect true design choices for turbo-machinery shafts or cases made the results somewhat equivocal. The same data creation methods were tried with another data set, the ‘Saginaw’ data, which dealt with biological data from Saginaw Bay, Michigan (Millie and Weckman, 2006).

Table 1 Size of data sets

| | Base data | Double data | Double data with holdout | Created data |
|---|---|---|---|---|
| **Shafts** | 36 | 72 | 67 | 3644 |
| **Cases** | 60 | 120 | 115 | 5920 |
| **Saginaw** | 50 | 100 | 95 | 4937 |

### 3.3. Neural Network Training

For each data set, the data was randomized. The networks were trained multiple times to increase the probability of producing a set of good initial weights in the model. Each training run is stopped when one of the following three conditions was met:

1) The maximum number of epochs was reached,
2) The cross validation error started increasing, or
3) Five hundred epochs without improvement in error is reached.

For the purposes of this research project, the maximum number of epochs was set to 15,000, giving the network sufficient time to learn without overwhelming computing resources and time. The network was set to stop learning after 500 epochs without improvement. This limits the required computation time while assuring that the network has not reached a local minimum. Cross validation was used in all neural network runs to avoid over-learning.

### 3.4. Neural Network Results

The quality of the cost estimate was determined by the $\boldsymbol{R}$ value and the mean squared error value. The $\boldsymbol{R}$ value is calculated by taking the square root of the $\boldsymbol{R^2}$ value. $\boldsymbol{R^2}$ is also known as the coefficient of determination, showing the amount of variation the model is able to describe. Least squares regression was used as a baseline to which the neural network's results were compared.

As a comparison, the computer software Minitab was used to conduct stepwise regression on both case and shaft data sets. Stepwise regression was used due to the large number of variables and the small number of data points. The $\boldsymbol{R^2}$ value for the regression model was computed and compared to the neural network results.

### 3.5. Knowledge Extraction

Due to the black box nature of neural networks, sensitivity analysis was used to evaluate the relationships found between inputs. Sensitivity analysis is a tool capable of quantifying some of what the neural network learned.

Sensitivity analysis gives an insight into the relationship between input variables and the output. This allows the user to see how the input affects the output over a range of values if all other inputs remain constant. In a neural network, this process has no effect on the training or the network's weights and is conducted on a fully trained neural network. The network weights that provided the best result are used for the sensitivity analysis. Once the network weights were loaded, the neural network was run with the input data. The input value of interest was varied by a small percentage while the rest of the input values were held constant. The network was run again with the altered input value and the difference in the result was calculated. The attribute value continued to be changed and the network ran until the range of interest had been covered. The result values could then be graphed to show how the output was affected by that particular attribute.

## 4. RESULTS

### 4.1. Shaft Network Design

The shafts network design analysis results are presented in Table 2. The 12-6-1 and the 16-8-1 networks provided superior results over the other options tested. Once the network exceeded 48-24-1, the network was unable to train.

The 12-6-1 and the 16-8-1 network structures were both run a second time in order to optimize design. The 16-8-1 network structure provided both the highest and the lowest ***R*** value of the group and showed signs of training difficulty. The 12-6-1 network structure provided a consistent ***R*** value along with the best training mean square error and training. Due to its stable training and favorable results, the 12-6-1 network structure was used for all shaft networks.

Table 2 Shaft network testing results

| # Nodes Hidden Layer 1 | # Nodes Hidden Layer 2 | Max Training epoch # | Training MSE | Testing R | Testing NMSE |
|---|---|---|---|---|---|
| 4 | 2 | 814 | 0.00061600 | 0.8046 | 0.6042 |
| 8 | 4 | 1496 | 0.00019000 | 0.7948 | 0.5050 |
| 12 | 6 | 6000 | 0.00000003 | 0.8409 | 0.4917 |
| 12 | 6 | 2681 | 0.00000450 | 0.7504 | 1.3910 |
| 16 | 8 | 6000 | 0.00000016 | 0.9220 | 0.3130 |
| 16 | 8 | 550 | 0.00001590 | 0.6599 | 0.6248 |
| 32 | 16 | 512 | 0.00000000 | 0.7610 | 0.6600 |
| 48 | 24 | 504 | 0.00000000 | 0.7664 | 0.5670 |

### 4.2. Case Network Design

The process for shafts was repeated for cases and the results can be seen in Table 3. Due to these factors, a 48-24-1 network structure was used for all case networks.

