Identification of Risk Factors in Globally Outsourced Software Projects using Logistic Regression and ANN

R. P. Mohanty, G. Sahoo, J. Dasgupta*

Vice Chancellor, SOA University, Bhubaneswar 751030, India
rpmohanty@gmail.com

Head, Department of Information Technology
Birla Institute of Technology, Mesra, Ranchi 835215, India
drgsahoo@yahoo.com

Ph. D. Scholar, Department of Management
Birla Institute of Technology, Mesra, Ranchi 835215, India
jyotidasgupta@yahoo.co.in

(*Corresponding author)

Abstract— **Global supply chains are often critically dependent upon globally outsourced Information Technology (IT) and Business Process Outsourcing (BPO) projects. Effective risk assessment and mitigation of these projects is therefore of great importance for such supply chains. Traditional risk management techniques used so far in IT and BPO projects have depended almost entirely on the 'Expected Utility Theory' that computes risk exposure as the product of risk probability and risk impact. Although this method is considered the gold standard in risk assessment, it has severe limitations due to the fact that accurate computation of risk probability and impact is difficult. In this and earlier papers, we have advocated the use of 'risk factors' in conjunction with other existing methods for risk assessment. Risk factors are conditions that affect project performance. In this paper, we use Logistic Regression and Artificial Neural Network (ANN) based methods on a set of globally outsourced projects to predict project performance and to rank the relative importance of project risk factors. Although in this paper these techniques have been tested in outsourced IT projects, these can also be used in identifying and determining project risk factors in other industries.**

Keywords— *Outsourced projects, Risk Management, Risk Factors, Logistic Regression, Artificial Neural Networks*

1. Introduction

Supply Chain Management (SCM) is needed for various reasons: improving operations, better outsourcing, increasing profits, enhancing customer satisfaction, generating quality outcomes, tackling competitive pressures, increasing globalization, increasing importance of E-commerce, and growing complexity of supply chains. SCM helps the business organization to compete in the dynamic global market. The goal of SCM is to integrate activities across and within organizations for providing the customer value [1].

Further, Lee *et. al.* [2] have reported that outsourcing is very commonly used by organizations to reduce supply chain management costs. However, as outsourcing has moved from peripheral to more vital functions, it leads to myriad risks [3]. In global supply chain, these risks may lead to disruptions of the supply network.

Outsourcing of both development as well as operations of supply chain IT systems is quite common. In fact, this has led researchers to experiment with the possibility of executing IT/ BPO projects themselves as supply chains [4].

Unlike software projects undertaken in-house, global software services outsourcing industry run hundreds and often thousands of software projects concurrently. Also – outsourced projects are executed by vendors on behalf of their clients who are the actual project owners. These differences lead to additional risks that must be managed for projects to succeed. Current 'risk management' best practices prescribed by various standards such as the ones from the International Standards Organization (ISO), SEI CMMI model from the Software Engineering Institute at Carnegie Mellon University, the Project Management Body of Knowledge from Project Management Institute; as well as bulk of the research conducted until now focus on managing risks for individual projects using various techniques derived from the 'Expected Utility Theory' [5]. Our research has instead focused on risk management for a group of projects, and more specifically for the outsourced ones. This has led us to propose a new term named 'project risk factors' that can be identified by examining a group of projects. These 'risk factors' increase or decrease the chances of project success or failure, and are similar to the risk factors one sees in other areas such as medicine (risk factors for coronary thrombosis for example) or risk factors for credit card defaults [5] [6] [7]. In this paper, we present two methods based on Logistic Regression and Artificial Neural Networks to identify and rank project risk factors. These methods have been tested and validated in a mid-sized global software services outsourcing organization.

2. Literature Review

Faced with the task of creating an index that could be used to decide which prisoners could be granted parole, John Copas [9] devised a 'Risk of Reoffence' (ROR) index that attempted to predict the probability of the paroled prisoner committing another criminal act within time t, say 3 months to 2 years. This is a classic example for risky decision-making, or decision making under uncertainty. Notice that the prisoner in question may exhibit one or more of several risk factors, which have been observed to have some correlation with re-offence. So while the outcome or dependant factor is a single outcome, that of re-offence, the factors we can base our judgment on are several. Further we are not aware if the factors themselves have some correlation between themselves or not.

