TEAM FACTOR EVALUATION TO ENHANCE AGILITY IN AGILE SOFTWARE PROJECT DEVELOPMENT USING MACHINE LEARNING

Laxmi Shankar Awasthi [1, 2*] and Sudhanshu Nigam [3]

[1] Ph.D. Research Scholar, Department of Computer Science and Engineering, School of Engineering and Technology, Shridhar University, Pilani-Chirawa Road, Pilani, Rajasthan- 333031 (India).

[2] Assistant Professor, Department of Computer Science, Lucknow Public College of Professional Studies, Vinamra Khand, Gomti Nagar, Lucknow - 226010 (India).

[3] Assistant Professor, Department of Computer Science and Engineering, School of Engineering and Technology, Shridhar University, Pilani-Chirawa Road, Pilani, Rajasthan- 333031 (India).

Corresponding author: E-mail: down_top@yahoo.co.in

ABSTRACT: In modern times, the software industry has revolutionized the old way of software development. The Agile approach strongly emphasizes increased delivery, reduced risk and customer satisfaction and in modern times, one of the factors in the successful completion of a project is the team factors. The amount of machine learning, performance intelligence or software engineering projects related to agile team data using the agile method agility is growing. However, there are very few studies on how such projects work. In this paper, we have analyzed agile team factors impact for the enhancement of the agility using the machine learning approach.

KEYWORDS: AGILE TEAM, FUZZY RULE BASE, FUZZY INFERENCE SYSTEM, MACHINE LEARNING, NEURAL NETWORK, TEAM SUB FACTORS.

INTRODUCTION: In Agile software development process, it is important that team is a crucial in delivering great software as outcome and that the agile growing teams are always talking about "us" instead of "Me." There is nothing more rewarding than sharing a new journey to build something really important with your participating partners [1][2]. Since there is not any formula for the straight forward selecting agile team members, agile teams share common values. Anyone can use a scrum and someone other can apply a kanban. Agile developers are interested with co-located teams, but since some business models requirement is in this way that team needs to distribute across globe. Generally agile teams have all required skill set but sometime it is required on some projects to manage specialized person for specific work. Hence it is not easy to find out whether agile team is on the path of greatness [3][4][14].When team is working , there is need to give some time to develop itself since any team works as individuals to grow. Agile teams go through four key stages as they progress [5] [7]. Working together is great experience and this experience develops trust in each member, understand each other's strengths, and use that understanding to improve the way they build software[9]. Form keeping agile team integrated, agile team needs to follow organizational discipline with acceptable rules.

When some changes are done in team like recruitment of new members, left out of member, then team should be sufficient robust to absorb the new changes[15]. Well-functioning teams work well and are built on sound engineering processes such as code review, job integration, continuous integration, and regular release cadence. We cannot stress this enough: the basics of engineering are important in building good teams. For agile teams two important pillars are considered i.e. continuous mentoring and sets of shared skills. While we work in the team, there is one big benefit can be observed that the team members learn from other members. Mentoring in team does not have benefit to learning with working, it has deep impact that everyone in the group learns from each other so that the impact of the group as a whole is greater than the total impact made by its individual members [20]. At the same time, shared skills open up the team's ability to perform a variety of tasks. As an engineer, it is always important to learn new skills because they make us more valuable to the organization and better equipped to support each other's work. It also protects the person from being a sensitive path, removing the burden from everyone's mind [21]. Team Agility is based on Lean-Thinking which makes the business decision in a lean way. It can be considered as a combination of Scrum, Kanban and eXtreme Programming based on Lean-Thinking. It is based on principles that have proved to be essential and that do not all fit into the picture. Team Agility is defined by looking at what is required to achieve at the team level and incorporating Scrum and Kanban processes where appropriate. This allows for a portable set of practices while being designed by the teams that use them. Agile team does not focus on the team only but it describes how team can work in best way for the organization. Agile team is a part of the people, the work flow and the skills needed to achieve this. Team Agility must be educated within that context. Team Agility deals with the following issues [1][13][15].

(i) Identifying that group work is part of the larger project.

(ii) Teams are components of sophisticated system and it is important how the teams deal with the rest of the organization and not focusing only on the individual team performance.

(iii) Train the team in its part of the Agile Product Management process to identify and refine those components which are to be built.

(iv) Know that we need to choose the right processes for the team and not select a set of predefined habits that try to work everywhere.