Table 3 Case Network Testing Results

| # Nodes Hidden Layer 1 | # Nodes Hidden Layer 2 | Max Training epoch # | Training MSE | Testing R | Testing NMSE |
|---|---|---|---|---|---|
| **8** | **4** | 839 | 0.00022150 | 0.0180 | 1.4017 |
| **12** | **6** | 571 | 0.00025410 | 0.7259 | 0.6560 |
| **16** | **8** | 536 | 0.00006960 | 0.7296 | 0.5510 |
| **32** | **16** | 2421 | 0.00000696 | 0.7667 | 0.6663 |
| **32** | **16** | 514 | 0.00004917 | 0.6020 | 0.7967 |
| **48** | **24** | 526 | 0.00003259 | 0.7846 | 0.6300 |
| **52** | **30** | 505 | 0.00001300 | 0.4900 | 0.9540 |
| **64** | **32** | 513 | 0.00000623 | 0.6042 | 0.8756 |

### 4.3. Neural Network Cost Estimation Results

In this section, the results of using a neural network to create a cost estimate for shafts and cases are presented and discussed in comparison to the current cost estimation method software and regression.

The network parameters for shafts and cases were applied to their respective base data sets. The data was randomized and divided into training, cross validation, and testing sets. This process was repeated and the networks were run eight times for both shafts and cases. The ***R*** values achieved by the neural networks using the base data sets, as well as the ***R*** values regression, and the cost estimation software achieved using the same data are presented in Table 4.

Table 4 ***R*** of base data set

| Data Set | Neural Network | Cost Estimation Software | Regression |
|---|---|---|---|
| Shafts | 0.9736 | 0.9277 | 0.6840 |
| Cases | 0.7619 | 0.7233 | 0.0594 |

The results show that the neural networks were able to produce an estimate with a 0.9736 ***R*** value for shafts, a 4.71% increase compared to the current traditional cost estimation software, and a 0.7619 ***R*** value, or 5.07% increase, for cases. This improvement is obtained with minimal to no understanding of the labor, processes, or material that goes into producing the parts.

When compared to regression, neural networks performed 29.75% better with an ***R*** value 0.2896 greater for shafts, and 92.2% better with an ***R*** value 0.725 greater for cases. This can largely be attributed to regression's inability to produce a generalized model from the small complex data sets. The downward trend from neural networks to the cost estimation software is consistent from shafts to cases and can be associated with the neural networks' ability to model complex functions. The high variability in the downward trend from neural networks to regression is a direct reflection on the complexity of the data sets and weakness of the regression technique.

### 4.4. Knowledge extraction, sensitivity analysis

Sensitivity analysis was performed on the best networks provided by the shafts base and the cases base data sets. The sensitivity analysis allowed the relationships between part attributes and the part costs derived by neural networks to be examined. Sensitivity analysis graphs were generated by varying each input attribute slightly and reporting the effect the change had on the result. The cost values reported in the sensitivity graphs were scaled in order to mask proprietary data.

Sensitivity results for the shaft attributes were found to have a major impact on the shaft total cost (Figure 5). The forward flange type, aft flange type, manufacturing source, and presence of a radial hole set, had the greatest impact on the part cost. The remaining attributes had lesser varying degrees of impact on the part cost. When the shaft attributes with large impacts were examined, some unexpected relationships were seen. The sensitivity report showed that outsourced labor cost was significantly higher than the in-house labor options and parts with radial holes cost less than parts without radial holes. This seems counterintuitive as the radial hole process requires milling and drilling that would not be present in parts without radial holes. The neural network was most likely using radial holes as a way to differentiate shaft types and associating the cost of that closely correlated, undefined attribute to the radial hole attribute.