We can liken this situation with that of a software project. If we have observed over a period of time that the presence or absence of some factor, say that of unclear requirements, has an effect (ideally causative effect) on project failure, and can find several other such factors, we could use a statistical technique for computing the probability that the project may fail by using 'logistic regression'.

Logistic regression has a sound mathematical and statistical foundation, and uses a theorem in mathematical statistics, the Neyman-Pearson Lemma. For any given combination of values of the risk factors x_i, the model allows us to evaluate two probabilities, P $(x_1, x_2, \ldots, x_n | E)$, the probability of this combination of risk factors occurring amongst cases when event E happens, and P $(x_1, x_2, .., x_n | E')$, the same probability but among cases when E does not happen. The theorem then asserts that the best risk score is:

$$S (x_1, x_2, \ldots, x_n) = \log \quad P (x_1, x_2, \ldots, x_n | E) / P (x_1, x_2, .., x_n | E') \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

This method of risk assessment is consequently one that is derived from observed pattern of data using logically argued scientific principles.

One interpretation of the word 'best' used in describing the above score is in terms of the 'false positive' and 'false negative' rates of a decision instrument which decides that E occurs if $S>k$ and that E does not occur if $S\leq k$, for some fixed threshold value of k. Of course this decision instrument would sometimes be right and sometimes be wrong. The false positive rate is the chance that $S>k$ given that E has not in fact occurred, and the false negative rate is the chance that $S\leq k$ given that E has occurred. These two error rates cannot both be minimized, since adjusting threshold k can only reduce the size of one at the expense of the other. However, of all possible decision instruments based on these risk factors, the score S is the one which gives the decision instrument which minimizes the size of either one of these error rates a fixed value of the other [9].

It does appear therefore, that in the context of a software project, where many different factors cause projects to fail, a risk score based on above principles would provide following advantages:

a) Sound mathematical and statistical base, overcoming limitations of other methods which are rather ad hoc in nature specially when it comes to aggregating risks
b) The inter-correlation between risk factors can be allowed under this model
c) The resulting risk score gives an assessment of risk for all possible values of the risk factors. Without a model, there would never be enough data to assess each of these combinations separately.
d) With advances in computing one can use tools such as Decision Tools Suite from Palisade, the tool used in this paper [8] or other tools such as StatTools, SPSS, SAS, R, or GRETL to handle large number of factors and data easily.

For example, if the risk factors identified are X_1, X_2 and X_3, and the probability of the project failing is

P, by using any of the above tools we can find a logistic regression equation of the following form

$$\mathbf{P = [e^{(a+K1X1+K2X2+K3X3)}] \; / \; [1 + e^{(a+K1X1+K2X22+K3X3)}] \ldots\ldots (2)}$$

Where the values a, K_1, K_2 and K_3 can be computed using one of these tools and specifying either 'Enter' or 'Stepwise' method depending upon sample size and other considerations [10]

Further literature review established that concept of risk factors is not at all new in non-software areas. For example, a recent study [11] estimated the percentage of cancers (excluding non-melanoma skin cancer) in the UK in 2010 that were the result of exposure to 14 major lifestyle, dietary and environmental risk factors: tobacco, alcohol, four elements of diet (consumption of meat, fruit and vegetables, fiber and salt), overweight, lack of physical exercise, occupation, infections, radiation (ionizing and solar), use of hormones and reproductive history (breast feeding). In another study reported in the book Outliers [12] researchers Dr. Stewart Wolf and sociologist John Bruhn looked at a community of Italian immigrants at Roseto, Pennsylvania, USA that never reported any cardiac arrest cases – and found that leading less stressful life and strong family and community bonding as well as support reduces cardiac arrest incidents. So they found factors that reduce the risk of heart attack. In Freakonomics [13] Levitt & Dubner report factors that determine the financial well being of drug dealers, or the possibility of wins by Japanese sumo wrestlers!

We therefore hypothesized that it should be possible to examine a group of projects (as opposed to a single project) and identify 'risk factors' that increase or decrease the chances of a project meeting all project objectives such as cost, schedule, quality etc.; and estimate impact and/or sensitivity of these various risk factors as well.

