DETERMINING THE RISKY SOFTWARE PROJECTS USING ARTIFICIAL NEURAL NETWORKS

Etkin Sakucoglu[1], Laman Valizada[1], Ayse Buharali Olcaysoy[2] and Oya Kalipsiz[1]

[1]Department of Computer Engineering, Yildiz Technical University, Istanbul, Turkey
[2]Strategy and Transformation Department, LC Waikiki, Istanbul, Turkey

ABSTRACT

Determining risky software projects early is a very important factor for project success. In this study it is aimed to choose the most correctly resulting modelling method that will be useful for early prediction of risky software projects to help companies to avoid losing time and money on unsuccessful projects and also facing legal requirements because of not being able to fullfill their responsibilites to their customers While making the research for this subject, it is seen that in previous researches, usually traditional modelling techniques were preferred. But it is observed that these methods were mostly resulted with high misclassification ratio. To overcome this problem, this study proposes a three-layered neural network (NN) architecture with a backpropagation algorithm. NN architecture was trained by using two different data sets which were OMRON data set (collected by OMRON) and 2016-2020 ES.LV data set (collected by the authors) separately. For the made of this study firstly the most relevant classification method (Gaussian Naive Bayes Algorithm) and the most relevant neural network method (Scaled Conjugate Gradient Backpropagation Algorithm) was chosen and both data sets were trained by using each method seperately for the purpose of observing which type of modelling architecture would give better results. Experimental results of this study showed that the neural network approach is useful for predicting whether a project is risky or not risky.

KEYWORDS

Software Project Management, Software Risk Management, Software Risk Analysis, Artificial Neural Network, Data Analysis

1. INTRODUCTION

1.1. Purpose of the Study

With the development of technology, software projects have become much more important in every company. During their work life, it has been observed by the authors that software projects in companies do not always result in success and this situation is caused by many different factors within the company. In unsuccessful software project processes, the reason for this failure is not clearly understood by the employees, and therefore the time spent on projects that are almost certain to fail is getting longer. Based on this observation, the authors searched for a solution and started to examine previous studies in this field. During the research it is realised that a recent industry survey has revealed that software projects can fail due to a variety of problems including cost overload, schedule slippage, requirement misunderstandings, and client dissatisfaction. [1] These software projects aim to satisfy various needs. But unfortunately, not every project ends up with success and not foreseeing failure may cost companies lots of financial problems. Also losing time on an unsuccessful project is not only bad for the time and money but also may force companies to face some legal requirements because of not being able to complete their responsibilities to their customers. This fact highlights the need for early identification of risky projects to enable the planning of essential risk management activities and resources during their implementation. A relevant study which focuses on the problem solution by using Bayesian Classification was examined. [3] It is observed that the success ratio of prediction is 82.5%. In this information and technology era, it is wondered how this success ratio could be improved. [2] In this study creating a system by using neural network model to find a solution to this problem is aimed.

1.2. Content of the Study

This study contains two different data sets (created by asking same questions to groups of people who are from different geographies, working in different companies, different positions and sectors, on different years but it was common that all of them were working in sofware projects in corporate companies). Also, in the made of study it is decided to use the most relevant classification method (Gaussian Naive Bayes Algorithm) [7,8] and the most relevant neural network method (Scaled Conjugate Gradient Backpropagation Algorithm). [9] Both of these methods are applied to both data sets separately.

1.3. Process of the Study

Steps that are followed while doing this study can be seen on Figure 1:

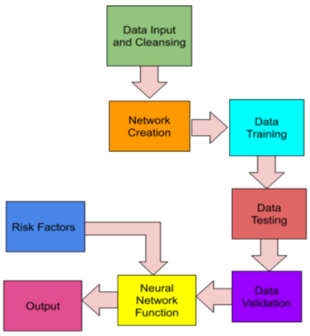

Figure 1. Draft Diagram of How This Study Works

It was decided to use the OMRON data set when starting the project. But those data were collected in 1996-1998. So it is decided to give a new direction to our study by creating our own data set. While doing this study, programmers who work on software projects (preferably computer engineers) in various corporate companies were asked about 22 basic risk factors that will determine the success criteria of software projects. Then our questionnaire answers (data) were collected and a table was made to see them clear. But this data is raw (not processed). Which means these are the results of questionnaire but it requires data cleaning and data validation to get rid of incorrect data and to reach actual data which will be worked on. So data cleaning and data validation was made and final version of our data set (2016-2020 ES.LV data set) was reached. Then answers of these 22 factors was taken as input and results as targets. Then %80 data was used for training and %10 of data was used for test and %10 of data was used for validation. And back-propagation algorithm was used for training our data. This way a network was created. Then the network was tested and a plot of necessary functions were obtained. To achieve the goal, first, most relevant classification algorithm and most relevant neural network algorithm was chosen. Then different attributes of those methods were changed to see their effect on result. (number of neurons on hidden layer, number of data that will be used for training, test, validation; which data will be used for training, test, validation etc.) These trials were made for both classification method and neural network method which were chosen. At the end the settings that gives best results were chosen and applied. The network architecture included 23 attributes: 22 risk factors as input layer variables and the status as the output layer variable. Considering the studies on hidden layers; a single hidden layer is sufficient for constructing any complex problems with the desired accuracy, our network model design has one hidden layer. There are 22 neurons on hidden layer. The nonlinear transfer function used by the hidden and output layers is a sigmoid function. The chosen classification algorithm which was Naive Gaussian Bayes Algorithm and the chosen neural network algorithm which was Scaled Conjugate Gradient Backpropagation Algorithm (Artificial Neural Network Model) was applied to both of the data sets. It is observed that while neural network model's success ratio on both of the data sets were higher then classification method's success ratio, also different outcomes has been realised such as; while Naive Gaussian Bayes Algorithm worked better on Omron's Data set between two data sets, Scaled Conjugate Gradient Backpropagation Algorithm (Neural Network Model) worked better on 2016-2020 ES.LV Data set between two data sets. Even though Gaussian Naive Bayes is a powerful algorithm for predictive modeling it is observed that Scaled Conjugate Gradient Backpropagation Algorithm (Artificial Neural Network Model) is more suitable, effective and giving correct results when applied to these data sets.

2. RELATED WORK

According to ISO/IEC 16085 published by The International Organization for Standardization (ISO) that focusses on the processes to continuously manage risks during the lifecycle of a product, a completed risk management process should consist of seven key activities: "plan and implement risk management"; "manage the project risk profile;" "to perform risk analysis;" "Perform risk monitoring;" "apply risk therapy;" "evaluate the risk management process"; and "technical and management processes". [25] Many predictive models have been offered to aid project managers in managing risky projects in the early phases. For example, Karolak [26] used the Bayesian probability tree approach to develop a software engineering risk management (SERMI) method that forecasts the risk of software projects. The method suggests a structure with 81 risk questions, but no empirical studies or data have been used to inspect or improve its performance yet. According to Tiwana et al. [27] explored the importance of six risk factors for software projects. They then constructed a multiple linear regression model to obtain a general risk score, which was then changed into five given project risk levels (high, moderately high, moderate, moderately low, and low). Even though the basis of the model was based on a large data set containing 720 projects, the estimation performance of the model could not be achieved and the external validity of the extra new projects could not be realized. Mizuno et al. [3] utilized a Bayesian classification approach to estimate the risk trend of a software project based on 40 historical projects at OMRON. The 10-fold cross-validation outcomes showed that seven projects were not forecasted correctly, matching to 17.53% inaccuracy. Also, two of the seven misclassified projects were risky projects. Takagi et al. [28] utilized the same dataset as Mizuno et al. to classify risky projects according to logistic regression. Solely one validation project was misclassified. Unluckily, this misclassified project was a risky one. On the other hand, Amasaki et al. [29] too applied the OMRON dataset to build a predictive model. Eleven rules were mined by the union rule and their minimum confidence was between 0.63 and 0.91. They reached a general accuracy of 75% based on 12 specific projects from 2003 to 2004. But three risky projects were still erroneously classified as non-risk projects. Three known models from the OMRON dataset were developed to construct predictive models for the risky nature of a project. Unfortunately, all of these approaches have the same problem that some risky projects are considered not risky. For this reason, there is great interest in using NN for the same aim. [2]

3. METHODS AND IMPLEMENTATION

3.1. Methods

In this part, methods that are followed while doing this study is explained step by step.