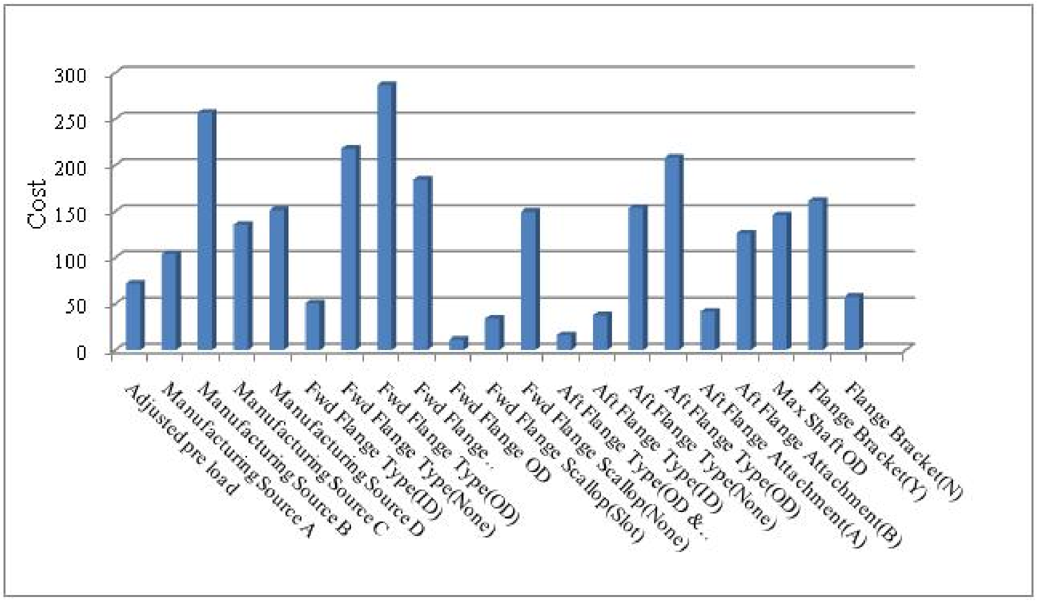


Figure 5 Partial shaft sensitivity report.

The sensitivity report for flange type showed both expected and unexpected relationships - flange type OD had the greatest impact, followed by flange type "none," and then flange type inner diameter or ID. Due to the increase in machining and material, it was expected that the absence of a flange would cause a significant reduction in sensitivity from OD and ID flange types. The same relationships between flange type OD, flange type ID, and flange type "none" were seen for both the forward and aft flanges. This suggests some factor was causing ID flanges to have a smaller cost impact than with no flange or OD flange. One possibility was that the neural network was using flange type to classify the shafts and associating the cost of that undefined attribute to the flange type. Another reason that a part with an ID flange would cost less than a part with no flange was that they are forged the same, and the part with no flange requires more machining in order to remove the material that forms the flange on the part with an ID flange.