Review of ANN literature revealed widespread use of ANN in addition to logistic regression for identification of risk factors. ANN research experienced its first peak of activity in the 1940s, followed by extensive activities in the 1960s. There was a lull of almost twenty years until 1980s when interest in ANN again picked up [14]. In the recent past, availability of powerful but easy to use ANN packages on PC has revived interest in ANN based methods in diverse areas. Their use however have been limited to relatively simple problems, as the most advanced ANN methods also do not compare well with certain tasks that humans do with ease. Recently – Queral Networks have been proposed to overcome ANN limitations [15].

Neural networks estimate functions from sample data. Statistical approaches also estimate functions. For each problem, statistical approaches require that we guess how outputs functionally depend on inputs. Neural systems do not require that we articulate such a mathematical model. They are model-free estimators in that sense [16].

Artificial neural systems, or neural networks, are similar to physical cellular systems which can acquire, store, and utilize experiential knowledge [17]. Artificial neural networks (ANN) are created using artificial neurons, which are information-processing units. Further discussions on ANN would not be attempted in this paper and interested readers are invited to refer to several very good books available on the subject, of which one was extensively referred to by the authors. [18].

Artificial Neural Networks have been used in predicting future states and outcomes in many areas. Quality performance of construction projects has been predicated using ANN [19]. Maciulis [20] used ANN for foreign exchange hedging. Okoroh *et. al.* [21] used ANN to model risk-management in healthcare facilities. Ko *et.al.* [22] reported superior prediction capability using a hybrid AI approach that fuses genetic algorithms, fuzzy logic and ANN.

In risk management area, Palaneeswaran [23] have reported rework causes in construction projects, and how these can be predicted and controlled using ANN. It is very interesting to note that the rework causes in construction industry are remarkably similar to those encountered in software projects. Skorupa [24] used neural networks for mechanical equipment failures. Al-Mutairi *et. al.* [25] have used ANN successfully for predicting accident incident rates.

Although we did not find evidence that ANN has been used to predict outsourced software projects risks, we found that the most relevant research from outsourced projects risk management view was found from the field of medicine. A research team led by Dr. G. Sahoo along with Dr. D. Shanthi and Dr. N. Saravanan reported significant progress in prediction of Thrombo-embolic stroke [26]. This was achieved through use of ANN on 25 patient risk factors such as age, Sex, Pre-stroke Mobility, Hypertension, Diabetes Mellitus, Myocardial Infarction, Cardiac Failure, Atrial Fibrillation, Smoking, High Blood Cholesterol, Alcohol abuse, Weakness of left arm and left leg, Weakness of right arm and right leg, Slurring of speech, Giddiness, Headache, vomiting, Memory deficits, Swallowing difficulties, Loss of vision, Isolated vertigo, Transient double vision, Sudden difficulty in walking dizziness or loss of balance, Hand/ Leg numbness, Transient loss of consciousness. The team used an ANN tool named Neuro-Intelligence. Data from 50 patients were collected, and ANN techniques were used to predict the likelihood of a patient getting a stroke. The results were found to be quite accurate.

3. Research Methodology

Research methodology used in conducting the experiments reported in this paper is described below.

a) Literature review to find applicable statistical or other methods that can be used to identify risk factors. This step revealed several methods that researchers have used earlier in other (non software) fields. Out of these, the ones that could also be used for software projects were reported by the authors earlier [5] [6] [7]

b) Review of Logistic Regression and ANN Literature to find whether these may be used in risk management of outsourced projects

c) Establish initial hypothesis on possible risk factors for software projects (Table 1) based on our own experience, as well as discussions with other researchers and practitioners

d) Collect risk factor and project status data across 118 outsourced projects

e) Set up Logistic Regression and ANN tool [8]

f) Perform Logistic Regression and ANN sensitivity analysis to identify the risk factors with most impact on project success or failure

g) Report results, and discussions

h) Report limitations and scope for further research

Table 1: Outsourced Project Risk Factors (Hypothesis)

