There are several advantages to using regression trees. Visually, they are very easy to understand. They can be presented graphically and interpreted easily by non-technical reviewers. Regression trees can be used in preliminary data exploration to understand the most significant variables within a dataset. Pairwise analysis combined with regression trees can help shorten the time running regression models in search of significant relationships. Regression trees can handle both numerical and categorical variables. Also, since this method is nonparametric, it does not rely on data belonging to a particular type of distribution.

There are disadvantages to regression trees. As with KNN, overfitting is also a common issue. To remedy this, constraints must be defined on tree size which can be set in R and Python. Another disadvantage is despite the methods ability to handle both numerical and categorical variables, decision trees can lose information when it categorizes continuous variables in different categories.

Random Forests

One of the flaws of decision trees is that they tend to overfit the data leading to a confusion of noise with signal. To correct for this, an ensemble approach is used to create a random forest. This approach combines the estimates of multiple trees to produce an average. Random forests have been proven to provide better prediction ("wisdom of the crowd" effect). They are also more stable (robust to small amounts of noise). However, since the predictions are rather complex, there is no single equation or CER, which may make communicating the model to management challenging.

We applied to our data set – used 500 trees. We used random sampling of the dataset to randomly generate subsets of trees (bootstrap). Final prediction is an average of the result of 500 individual trees (bootstrap + aggregating = bagging).

Out-of-sample testing indicates that the mean squared error is 38% and the model explains 58% of the variation in the data. Using our example, the estimate is $369 million.

Support Vector Machines

Support vector machines originated in the 1990s in the field of optical character recognition, where it was very successful. The basic idea for classification is to maximize the margin between classes, which yields maximally robust classification. To apply to continuous output, the analogous idea is to find an equation that is:

- As "flat" as possible, i.e., the coefficients are as small as possible
- Emphasis on sparseness, parsimony
- Makes model less sensitive to errors in inputs
- Minimizes the residuals that are outside a specified range of the estimate (☐-insensitive), e.g., 15%

For a linear equation $\boldsymbol{Y = a+bX}$, with $\boldsymbol{n}$ data points the problem becomes

$$\boldsymbol{Minimize}: \frac{\mathbf{1}}{\mathbf{2}}(\boldsymbol{a}^{\mathbf{2}} + \boldsymbol{b}^{\mathbf{2}}) + \boldsymbol{C} * \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{n}} \boldsymbol{\delta_i}$$

$$\boldsymbol{Subject\ to}\ |\boldsymbol{y_i} - \boldsymbol{a} - \boldsymbol{bx_i}| \leq \boldsymbol{\varepsilon} + \boldsymbol{\delta_i}\ \boldsymbol{for\ all}\ \boldsymbol{i} = \mathbf{1}, \ldots \boldsymbol{n}$$

where the delta values are non-negative, and the loss function is insensitive to residuals less than □□ (user specified), and a weight equal to *C* is given to the errors (controls for degree of parsimony). For example of insensitive losses, for a $10 million project, you may not care about the residual as long as it is no larger than $1 million.

Given a nonlinear equation $\boldsymbol{Y = aX^b}$, take log transforms of the data and apply the linear support vector set up. The insensitivity is now in log-space – the log of the differences between the actual and the estimate.

As an alternative to logarithmic transformation, you can apply the same notion to the absolute value of percentage difference between the actuals and the estimates, i.e.

$$Minimize: \frac{1}{2}(a^2 + b^2) + C * \sum_{i=1}^{n} \delta_i$$

$$Where\ \delta_i = \left\{ \begin{matrix} \left|\frac{y_i - ax_i^b}{ax_i^b}\right| - 0.15\ if\ \left|\frac{y_i - ax_i^b}{ax_i^b}\right| \geq 15\% \\ 0\ \ otherwise \end{matrix} \right\}\ for\ i = 1, \ldots n$$

For solving this optimization problem, can use Excel's Solver capability. We use this latter formulation to apply to our example problem. We set the □-insensitivity to 15% and *C* to a low value to emphasize parsimonious models.

The CER is

$$0.33 * Weight^{0.66} Schedule^{0.57} Design\ Life^{0.23} 1.20^{Extensive\ Testing}$$

The Pearson's R^2 is equal to 73%. For the 47 data points, the equation is within 15% of the actuals for 17 (36% of the total number of data points), and for the other 30 data points, the average error is 53%. The estimated Bus Cost is $467M.