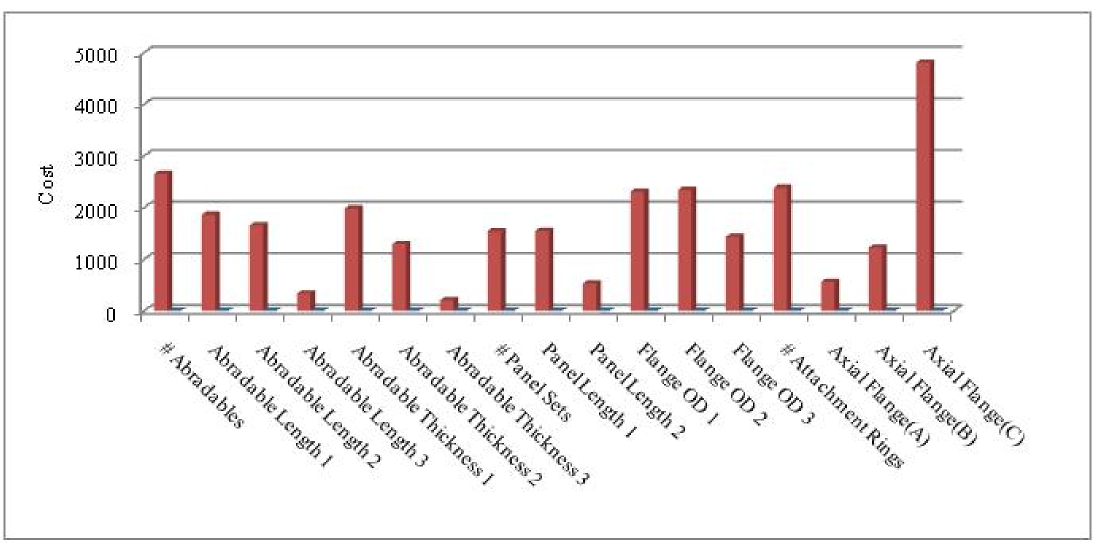


Figure 6 Partial case sensitivity report

The sensitivity results for the case attributes were found to have a major impact on the case's total cost (Figure 6). The report shows that the axial flange type has the largest impact on the case cost with type C having the largest impact and type A having the smallest impact. This is an interesting relationship as type C requires more material but less machining then type A, showing that material is the main cost driving factor.

The abradable and panel case attributes significantly influenced the case total cost as both abradables and panels are additional parts of a case, separate from the body of the part. Each of these separate pieces requires additional time to assemble and cost to manufacture depending on their size, making thickness and length cost-driving attributes.

The case flange diameter and number of case attachment rings were also found to have a significant impact on case costing. The impact of the flange diameter can be attributed to the generality with a larger flange OD, the part will be larger and require more material; therefore, more expensive. An attachment ring is a circumferential ring of material located on the outside diameter of the shaft that requires machining. The impact of the number of attachment rings can be attributed to the significant increase in material and machining each attachment ring represents.

Sensitivity analysis can also be performed on individual attributes by holding all other attribute values at their average and varying the attribute of interest over its range of possible values. The resulting graphs showed the effect an attribute of interest had on the overall part cost over its range of possible values. The graphs have been scaled to show the change in cost instead of the change in total part cost.

Figures 7 and 8 present the effect of total length on cost. As the total length increased, the cost increased at a near linear rate. This was an expected relationship; explained by the linear increase of material and machining required as the part increases in length.