Data Name & Abbreviation	**Possible Values**	**Remarks**
Project Number	1 to any number	Unique Identification for the project
Project Type	D/ M/ T	For Development, Maintenance or Testing type projects respectively
Client Age	New/ Old	New means vendor has less than 1 year experience in working with this client
Client Project Manager Age	New/ Old	New means client project manager has less than 1 year experience in working with this project
Vendor Project Manager Age	New/ Old	New means vendor project manager has less than 1 year experience in working with this client project
Peak Project Team Size	1 to any number	Peak staffing strength of the project
Billing Type	FP/ TM	Billing is to be done on Fixed Price or Time & Material basis
Location	New/ Old	New means vendor has less than 1 year experience in operating at this client location

Penalty Clause	Y/ N	Y indicates presence of penalty causes in the contract
Special Infrastructure	Y/N	Y indicates that special hardware or software needed for the project
New Technology	Y/N	Y indicates that more than 50 % of the team team members had less than 1 year experience with 1 or more technologies needed for the project
Rapid Ramp- up	Y/N	Project was forced to add more than 10% new members in the team in any month after the project started due to any reason including attrition
New Business Domain	Y/N	Y indicates that more than 50 % of the team members had less than 1 year experience in the business domain (such as Insurance, Life Sciences etc.) needed for the project

4. Results

a) Logistic Regression: Using StatTools 6 from Palisade Corporation [8] and data from 118 outsourced projects as described earlier, Logistic Regression was performed on the dataset containing data on the hypothesised risk factors. Model summary is given in Table 2. As we can see, p – value is very low which means that model has good explanatory power.

Table 2: Logistic Regression Model Summary

Logistic Regression for Project Status	
Summary Measures	
Null Deviance	149.7510023
Model Deviance	55.27236381
Improvement	94.47863847
p-Value	< 0.0001

Table 3, summary classification also confirms model validity with high percentage of correct predictions.

Table 3: Summary Classification

Summary Classification	**Percent**
Correct	90.68%
Base	66.95%
Improvement	71.79%

Table 4, classification matrix also shows high correct percentage for both failed and successful projects, supporting model validity.

Table 4: Classification Matrix

Classification Matrix	**1**	**0**	**Percent Correct**
1	76	3	96.20%
0	8	31	79.49%

Regression coefficients for the risk factors are shown in Table 5.

As we showed in Equation 2, the coefficients can be used to predict the performance of any project using the below method:

Risk Score of project P_i (RSP_i) = Constant Coefficient + Value of Factor 1 for project$_i$ x Coefficient for Factor 1 + Value of Factor 2 for project$_i$ x Coefficient for Factor 2+ …. + Value

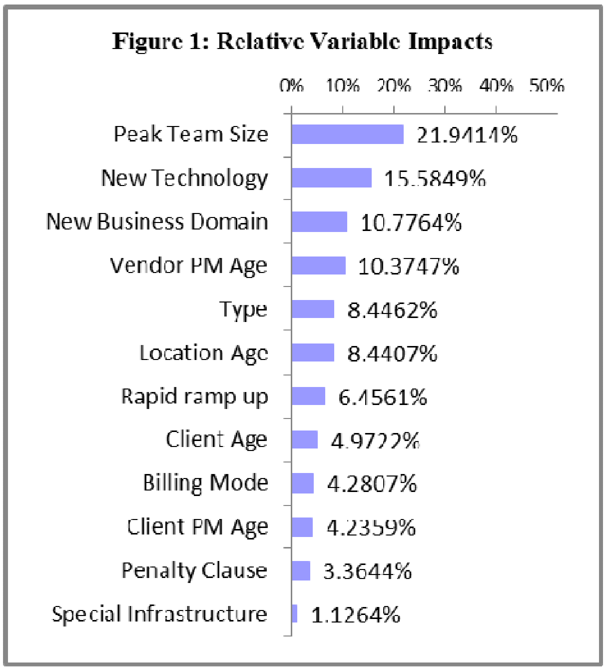
of Factor n for project$_i$ x Coefficient for Factor n ….. (3)

Hence, it follows that the probability of Project$_I$ being successful is given by P below

$$P = e^{RSPi} / 1 + e^{RSPi} \quad …….. \quad (4)$$

b) Artificial Neural Networks (ANN): Using Neural Tools 5.5 from Palisade Corporation [8] AND Multi-layer Feed forward Network with 4 nodes, the dataset of these 118 projects was trained and tested. The findings are given in Table 6.