LITERATURE REVIEW: Power cannot be achieved without the use of staff knowledge and skills. Some researchers have proven that cross-training of employees is a powerful strategy that can ensure the well-being of employees. While studying staff technology in landscaping and remodeling, Payne et al. in 1990 also emphasized the role of training in staff skills. An effective training climate requires the organization to grow and develop a learning environment within the organization that can encourage people to be open and innovative in their search for new ideas. This facilitates the acquisition of knowledge and skills learning and, consequently, enhances strategic flexibility and adaptability and responds to changes in markets and workplaces. Organizations that are committed to learning develop staff and managers who are able to manage

and adapt to change. These people are more comfortable doing new and useful things [10][16]. The reward system can also be important from a staff perspective. Employee engagement processes appear to be important in making employees truly agile. While studying the contribution of employee engagement mechanisms to work harder, it has been proved that higher employee engagement processes (job enrichment; job growth; and self-governing teams) are more likely to promote employee efficiency than low-level work engagement processes [2][7][10]. The researcher noted that although low level of employee engagement processes has the potential to directly encourage employee mobility, they serve primarily as a basis. Eccles, R. in 1991 concluded that power-sharing processes provide the greatest potential for supporting the work ethic of employees, such as improving training efficiency, flexibility, multitasking and collaboration. The work environment of an organizational team is able to promote the efficiency of employees [11][12]. An appropriate team work environment promotes staff speed related to the internal workplace, the external team work. Synergy produced in multi-sectoral collaborations can assist organizations in completing projects that are taken from time to time. Computer-assisted technologies such as user interface, expert systems, team decision support system, etc. helps the organization to accept appropriate information system. Employees receive cell phone information about work data, business document and employee information, etc. [9][18][19].

PROPOSED MODEL: We have identified various team factors like Project team skill, commitment, outstanding communication, proper participation of project team with proper coordination which play important role to streamline entire energy in one direction to achieve the goal to enhance agility. Four sub factors are identified for agile team factor [1][6][8][11].

TABLE 1 : AGILE TEAM FACTORS TO ENHANCE AGILITY

S.No.	FACTORS	ABBREVIATION
1	Project Team Skills	T1
2	Project Team Commitment	T2
3	Internal Project Communication	T3
4	Participation of Project Team	T4

TABLE 2 : RULE BASE FOR AGILE TEAM FACTORS TO ENHANCE AGILITY

S.No.	T1	T2	T3	T4	T
1	H	H	H	M	H
2	L	H	H	H	L
3	M	H	H	H	H
4	L	L	L	H	L
5	H	L	L	M	H
6	H	M	M	M	M
7	M	M	M	L	M
8	M	L	L	L	L
9	M	L	H	M	M
10	H	M	L	L	L

(i) **Project Team Skills**: Skill is an import component for the growth of any organization If team is skilled, certainly agility will be enhanced.

(ii) **Project Team Commitment**: Team commitment is always helpful to promote the organization. If team is committed for their duties and performance, customer attraction will be enhanced. This will enhance the trust level with each other and also enhance agility.

(iii) **Inter Project Communication**: Project is completed with various small team members. If communication with teach team is better than productivity and agility will be enhanced.

(iv) **Participation of Project Team**: For the success of any projects depend on the various team who participate in proper way. Proper participation of various teams will ensure agility.

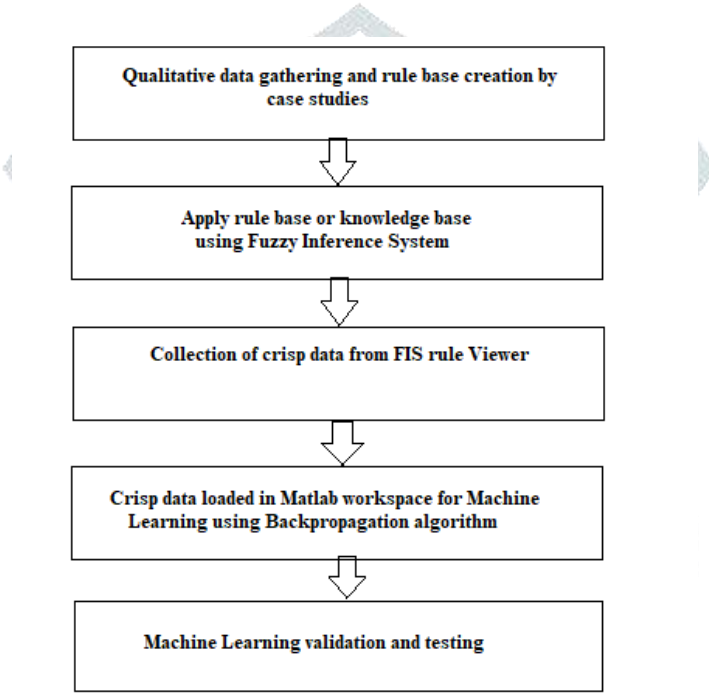

Figure 1. Team Factor Agility enhancement machine learning model

STEP-1 Qualitative data gathering and rule base creation by case studies. We have studied different allied literature and questionnaire based on the case studies of organizations and agile software survey based portals. The ten rules are established which are based on the tacit knowledge of agile software project development professionals. In the given table 1 which shows 4 Team sub factors possess the value in qualitative form Low (L), Medium (M) and High (H), on the basis of these qualitative values we have performed case studies and created the rule base for the same.