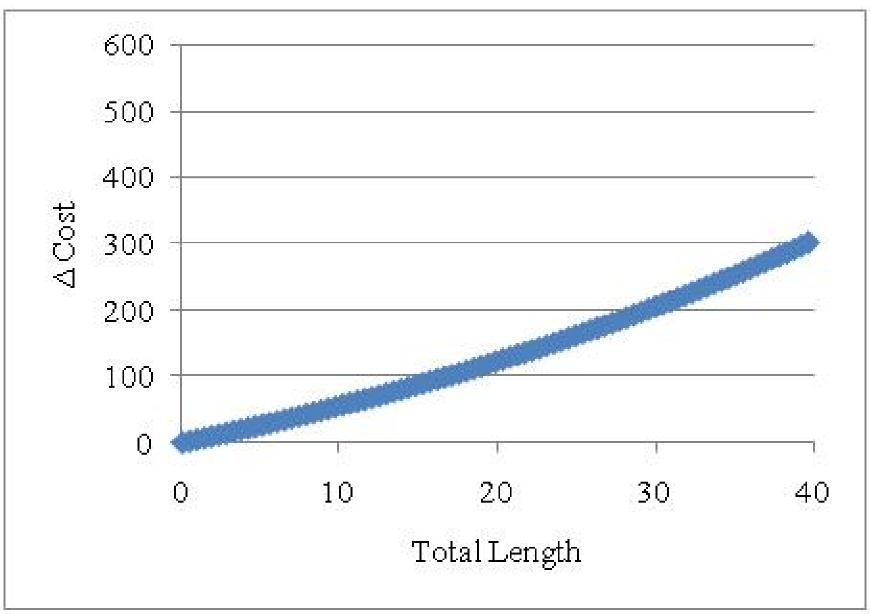


Figure 7 Shaft sensitivity for total length

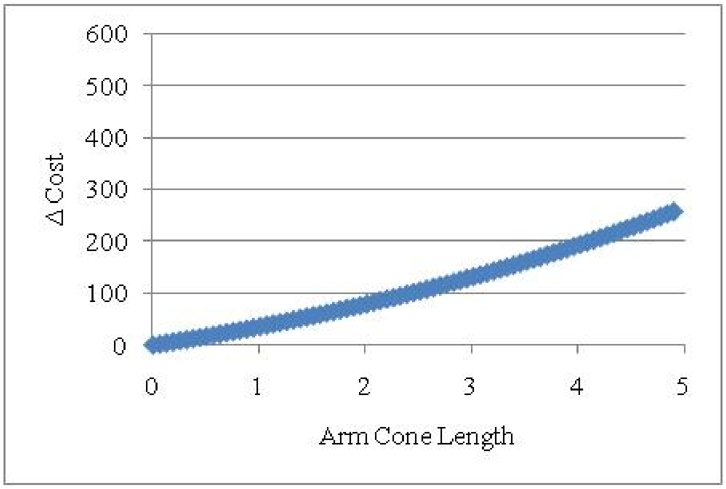


Figure 8 Shaft sensitivity for cone length

The flange hole tolerance was known to have a considerable impact on the shaft overall cost, with TH or tight tolerance holes being more expensive to produce then LH or loose tolerance holes. The relationship the neural network derived for a shaft flange depending on if it has TH or LH holes is presented in Figure 9. A value of zero represents a LH flange and a value of one represents a TH flange. A LH flange costs 600 units less than a TH flange, which is the expected relationship as TH flanges are known to require extra machining time.

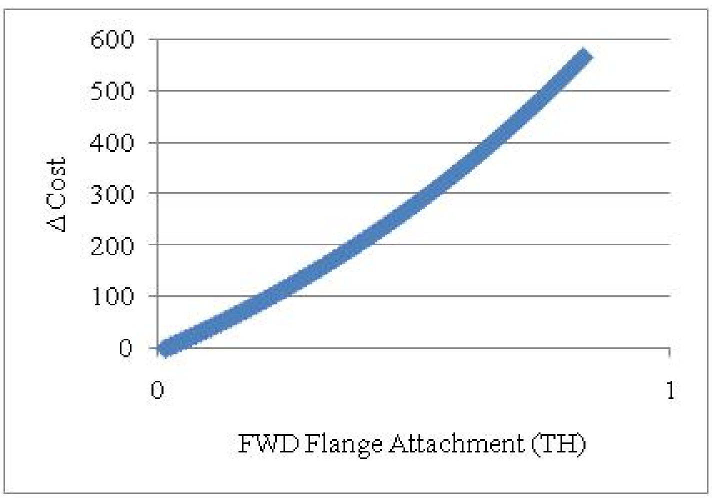


Figure 9 Shaft sensitivity for Fwd Flange Attachment (TH)

Figure  presents the cost impact relationship found by neural networks that the Aft ID had on shaft total cost. As expected, an increase in Aft ID required more material making the shaft cost higher.

The sensitivity graph for shaft Fwd ID seen in
Figure  showed the opposite relationship as the previous example. As the Fwd ID became larger, the shaft total cost decreased. This relationship seems illogical, as a larger Fwd ID would mean a larger

part and require more material. In addition, the relationship predicts an Fwd ID above 12 would have a negative cost, which is not realistic.