Impacts of independent variables (hypothesized risk factors) are shown in Figure 1



.

Figure 1: Relative Impacts

As we can see from Figure 1, highest impact for failed projects came from risk factors Peak Team Size, New Technology, New Business Domain, and vendor PM age. The least impact came from special infrastructure. Other risk factors had impacts between these.

5. Discussions

Using risk factors and project performance data from 118 outsourced software projects, and applying Logistic Regression and ANN techniques on the data set, we have shown how these techniques can be used to:

1. Test hypothesis about risk factors in projects
2. Predict the probability of a project's success, and
3. Rank the relative impact of various risk factors on the project's success

We do not make any comparisons between these two complimentary techniques. As one can see from the results – used together, these two techniques can be of immense help to project managers for assessing and mitigating risks.

These findings can usher in major changes in the way risks are managed in the software outsourcing industry, which constitutes an important part of most global supply chains. Current state of the art in this industry draw upon the models propounded by organizations such as the Software Engineering Institute of the Carnegie Melon University – popularly known as the SEI CMMI model which is also mandated by the US Department of Defence [27] or the ISO 31000: 2009 [28] standard for risk management do not have the concept of risk factors. This is a major shortcoming of these models in our opinion, as the concept of risk factors is well understood and entrenched in several fields including medicine, insurance, credit or finance.

As we have reported in our earlier papers [5] [6] the gold standard in risk assessment remains the following equation which is based on the expected utility theory.

$$RE = Prob\ (UO)*Loss\ (UO).\ ...\ (5)$$

Where, Prob (UO) is the probability of an unsatisfactory outcome,

And, Loss (UO) is the loss to the parties affected if the outcome is unsatisfactory.

However, accurate estimation of Prob (UO) is a challenge in most real life situations, rendering this technique of limited use in many situations. This is true in other fields, such as medicine as well. While researchers in those other fields have been identifying the risk factors for the last several years, the concept is yet to find adoption by researchers and practitioners in the software industry. We believe that the results reported in this paper would therefore be of great significance to the software industry, and by association – for global supply chains.

6. Conclusion:

a) Contributions of this paper: Although both logistic regression and ANN techniques have been used earlier in prediction of project performance in various industries, it had not been used for outsourced software projects. The techniques and findings described in this paper would thus be very useful to practitioners and researchers.

b) Limitations and further research: The dataset contained data from only 118 projects completed in the last two years, executed at several locations of an outsourcing vendor. Although fairly representative, a larger dataset obtained from more than one company and across more project locations may offer further insights. Also, underlying causes for above relative impacts may be analysed, and root causes determined to explain the findings.

Based on above findings, special processes were designed in the organization to better execute projects with very large team size. This resulted in significant improvement in project performance at this organization. Further research and corroboration over more time and projects would further validate efficacy of these methods.

Recently – few newer techniques combing Fuzzy and Logistic Methods [29] or other methods such as Logic Regression, Random Forest or Bayesian Logistic Regression [30] have been used in ascertaining risk factors. Efficacy of these methods in ascertaining software projects risk factors can also be further investigated.

Acknowledgments

The authors would like to thank the anonymous reviewers for their review comments which added a lot of value, and improved the organization of this paper. The authors would also like to thank the organization where these experiments were conducted.

References

[1] Dr. Md. Mamun Habib, *"An Exploratory Study of Supply Chain Management for Tertiary Educational Institutions"*, IEEE Int'l Technology Management Conference, IEEE, San Jose, USA, pp.1112 - 1119, 2011.

[2] CKM Lee, YC Yeung, Z Hong, "*Managing the Risks of Outsourcing in Supply Chain Networks"*, IEEE Int'l Technology Management Conference, IEEE, San Jose, USA, pp.507 - 513, 2011.