STEP- 2 Apply Rule base or knowledge base using fuzzy Inference System: Finally we will try to aggregate the qualitative value of rule base using the centroid method in fuzzy logic to convert the values in quantitative form. For this we have used MATLAB FIS simulator as given in Figure 2.

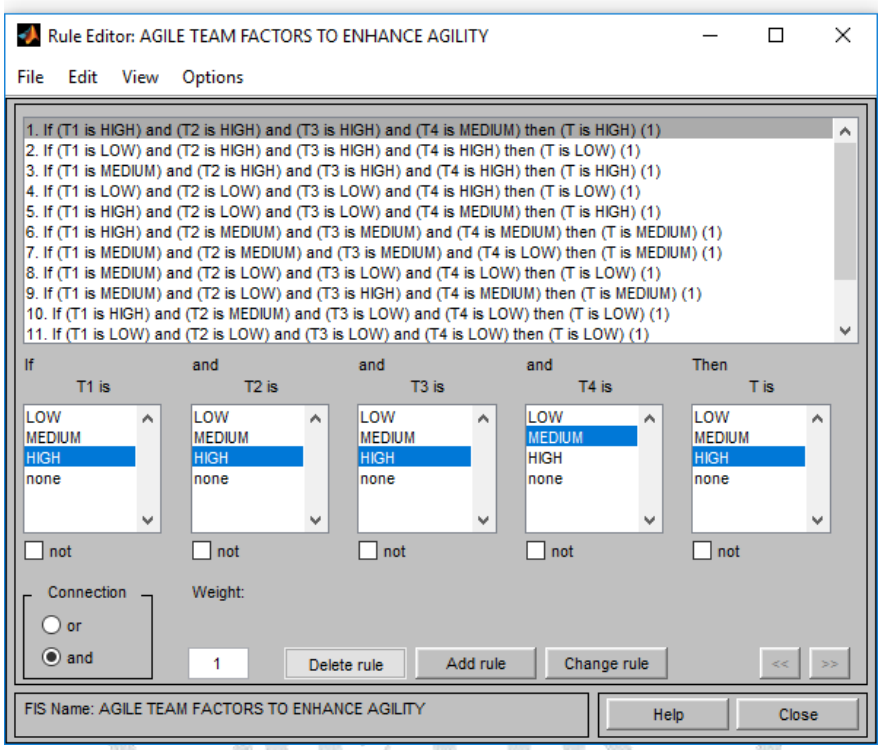

FIGURE 2: RULE EDITOR FOR AGILE TEAM FACTORS TO ENHANCE AGILITY

STEP-3 Collection of crisp data from FIS Rule viewer: We can find the new values by moving slider (Figure 3) which are given with each subfactor in the given and finally we get the result on the basis of rule based applied in the fuzzy inference editor in figure and data shown in Table 3.