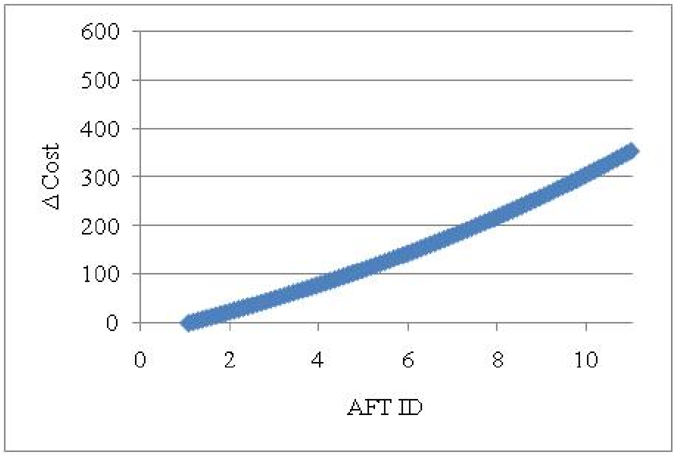


Figure 10 Shaft sensitivity for Aft ID

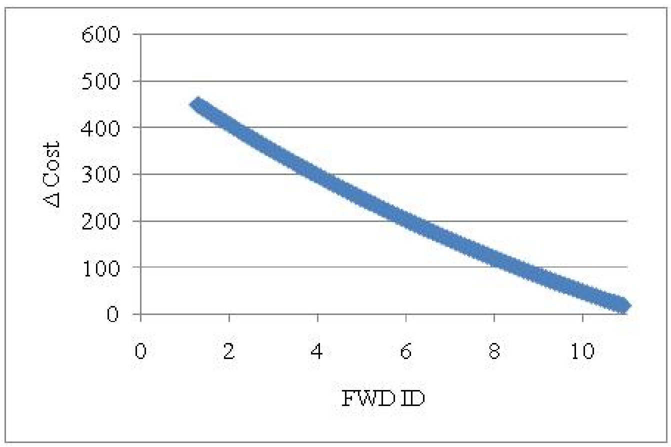


Figure 11 Shaft sensitivity for Fwd ID

A small amount of unrealistic or illogical sensitivity graphs are not uncommon for a neural network. They are most commonly caused by two or more closely correlated attributes working together. One of the attributes will show the expected relationship while the other correlated values are used by the neural network, in possibly illogical ways, in order to obtain the best overall result. This is possible because the neural network works without constraints or restrictions.

The sensitivity graph for the case thickness cost relationship is presented in Figure 12 and again shows an illogical result. As the case becomes thicker, its cost is reduced. This is counterintuitive and would mean a case that is infinitely thick would have less of a cost impact than a case 0.1 thick. This might be due to increased cost for machining; a thinner case being more difficult to work possibly leading to many rejects or need for rework. The labor costs are very high for the skilled trades performing finish work with more difficulty in producing a thinner case.

The number of tabs on a case had a considerable impact on the case overall cost - each tab added 5000 units to the case overall cost. The relationship was found to be linear and, as expected, when no tabs were present the additional cost impact was zero. The increasing cost as the number of tabs increases can clearly be attributed to the extra material and machining.