[3] Liu xun, "*Analysis On Risk Management In Supply Chain Based on the CDS Model"*, ETP International Conference on Future Computer and Communication, IEEE Computer Society, USA, pp. 205 – 208, 2009

[4] Gautham Reddy N, "*Designing software project management models based on supply chain quality assurance practices"*, 2009 World Congress on Computer Science and Information Engineering, IEEE Computer Society, USA, pp. 659 – 663, 2009

[5] Dasgupta, J., Mohanty, R. P., "*Estimating operational risk indices for software services outsourcing industry: a case"*, Int. J. Services and Standards, Vol. 6, No. 1, pp. 43 – 61, 2010

[6] Dasgupta, J., Mohanty, R. P. Towards formulation of software project risk radars, *Int. J. Business Information Systems*, Vol. 6, No. 2, pp. 151 – 178, 2010

[7] Dasgupta, J., Mohanty, R. P., Sahoo, G. Monte Carlo Simulation based estimations:

Case from a Global Outsourcing Company, *Proceedings of the 1st IEEE International Technology Management Conference, San Jose, USA*, pp. 750 – 755, 2011

[8] Decision Tools Support Suite 5.5 and 6, www.palisade.com, 02-05-2012

[9] Copas, J., "*Statistical Modelling for Risk Assessment*", Risk Management: An International Journal, Vol. 1, No. 1, pp: 35-49, 1999

[10] Nargundkar, R.,"*Marketing Research, Text and Cases"*, Tata McGraw-Hill, New Delhi, 2008

[11] Parkin, D. M., "The fraction of cancer attributatble to lifestyle and environmental factors in the UK in 2010", *British Journal of Cancer* 105, S2 – S5, doi:10.1038/bjc.2011.474, www.bjcancer.com, published online 6 December 2011

[12] Gladwell, Malcolm, Outliers, Penguin Books, England, pp. 3 – 11, 2008

[13] Levitt, Steven D. and Dubner, Stephen J., Freakonomics, Penguin Books (England, 2006)

[14] Jain, A. K., Mao, J., Mohiuddin, K. M., "*Artificial Neural Networks: A tutorial"*, Computer, IEEE, pp. 31-44, March, 1996

[15] Hoffman, T. A., Rozenblit, J. W., Akoglu, A., Suantak, L. "*Queral Networks: Towards an approach for engineering large Artificial Neural Networks"*, Proceedings of the 18th IEEE International Conference and Workshops on Engineering of Computer Based Systems, pp. 81 – 88, 2011

[16] Kosko, B., "*Neural Networks and Fuzzy Systems, A Dynamical Systems Approach to Machine Intelligence"*, Prentice-Hall India, New Delhi, 2005

[17] Zurada, J. M., "*Introduction to Artificial Neural Systems"*, Tenth impression, Jaico Publishing House, Mumbai, 2006

[18] Haykin, S, "*Neural Networks, A Comprehensive Foundation"* Dorling Kindrsley (India) Pvt. Ltd., 2006

[19] Jha, K. N., Chockalingam, C. T., "*Prediction of quality performance using artificial neural networks, Evidence from Indian construction projects*", Journal of Advances in Management Research, Vol 6, No 1, pp. 70 – 86, 2009

[20] Maciulis, N., "*Foreign exchange risk hedging in the context of speculative revaluation attacks"*, Baltic journal of Management, Vol 3, No. 3, pp. 266 – 288, 2008

[21] Okoroh, M.I., Ilozor, B.D., Gombera, P.P., "*Modelling of risk management in health care facilities*", Facilities, Vol 24, No. 5/6, pp. 197 – 210, 2006

[22] Ko, C., Cheng, M., "*Dynamic Prediction of Project Success Using Artificial Intelligence"*, Journal of construction engineering and management, Vol. 133, No. 4, pp. 316 – 324, 2007

[23] Palaneeswaran, E., Love P.E.D., Kumaraswamy, M.M., Ng Y.S.T., "*Mapping rework causes and effects using artificial neural network"s*, Building research and information, Vol. 36, No. 5, pp: 450 – 465, 2008

[24] Skorupa, D. "*Neural Networks in the Risk Management of a Project*", 2004 AACE International Transactions, Chapter Risk.15, pp 15.1 – 15.7, 2004

[25] Al-Mutairi, A., Haight, J. M., "*Predicting Incident Rates"*, Professional safety, pp: 40 – 48, 2009