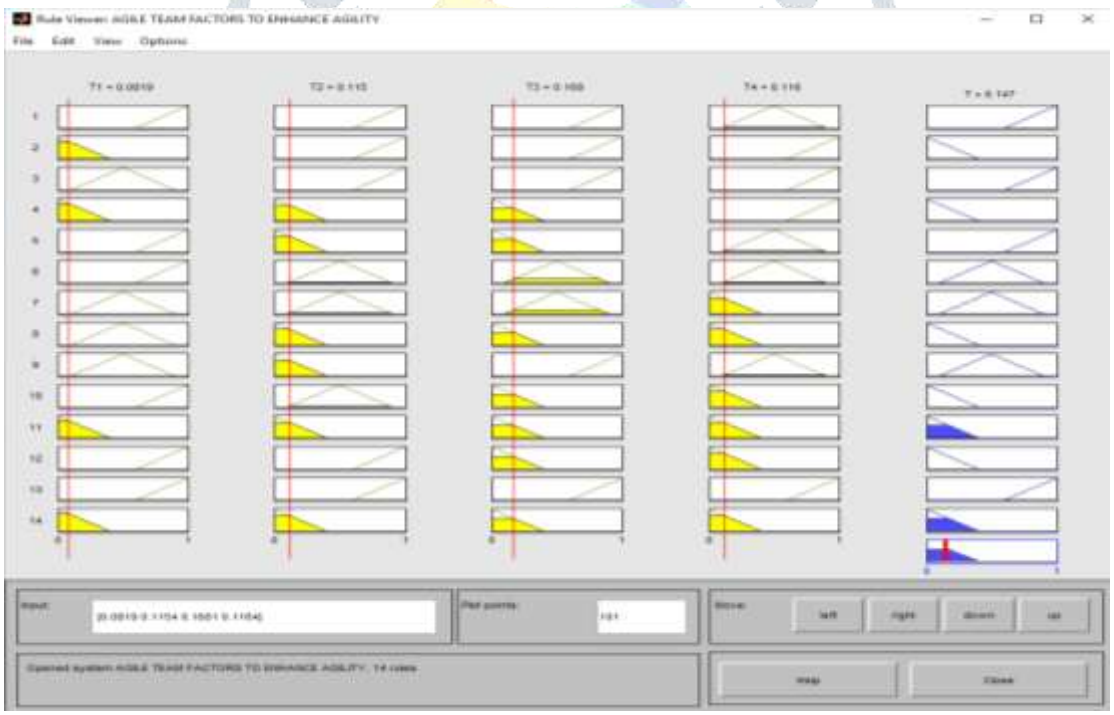

FIGURE 3: RULE VIEWER FOR AGILE TEAM FACTORS TO ENHANCE AGILITY

TABLE 3 : DATA COLLECTION USING FIS RULE VIEWER					
1	0.0380	0.9402	0.9076	0.8207	0.1490
2	0.0380	0.1685	0.1359	0.2880	0.1710
3	0.0380	0.2228	0.0598	0.2554	0.1640
4	0.0380	0.1685	0.1359	0.1685	0.1470
5	0.0380	0.2880	0.1902	0.0707	0.1710
6	0.0380	0.3750	0.1902	0.0707	0.1910
7	0.0733	0.0560	0.2371	0.1681	0.1600
8	0.0747	0.2586	0.0000	0.0000	0.1850
9	0.0798	0.1543	0.1842	0.1211	0.1500
10	0.0815	0.0707	0.2120	0.2663	0.1660
11	0.0815	0.0707	0.0924	0.0815	0.1360

CRISP DATA LOADED IN MATLAB WORKSPACE FOR MACHINE LEARNING USING BACK PROPOGATION ALGORITHM: More than 1000 Data set is collected and uploaded in MATLAB fuzzy inference system for machine learning. This entire working is shown in Research Analysis 4.0.

MACHINE LEARNING TESTING AND VALIDATION: Using Nural Network and fuzzy logic inference system proper testing and validation is done and shown in the resesrch analysis part[17][22].

RESEARCH ANALYSIS: Proposed model of Team Factor also has been implemented with MATLAB based AI Simulator also. To train this model we have collected significant data from MATLAB fuzzy inference system based on qualitative data in knowledge base. In Figure 4 Network architecture shows that 4 neurons are used for input purpose and single neuron is used for output purpose. Initially we started training with 10 hidden neurons and 600 data set were loaded in MATLAB work space. We have chosen randomly 10% data for validation and 10% for testing purpose (Figure 5). We have tained this data but results were not proper then we have changed hidden neurons gradually upto 20 still result was not found up to the mark. Then we collected 350 more data and loaded total data set 950 and started training with 10 hidden neurons and 10% data for validation and 10% data for testing purpose. When I trained this data again and again but best regression result was approx 70% - 90% means final result couldnot be considered.We added more data sets again and again and at the end we got successful training results with 1550 data set which loaded in work space, with 12 hidden neurons and 15% data for validation and 15% data for testing. All four graphs of Regeression found successful with more than 93%.