Text Analytics

Text analytics, or text mining, is supervised learning method that explores large quantities of textual data and finds patterns. Imagine having the capability of reading hundreds of pages of information in seconds to obtain information about one important topic. Text analytics provides this capability.

Text analytics provides the ability to process large amounts of structured or unstructured data from different mediums and output data related to a specific topic. This method seeks to find correlations between multiple documents, groups of words and single words. The text analytic tools search and scan data from documents, websites, databases and other data repositories and scans the data providing analysts the ability to search and explore relationships. The main goal is to break down large quantities of data into smaller, more manageable chunks to facilitate analysis.

Advantages of text analytics are obvious. Having the ability to scan and dissect large amounts of data quickly is invaluable to analysts. This method not only can find information on a specified topic but can also understand nuances in language. A disadvantage is that emotion cannot be captured during the review of text with this method. Without understanding tone and intent, sometimes text can be misinterpreted and information misused.

For this paper, text analytics is being introduced but not explored using data.

Conclusion

Traditional regression analysis is a tried-and-true method for cost estimating. The promise of big data and machine learning brings additional methods worth considering, including k-nearest neighbor, neural networks, regression trees, random forests, and support vector machines.

K-nearest neighbors is truly a non-parametric method. It is worth trying if you cannot find meaningful parametric relationships. The authors have found that this method came in very handy on a recent project.

Neural network is an artificial intelligence technique that has been successfully used in cost estimating in the past (see the references for examples). It is prone to overfitting, especially for small data sets. It was a big buzzword in the 1990s and is the basis for deep learning, a current hot topic in data science. However, before we can use deep learning in cost estimating, we need much bigger data sets.

Regression trees provides a different look at the data, in a tree format, which is useful in and of itself. However, it is prone to overfitting, and the results can be nonintuitive.

Random forests bring robustness and stability to the regression tree methodology. However, this method is largely a black box, and is there is no single tree or equation that produces the estimate. This can make the communication of this method to management a challenge.

Support vector machines are the closest in form to traditional regression and bring to it concepts that make regression more robust and less prone to error. The method is easy to implement in a spreadsheet using Excel Solver.

When using the same inputs, the methods produced the following Bus Costs:

- KNN – $750M
- Neural Networks – $963M
- Regression Trees – $464M
- Random Forests – $369M
- Support Vector Machines – $467M

The methods produce results that range from $369M to $963M. It is interesting to see how, when using the same dataset, the different methods produce results that vary

drastically. It is worth exploring further why the Neural Networks algorithm predicted costs must higher than the other methods.

R code used for the methods in this paper is available upon request.

References

1. Anderson, C., "The End of Theory: The Data Deluge Makes the Scientific Method Obsolete," Wired, June 2008.

2. Davenport, D.H., and D.J. Patil, "Data Scientist: The Sexiest Job of the 21st Century," Harvard Business Review, October 2012

3. Dean,E., "Neural Network Cost Estimating Relationships," Proceedings of the 2010 ISPA-SCEA Conference, San Diego.

4. Hutchings, C., "An Approach Towards Determining Value Through the Application of Machine Learning," Proceedings of the 2018 ICEAA Conference

5. Kaluzny, B.L, "An Application of Data Mining Algorithms for Shipbuilding Cost Estimation," Proceedings of the 2011 ISPA-SCEA Conference

6. Mourikas, K., J. King, and D. Nelson, "Machine Learning Approach to Cost Analysis," Proceedings of the 2017 ICEAA Conference.

7. Newell, A. and P.S. Rosembloom, "Mechanisms of Skill Acquisition and the Law of Practice," in R Anderson(Ed.), Cognitive Skills and Their Acquisition, Hillsdale, NJ, Erlbaum.

8. Pincus, J., and O. Akbik, "Social Media and Submarines: How Machine Learning and Unconventional Methods Can Change Cost Estimating," Proceedings of the 2018 ICEAA Conference.

9. Rao, Venky, "Introduction to Classification & Regression Trees", January 2013

10. Singh,Aishwarya, "A Practical Introduction to K-Nearest Neighbors Algorithm for Regression", August 2018.