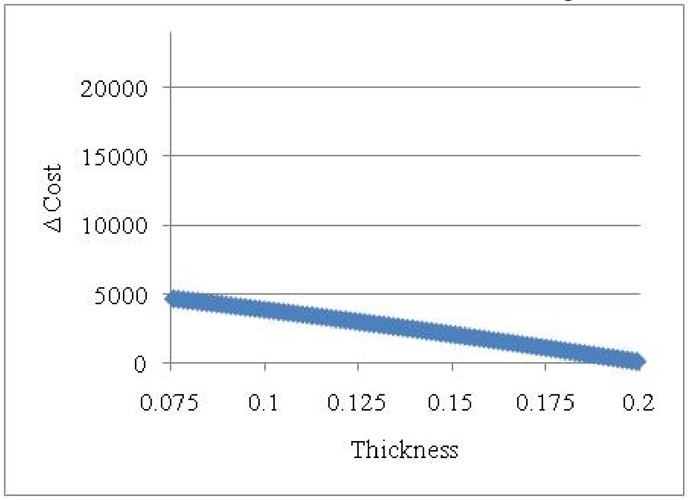


Figure 12 Case sensitivity for thickness

The number of variable vane stages also significantly influenced a case overall cost. As the number of stages increased, the cost impact approached a near linear rate. This cost increase can be explained by the extra material build up required around each variable vane hole set and the machining required when creating the holes.

## 5. DISCUSSION

### 5.1. Neural networks for cost estimation

The neural network approach to estimate the cost of jet engine components appears to have less variability and greater accuracy than regression methods. When compared to linear regression, the neural networks produces a 30% higher ***R*** value for shafts and 90% higher ***R*** value for cases. This significant improvement over linear regression can be attributed to the neural network ability to handle complex data sets with many inputs and few data points.

Compared to the current cost estimation method, the neural network produces a cost estimate with a 4.7% higher ***R*** value for shafts and a 5% higher ***R*** value for cases. This result was achieved in a fraction of the time required by the cost estimation software and with relatively little knowledge of the parts being produced or the underlying processes used to produce them. This improvement in ***R*** value came at the expense of transparency. The cost estimation software provides complete transparency, allowing the underlying cost drivers to be individually examined and altered.

### 5.2. Knowledge extraction

Sensitivity analysis provides a better understanding of how the neural networks derive their cost estimates. The sensitivity results show the maximum impact each variable had on the overall cost and how each attribute's impact varies over its range of possible values. The greatest cost impact for shafts involves the flange type, presence of radial holes, and manufacturing location. For cases, the axial flange type, number of attachment rings, and abradable attributes have the greatest cost impact. This sensitivity information provides reassurance of the cost estimate produced by the neural network and serves as a starting point for generating cost estimate relationships for cost estimation programs.

### 5.3. Data expansion

Due to the limited amount of data and the large number of attributes used to model each part family, data expansion techniques were examined. Doubling the data set and then randomly assigning the parts to either training, cross validation, or testing provides the best ***R*** values for all three data sets with cases improving by 23.6%. The dramatic improvement, especially in cases when using the doubled data set, raises questions as to possible over-fitting and memorization by the neural network.

To test for memorization, production data sets chosen from the large shafts database are run through the best network produced by each data expansion technique. The doubled data network produces 10% more error then the base data network and 17% more error then the doubled data with holdout network. This shows that a neural network trained on doubled data can be affected by over learning even when cross validation is used. To avoid over learning, a holdout testing data set should be used when determining the results of a network using doubled data. This produces more data and a generalized network compared to networks trained on the base data.

The final data expansion technique examined was the creation of new data points. To create new data points, the original data points are sorted into ranges based on their actual cost. Created part values are assigned based on equations and probabilities derived from the original data points found in that particular range. For every original part in a given range, 100 new parts are created. Twenty percent of the real data points were placed directly into the testing data set, with the remaining real data points and all the created data points being randomly assigned to a data set.

The networks trained on the created data set for shafts and the Saginaw created data set provided an ***R*** value significantly lower than the other data expansion techniques and slightly lower than the base data set. The network trained on the created data set for cases provided an ***R*** value slightly lower than the other data expansion techniques, but has a 14% improvement in ***R*** value compared to the base data set.