FIGURE 4: NEURAL NETWORK ARCHITECTURE

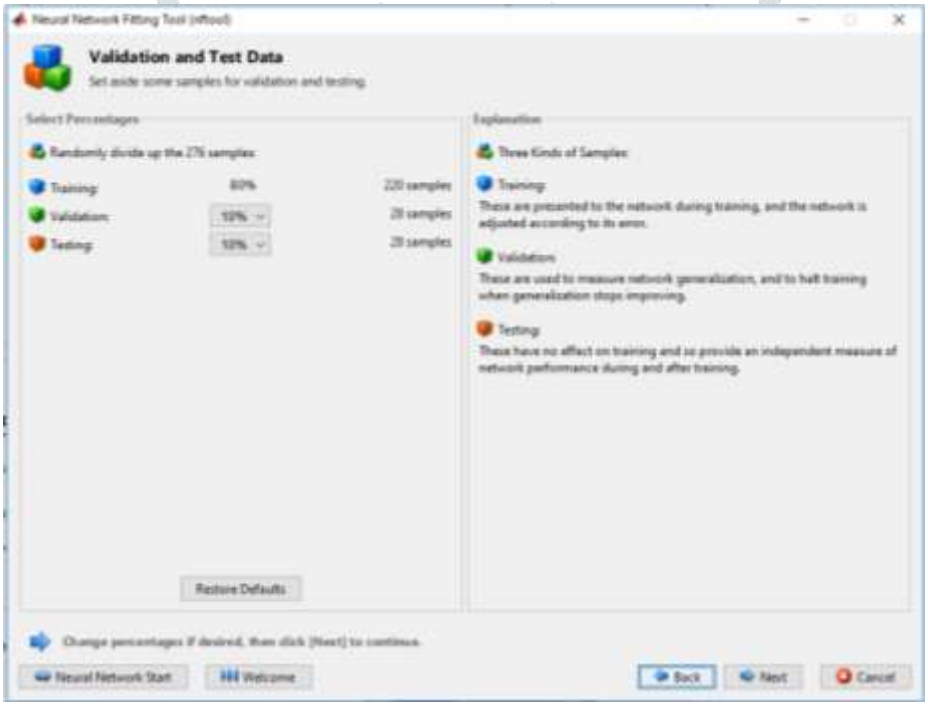

FIGURE 5: NEURAL NETWORK VALIDATION AND TEST DATA

In the figure 6, following four graphs showing the results of Training, Validation, Testing and All results.

(i) **Training of Team Factor:** In this graph bubbles shows the data, blue solid line show about qulified fit data and dotted lines shows the target. This graph show that training result is .98377 i.e. more than 98.3% which is a successful result.

(ii) **Validation of Team Factor:** In this graph bubbles show the data and green solid line shows the fit and dotted line shows the target to be achieved. Bubbles far away from the green line show the unfit data. Result is 0.97787 which is 97.7%, hence result is awesome.

(iii) **Test Graph for Team Factor:** In this graph bubbles shows the data and red solid line shows the fit data and dotted line shows the target data. Result is 0.9598 which is 95.9%.

(iv) **All Result Graph:** In this Result is 0.98046 which is 98.0% as aggregate of Training, Validation and Test results.