1. Firstly, the methods to be applied in this study were selected from the articles we referenced. Two main articles were examined and one method from each was chosen so that comparison between those articles could be made. Chosen methods are: Gaussian Naive Bayes Algorithm [3] and Scaled Conjugate Gradient Backpropagation [2].
2. Then Omron's data set was obtained. [3]
3. Then it was seen that the data in the Omron data set were collected between 1996-1998 and these data alone would not be sufficient for a current study. It is decided to create a questionnaire (which contains same questions with Omron's data set) and made sure it reached the relevant people and participants were asked to answer those questions based on a software project that they participated in any year between 2016-2020.
4. After that step data cleaning and data validation was made on questionnaire results to reach final data which creates our data set which authors prefer to name it 2016-2020 ES.LV Data Set.
5. Naive Bayes Algorithm (Classification) and Backpropagation Algorithm (Artificial Neural Network Model) were applied to both Omron's data set and 2016-2020 ES.LV data set seperately.
6. Results were examined and comparison between those results were made.

3.2. Implementation

3.2.1. Our First Approach: Classification Method; Gaussian Naive Bayes Algorithm [7, 8]

Firstly, relevant data set is imported. Which means naive gaussian bayes function (trainClassifier( ) function) takes all of data (including results) as parameters and returns trained model and accuracy. Then this function takes input data set and divides it to predictors and responses and makes a table of each one. Predictor table is made to use as input data and response table is made to use as result table. Then to train the classifier, function expands the distribution names per predictor. Then Gaussian is replaced with Normal when passing to the fitcnb( ) function. Because when gaussian bayes function is made, normal distribution is used. [18] Numerical predictors are assigned either Gaussian or Kernel distribution [19] and categorical predictors are assigned mvmn distribution. Gaussian distribution was used in the made of input and result tables. Because a categorical training is aimed; mvmn distribution was used in training. [20] Then the result struct was created with predict function. After that the code for five-fold cross-validation was made. [21] Then computed validation predictions are computed and finally validation accuracy was computed.

3.2.2. Proposed Approach: Neural Network Method; Scaled Conjugate Gradient Backpropagation Algorithm [9]

Firstly, input and target data sets are imported. Scaled conjugate gradient backpropagation was chosen as training function to benefit that this algorithm uses less memory and is suitable in low memory situations. Then a pattern recognition network was created. [22] Size of hidden layer was set. Network was created by using hidden layer size and training function as parameters. Input and output pre/post-processing functions were chosen. Data was divided for training, validation, testing. Data was divided randomly. %80 of data set for training, %10 of data set for testing and %10 of data set for validation. A cross-entropy function was set as performance function. [23] Then plot functions were choosen: plotperform, plottrainstate, ploterrhist, plotconfusion, plotroc. Then the network was trained with train( ) function. The arguments of train function are network, inputs and targets data set. After training, the network was tested. Then training, validation and test performances are recalculated and network was reviewed. myneuralnetwork( ) function was generated, parameters of this function is inputs and outputs. In myneuralnetwork( ) function neural network constants (inputs, layers, dimensions, outputs) has been set. Minimum and maximum input processing functions are mapped for normalization. Finally, sigmoid positive transfer function [16] and sigmoid symmetric transfer function [17] were generated. Figure 2 shows matlab's visualization on how Neural Network function which is used in this study works. [4]

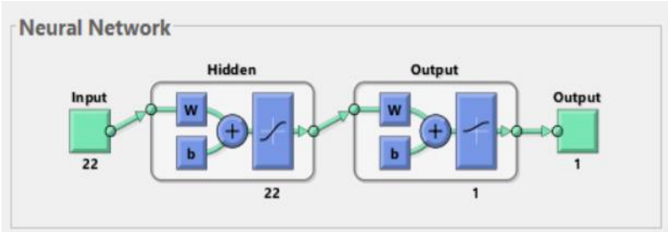

Figure 2. General Design of Neural Network [4]

P.S.: In the made of project various numbers of neurons on hidden layer and various numbers of hidden layer had been tried and comparisons between those results were made to reach the best result. This design above shows the most suitable values to use for this study.