## 6. CONCLUSION

Neural networks are an excellent option for cost estimation even when the relative amount of input data available is limited. This research showed that a neural network was able to achieve a superior ***R*** value when compared to the current cost estimation software. A drawback to the use of neural networks is that their black box nature may obscure the reasoning behind the estimate. Even with the use of sensitivity analysis, neural networks do not display the underlying processes and cost drivers as well as the current cost estimation software. For situations where transparency is an issue, neural networks can be used as a starting point for the creation of cost estimate relationships in the current cost estimation method. In situations where a project is at the innovation stage and multiple iterations are being considered, neural networks present a rapid approach toward cost-efficient design.

## REFERENCES

[1] Divelbiss D. (2005), Evaluation of the impact of product detail on the accuracy of cost estimates; Thesis, Ohio University; Athens, Ohio.

[2] Fine T. (1999), Feedforward Neural Network Methodology; *Springer Series in Statistics*; 1st edition, Springer.

[3] Günaydin, H. Murat, Doğan, S. Zeynep. (2004), A neural network approach for early cost estimation of structural systems of buildings; *International Journal of Project Management* 22; 595-602.

[4] Kim K., Han I. (2003), Application of a hybrid genetic algorithm and neural network approach in activity based costing; *Expert Systems with Applications* 24; 73–77.

[5] Layer A., Brinke E.T., Houten F., Kals H., Haasis S. (2002), Recent and future trends in cost estimation; *International Journal Computer Integrated Manufacturing* 15(6); 499-510.

[6] Millie D., Weckman G., Pigg R., Tester P., Dyble J., Litaker R.W., Hunter J.C., Carrick H.J., Fahnenstiel G.L. ( 2006), Modeling phytoplankton aAbundance in Saginaw Bay, Lake Huron: Using artificial neural networks to discern functional influence of environmental variables and relevance to a Great Lakes observing system; *Journal of Phycology* 42(2); 336-349.

[7] National Aeronautics and Space Administration (2004), NASA Cost Estimation Handbook; Washington.

[8] Prechelt L. (1998), Automatic early stopping using cross validation: quantifying the criteria; *Neural Networks* 11; 761–767.

[9] Principe J.C., Euliano E.R., Lefebvre W.C. (1999), Neural and adaptive systems: Fundamentals through simulations with cd-rom; John Wiley & Sons; New York.

[10] Schenker B., Agarwal M. (1996), Cross-validated structure selection for neural networks; *Computers and Chemical Engineering* 20(2); 175-186.

[11] Seo K.K., Park J.H., Jang D.S., Wallace D. (2002), Approximate estimation of the product life cycle cost using neural networks in conceptual design; *International Journal of Advanced Manufacturing Technology* 19; 461-471.

[12] Shlub A., Versand R. (1999), Estimating the cost of steel pipe bending, a comparison between neural networks and regression analysis; *International Journal of Production Economics* 62; 201-207.

[13] Smith A.E., Mason A.K. (1997), Cost estimation predictive modeling: Regression versus neural network; *The Engineering Economist* 42(2); 137–161.

[14] Swingler K. (1996), Applying neural networks, A practical guide; 3rd edition, Morgan Kaufmann.

[15] Walpole, Myers, Myers, Ye. (2002), Probability & Statistics for engineers & scientists; 7th edition, Prentice Hall.

[16] West D. (2000), Neural network credit scoring models; *Computers & Operation Research* 27; 1131-1152.

[17] West D., Dellana S., Qian J. (2005), Neural network ensemble strategies for financial decision application; *Computers & Operations Research* 32; 2543-2559.

[18] Yamin H.Y., Shahidehpour S.M., Li Z. (2004), Adaptive short-term electricity price forecasting using neural networks in the restructured power markets; *Electrical Power and Energy Systems* 26; 571-581.

[19] Yuval (2000), Neural network training for prediction of climatological time series, regularized by minimization of the generalized cross-validation function; *Monthly Weather Review* 128(5); 1456-1474.

[20] Zhang Y.F., Fuh J.Y.H. (1998), A neural network approach for early cost estimation of packaging products; *Computers and Industrial Engineering* 34(2); 433-450.