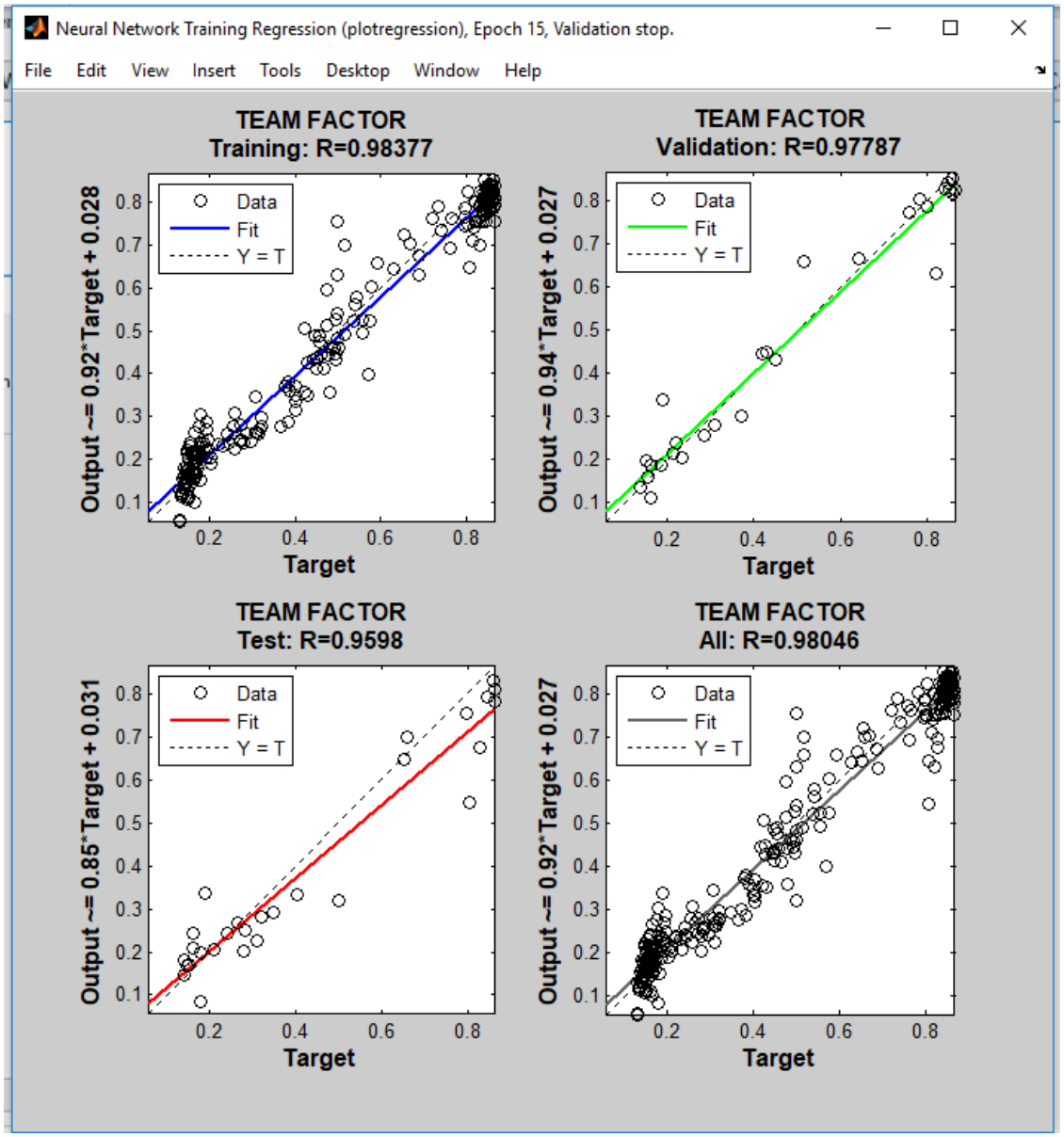

FIGURE 6: REGRESSION GRAPH

In Figure 7, we can see the Neural Network Training Performance for Team Factor with the help of mean square error method. In this graph various lines have different color with different meaning : blue solid line shows training performance, green solid line shows validation performance, red solid line shows Test performance and green dotted line shows targed validation performance. In given graph, it can be observed that mean square error is successively decresing as the training proceed at epoch 9, the validation performance error is near to 0 i.e. 0.0035471. It is obvious that validation is intersecting the best data at epoch 9 and training and test are approaching the same. This way we can see the validation performance is successful and results are acceptable.

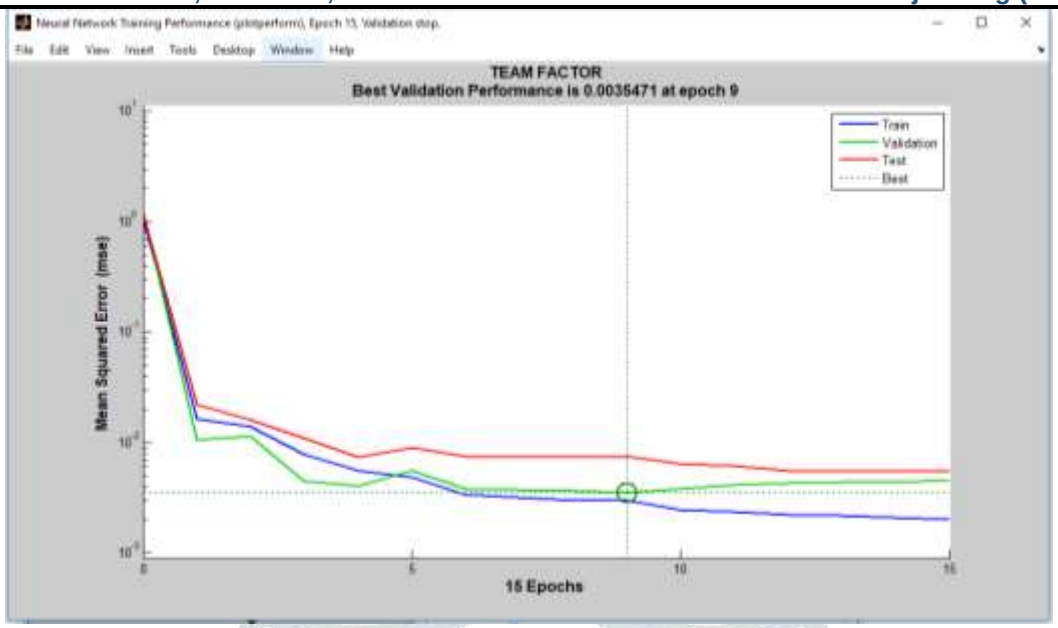

FIGURE 7: NEURAL NETWORK TRAINING PERFORMANCE

In the figure 8, Neural Network Training State graph of Team factor is shown which represents errors in gradients. It is desirable that gradients should be in decreasing order as the training proceeds, we can see that epoch 9 gradient value is 0.0034854 which is very near to zero. We found different performance from epoch 1 to 15 and best validation check at epoch 15.