The network architecture included 23 attributes: 22 risk factors as input layer variables and the status as the output layer variable. Considering the studies on hidden layers; a single hidden layer is sufficient for constructing any complex problems with the desired accuracy, our network modeldesign has one hidden layer. [2] There are 22 neurons on hidden layer. The nonlinear transfer function used by the hidden and output layers is a sigmoid function.

4. First Part of the Study: Collection and Usage of 2016-2020 Data Set

In this study two different data sets were used. The first one of these data sets are collected by the authors by an online questionnaire. A questionnaire with multiple choice questions were published on google forms [10] on 2020/11/23 and the questionnaire was closed to accession on 2020/12/15.While doing this study, programmers who work on software projects (preferably computer engineers) are asked about 22 basic risk factors that will determine the success criteria of software projects and after answering those 22 factors they were asked one final question that ask result of the project. They were asked to answer those questions based on a software project that they participated in any year between 2016-2020. The questionnaire questions were obtained by the study conducted by Mizuno et al. in 2004 [3] In this study it is aimed to determine whether there is a risk in the success of projects.

Table 1 shows questions that are asked in online questionnaire.

Table 1. Clarification of Questions Asked on Data Sets

1. Requirements	
1.1	Ambiguous requirements.
1.2	Insufficient explanation of the requirements.
1.3	Misunderstanding of the requirements.
1.4	Lack of commitment regarding requirements between the customer and the project members.
1.5	Frequent requirements or specification changes.
2. Estimations	
2.1	Insufficient awareness of the importance of estimation.
2.2	Insufficient skills or knowledge of estimation methods.
2.3	Insufficient estimation for the implicit requirements.
2.4	Insufficient estimation for the technical issues.
2.5	Lack of stakeholders commitment for estimation.
3. Planning	
3.1	Lack of management review for the project plan.
3.2	Lack of assignment of responsibility.
3.3	Lack of breakdown of the work products.
3.4	Unspecified project review milestones.
3.5	Insufficient planning of project monitoring and controlling.
3.6	Lack of project members' commitment for the project plan.
4. Team Organization	
4.1	Lack of skills and experience.
4.2	Insufficient allocation of the resources.
4.3	Low morale.
5. Project Management Activities	
5.1	Lack of resource management of project managers throughout a project.
5.2	Inadequate project monitoring and controlling.
5.3	Lack of data needed to keep track of a project objectively.

Table 2 shows four choices that are given as an answer choice to the questions on Table 1. [3]

Table 2. Definition of Answer Options

4 types of answer types that could be given to our questions:
S : Strongly Agree
A : Agree
N : Neighter Agree Nor Disagree
D : Disagree

After answering the 22 risk factors in Table 1, each participant answered one last question: How did the project resulted? Table 3 shows the three answer choices for that question and how those answers were categorized.

Table 3. Definition of Results

It resulted in success	Not Risky
The project could not be started due to the conditions	Risky
The project was started but could not be finalized	Risky

What is project success? [11] To qualify the project as successful means achieving the goals in the planned time, budget and efficiency. What is project failure? [12,13] To qualify the project as failure means not achieving the goals in the planned time, budget or efficiency. Project risk is an uncertain event or condition that, if it occurs, has a positive or negative effect on one or more project objectives such as scope, schedule, cost, and quality. [14] What is a risky project? [14, 15] Risky project is a project that may not be completed or achieved the expected success due to some conditions. What is not risky project? [14] Not risky project is a project where current conditions indicate that the project will be successful as a result. In this study projects that ended with success are categorized as "not risky" projects, and projects that ended with failure are categorized as "risky" projects. There was 49 participation to the questionnaire. After doing data cleansing and data validation it is observed that 5 out of these 49 projects were not answered logically. So as a result 44 projects and their evalutions were obtained and a table of our data was made as final data set (Table 4) which is called 2016-2020 ES.LV data set.