[26] Shanthi, D., Sahoo, G., Saravanan, N., "*Designing an Artificial Neural Network Model for the prediction of Thrombo-Embolic Stroke"*, International Journals of Biometric and Bioinformatics, Vol 3,No 1, pp. 10 – 18, 2009

[27] www.sei.cmu.edu/cmmi accessed on June 3, 2012

[28] www.iso.org/iso accessed on June 3, 2012

[29] Wu, S., Wu, D., Lin, S., Ma, H., "*Predicting Risk Parameters using Intelligent Fuzzy/ Logistic Regression"*, International Conference on Artificial Intelligence and Computational Intelligence, IEEE Computer Society, pp. 397 – 401, 2010

[30] Chen, C. C., Schewender, H., Keith, J., Nunkesser, R., Macrossan, P., "*Methods for Identifying SNP Interactions: A Review on Variations of Logic Regression, Random Forst, and Bayesian Logistic Regression"*, IEEE/ACM Transactions on Computational Biology and Bioinformatics, Vol. 8, No. 6, pp: 1580 – 1591, November/December 2011

Table 5: Regression Coefficients

Regression Coefficients	Coefficient	Standard Error	Wald Value	p-Value	Lower Limit	Upper Limit	Exp(Coef)
Constant	-5.477994474	17661761.87	-3.10161E-07	1.0000	-34617058.74	34617047.78	0.0041777
Type Num	0.072244445	0.56413499	0.128062337	0.8981	-1.033460135	1.177949025	1.07491807
Client Age Num	3.558027636	17661761.85	2.01454E-07	1.0000	-34617049.67	34617056.79	35.09391088
Client PM Age Num	3.172803767	395.2247314	0.008027847	0.9936	-771.4676698	777.8132773	23.87432865
Vendor PM Age Num	12.10659957	8830880.928	1.37094E-06	1.0000	-17308514.51	17308538.72	181062.8656
Peak Team Size	-0.021998683	0.026572567	-0.827871946	0.4077	-0.074080914	0.030083548	0.978241524
Billing Mode Num	-3.587713925	0.857602136	-4.18342466	< 0.0001	-5.268614113	-1.906813738	0.027661494
Location Age NUM	-2.355913108	429.0994683	-0.005490366	0.9956	-843.390871	838.6790448	0.094806898
Penalty Clause Num	-8.276037827	8830880.935	-9.3717E-07	1.0000	-17308534.91	17308518.36	0.000254544
Special Infrastructure Num	1.679106267	1.140237846	1.472592998	0.1409	-0.55575991	3.913972445	5.360762742
Rapid ramp up Num	2.963945245	0.793269151	3.736367716	0.0002	1.40913771	4.51875278	19.37425736
New Business Domain Num	1.710672887	1.002778773	1.705932487	0.0880	-0.254773509	3.676119283	5.532683096
New Technology Num	-7.670495423	8830880.925	-8.68599E-07	1.0000	-17308534.28	17308518.94	0.000466387

Table 6: Neural Net Training and Auto-Testing

Summary	
Net Information	
Name	Net Trained on Projects data (3)
Configuration	MLFN Category Predictor (4 nodes)
Location	This Workbook
Independent Category Variables	11 (Type, Client Age, Client PM Age, Vendor PM Age, Billing Mode, Location Age, Penalty Clause, Special Infrastructure, Rapid ramp up, New Business Domain, New Technology)
Independent Numeric Variables	1 (Peak Team Size)
Dependent Variable	Category Var. (Project Status)
Training	
Number of Cases	94
Training Time	0:02:40
Number of Trials	1000000
Reason Stopped	Auto-Stopped
% Bad Predictions	3.1915%
Testing	
Number of Cases	24
% Bad Predictions	25.0000%
Data Set	

Name	Projects data
Number of Rows	118
Manual Case Tags	NO
Variable Impact Analysis	
Peak Team Size	21.9414%
New Technology	15.5849%
New Business Domain	10.7764%
Vendor PM Age	10.3747%
Type	8.4462%
Location Age	8.4407%
Rapid ramp up	6.4561%
Client Age	4.9722%
Billing Mode	4.2807%
Client PM Age	4.2359%
Penalty Clause	3.3644%
Special Infrastructure	1.1264%