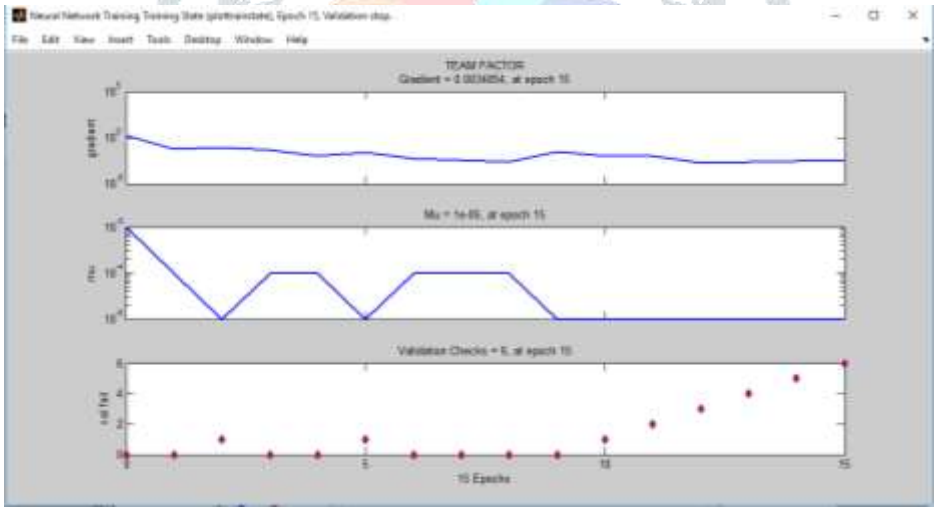

FIGURE 8: NEURAL NETWORK TRAINING STATE GRAPH

In figure 9, Error Histogram shows the error data distribution training, validation and Test for Project Factor. In Graph blue color is training data, green is validation data, red is test data and light yellow line is for zero error. In this graph it can be seen that in 60 instances and 20 data bins. Between 10 and 11 bins error is near to zero (-0.01114 to +0.01473).