Table 4. 2016-2020 ES.LV Data Set

Projects	**1: Requirements**					**2: Estimations**					**3: Planning**						**4: Organization**			**5: Manageme nt**			**Result**
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	6	1	2	3	1	2	3	
PR1	S	S	S	A	A	A	A	A	A	A	A	A	N	S	S	S	S	S	S	S	S	S	Risky
PR2	S	S	S	A	A	S	S	S	S	S	S	S	S	N	A	A	S	S	D	S	S	D	Risky
PR3	D	D	D	D	D	D	D	N	N	N	D	D	N	A	N	D	A	A	D	D	D	D	Not Risky
PR4	S	S	S	S	S	A	N	A	A	S	A	A	A	A	N	N	N	N	N	N	N	N	Risky
PR5	D	N	N	D	N	D	D	D	D	D	D	D	D	D	D	D	N	D	D	D	D	D	Not Risky
PR6	D	N	D	D	A	N	D	D	D	D	D	N	A	D	D	D	D	D	N	D	D	D	Not Risky
PR7	A	A	S	A	A	N	D	A	D	A	N	A	A	A	D	A	N	A	A	D	D	A	Risky
PR8	S	S	S	S	S	N	N	S	A	A	A	A	N	A	S	A	D	A	A	A	A	A	Risky
PR9	A	S	N	N	A	D	D	D	A	S	S	N	D	A	N	N	D	D	D	A	N	A	Not Risky
PR10	D	D	N	A	A	N	D	N	N	N	A	N	D	A	N	N	D	D	D	D	N	D	Not Risky
PR11	D	A	A	N	A	D	D	D	D	A	D	D	N	D	D	D	D	D	D	D	D	D	Not Risky
PR12	N	A	A	A	A	A	S	S	N	A	D	D	N	D	D	N	D	D	N	D	D	D	Not Risky
PR13	N	N	S	S	S	S	S	S	S	N	N	N	N	N	N	A	A	A	A	N	A	N	Not Risky
PR14	A	A	D	S	S	A	D	D	D	D	D	S	D	A	S	D	D	D	D	S	S	D	Risky
PR15	D	S	D	A	A	D	D	N	D	D	A	N	S	D	N	D	D	N	D	D	D	D	Not Risky
PR16	A	A	A	A	N	A	N	A	A	N	S	D	S	A	S	D	D	A	D	S	A	A	Risky
PR17	D	D	D	D	D	D	D	D	D	A	A	D	S	N	N	A	D	A	N	A	A	D	Not Risky
PR18	N	N	N	N	N	N	N	N	D	N	N	D	N	N	N	D	D	D	D	N	N	D	Not Risky
PR19	A	S	A	S	A	A	A	A	N	N	N	N	N	N	A	N	D	N	A	N	A	A	Risky
PR20	A	A	A	A	S	N	D	N	D	A	N	D	A	A	D	D	D	D	D	N	N	D	Not Risky
PR21	N	N	N	D	A	N	D	N	D	D	D	D	N	D	D	D	D	D	D	N	D	D	Not Risky
PR22	A	A	N	N	N	D	D	D	D	A	D	D	D	A	D	D	D	D	N	D	D	A	Not Risky
PR23	S	N	A	A	A	A	S	S	N	A	N	N	A	A	D	N	N	N	N	N	N	N	Not Risky