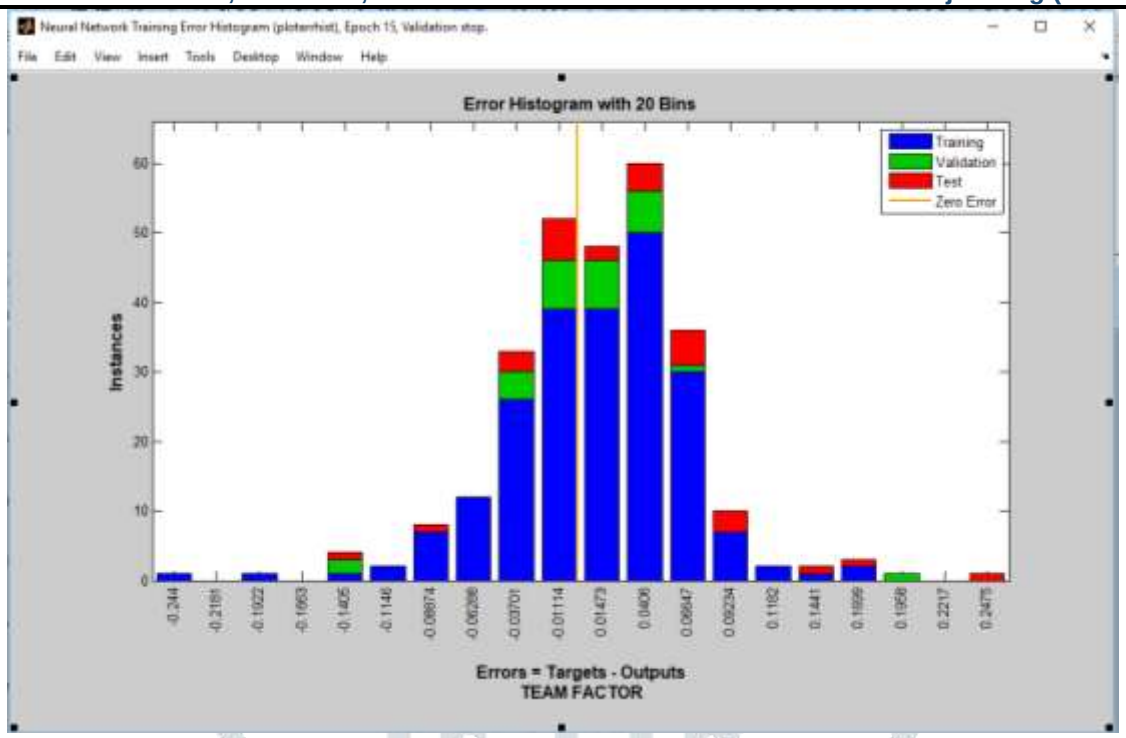

FIGURE 9: NEURAL NETWORK TRAINING ERROR HISTOGRAM

CONCLUSION: In this research paper machine learning approach has been applied to predict the agile team factors weakness and strength in enhancing the agility. The Numerical data was collected from the FIS MATLAB simulator and then used for machine learning training purpose. We have analyzed the good and bad data through the different functions available in the MATLAB AI simulator. We have tested the proposed learning model with different parameters finally we accepted this model with 98% of accuracy. In future this model can be enhanced with few more input parameters and with few more quantitative/ qualitative data collection based on the survey or ethnography techniques of data collection. This model can be made more accurate by the auto machine learning which would be learning technology of the future.

REFERENCES:

[1]. Amabile, T. (1998): "How To Kill Creativity. Harvard Business Review", 76(5), 77–87.

[2]. Bailyn, L. (1985): "Autonomy in the Industrial R&D Laboratory". Human Resource Management 24(1), 129–146.

[3]. Brooks, F. (1987): "No Silver Bullet: Essence and Accidents of Software Engineering". IEEE Computer 20(4), 10–19.

[4]. Beck, K. (1999): "Embracing Change with Extreme Programming", IEEE Computer 32(10), 70–77.

[5]. Cougar, J. (1990): "Ensuring Creative Approaches in Information System Design". Managerial and Decision Economics 11(2), 281–295.

[6]. Canibano, L., Garcia-Ayuso, M., Et Al. (2000): "Shortcomings in the Measurement of Innovation: Implications for Accounting Standard Setting", Journal Of Management And Governance 4(4), 319–342.

[7]. Carayannis, E., Coleman, J. (2005): "Creative System Design Methodologies", Technovation 25(3), 831–840.

[8]. Dougherty, D. (1992): "The Illegitimacy of Successful Product Innovation in Established Firms". Organisation Science 5(2), 200–218.

[9]. Domino, M., Collins, R., Et Al. (2003): "Conflict in Collaborative Software Development". In: Sigmis Conference 2003, Philadelphia, Pa, PP. 44–51.

[10]. Elam, J., Mead, M. (1987): "Designing for Creativity: Considerations For DSS Development", Information & Management 13(2), 215–222.

[11]. Eccles, R. (1991): "The Performance Measurement Manifesto". Harvard Business Review 69(1), 131–137.

[12]. Ekvall, G. (1996): "Organisational Climate for Creativity and Innovation". European Journal of Work and Organisational Psychology 5(1), 105–123.

[13]. Flores, F. (1993): "Innovation By Listening Carefully To Customers", Long Range Planning 26(3), 95–102.

[14]. Fowler, M., Highsmith, J. (2001): "The Agile Manifesto", Software Development 9(8), 28–32.

[15]. Gallivan, M. (2003): "The Influence of Software Developer's Creative Style on Their Attitudes to and Assimilation of a Software Process Innovation", Information & Management 40(1), 443–465.

[16]. Mathisen, G., Einarsen, S. (2004): "A Review Of Instruments Assessing Creative And Innovative Environments Within Organisations", Creativity Research Journal 16(1), 119–140.

[17]. M. Otair And A. S. Walid(2005) , "Speeding Up Back-Propagation Neural Networks," In Proceedings Of The 2005 Informing Science And It Education Joint Conference, Vol. 1, Flagstaff, Az, Usa, June 2005.

[18]. Nonaka, I. (1991): "The Knowledge-Creating Company", Harvard Business Review 69(6), 96–104.

[19]. Newman, M., Robey, D. (1992): "A Social Process Model of User-Analyst Relationships". MIS Quarterly 16(2), 249–266.

[20]. Nerur, S., Mahapatra, R., Et Al. (2005): "Challenges of Migrating to Agile Methodologies", Communication Of The Acm 48(5), 72–78.

[21]. Payne, R. (1990): "The Effectiveness of Research Teams: A Review", In: West, M., Farr, J. (Eds.) Innovation & Creativity at Work, Pp. 101–122. Wiley, Chichester.