PR24	N	A	N	D	D	D	D	S	N	N	D	D	N	S	D	N	D	N	D	D	D	D	Not Risky
PR25	D	D	D	S	D	S	S	S	D	N	D	N	S	S	D	N	N	N	S	D	D	D	Risky
PR26	D	D	D	D	D	D	D	D	N	D	D	D	N	N	D	D	D	N	D	N	D	N	Not Risky
PR27	S	S	A	D	A	D	D	D	D	D	D	D	D	A	N	N	A	A	A	D	D	D	Not Risky
PR28	A	A	N	N	A	A	D	A	A	D	A	N	S	S	S	N	D	N	N	D	A	A	Not Risky
PR29	N	N	A	A	A	N	N	N	D	D	D	N	A	A	D	D	D	N	N	D	N	N	Not Risky
PR30	D	N	D	D	N	D	D	A	D	N	D	N	S	N	D	D	D	D	D	D	N	N	Not Risky
PR31	S	A	D	A	D	N	D	N	D	N	D	D	S	A	D	D	D	N	D	D	D	S	Risky
PR32	D	N	N	D	D	D	D	A	D	D	D	D	D	D	D	D	N	N	D	D	D	D	Not Risky
PR33	N	A	A	N	D	D	D	N	D	D	A	D	D	D	D	D	A	D	D	D	N	D	Risky
PR34	A	A	D	D	D	D	D	N	D	D	D	D	D	D	N	D	N	D	D	D	D	D	Not Risky
PR35	N	A	D	D	N	D	D	N	D	D	D	D	D	D	D	D	N	D	D	D	D	D	Not Risky
PR36	S	A	D	D	D	D	D	N	D	S	N	D	A	A	D	D	N	A	N	D	D	D	Not Risky
PR37	A	S	S	N	A	A	A	S	N	D	N	D	S	S	D	D	D	D	D	D	D	D	Not Risky
PR38	A	A	A	A	A	N	D	A	D	D	D	D	A	D	D	D	D	D	D	D	D	D	Not Risky
PR39	S	S	A	S	S	D	D	A	D	N	A	A	A	D	N	D	D	A	A	A	A	D	Risky
PR40	D	D	D	D	D	N	D	N	D	N	D	D	A	N	D	D	N	A	N	N	D	D	Not Risky
PR41	D	D	D	N	D	D	D	D	D	D	D	D	D	D	D	D	N	D	D	D	D	D	Not Risky
PR42	D	D	D	A	D	A	A	A	N	N	N	A	A	D	D	D	A	A	D	D	N	A	Not Risky
PR43	D	D	D	N	D	N	N	D	D	N	D	D	A	A	D	D	D	D	D	D	D	D	Not Risky
PR44	A	D	D	D	D	D	D	D	A	A	D	D	A	N	D	D	D	D	D	D	D	D	Risky

While this neural network method which is explained above in part 3.2.2. is implemented to 2016-2020 ES.LV Data Set, firstly answers of these 22 factors were taken as input and results were taken as targets. Then %80 of our data was used for training, %10 of our data was used for test and %10 of our data was used for validation. And back-propagation algorithm [6] was used to train our data. This way a network was created. Then our network was tested and a plot of necessary functions were obtained. At the end the success ratio of our program can be seen also 22 risk inputs can be manually typed to our neural network model and the results can be checked by comparing with actual results.

4.1. Experimental Results and Analysis of Results

4.1.1. Scaled Conjugate Gradient Backpropagation Algorithm (Artificial Neural Network Model) when it's Applied to 2016-2020 ES.LV Data Set

Table 5 shows analysis of Confusion Matrix of Results of Scaled Conjugate Gradient Backpropagation Algorithm (Artificial Neural Network Model) when it's Applied to 2016-2020 ES.LV Data Set.

Table 5. Confusion Matrix of Results of Scaled Conjugate Gradient Backpropagation Algorithm (Artificial Neural Network Model) when it's Applied to 2016-2020 ES.LV Data Set Analysis Table

Actual Status	Training			Testing			Validation			All		
	Predicted Status			Predicted Status			Predicted Status			Precited Status		
	Risky	Not Risky	Accuracy	Risky	Not Risky	Accuracy	Risky	Not Risky	Accuracy	Risky	Not Risky	Accuracy
Risky	13	0	100%	0	0	NaN%	0	0	NaN%	13	0	100%
Not Risky	0	23	100%	0	4	100%	0	4	100%	0	31	100%
Total Percent	36.10%	63.90%	100%	0%	100%	100%	0%	100%	100%	29.50%	70.50%	100%
Column Numbers:	1	2	3	4	5	6	7	8	9	10	11	12

In the first part of Table 5, analysis of training confusion matrix can be seen:

- Total percent of the column number 1 is %36.10. Which means 13 out of total 36 projects were predicted risky.
- Also, total percent of the column number 2 is %63.90. Which means 23 out of total 36 projects were predicted not risky.
- Accuracy of first row of column 3 is %100. Because 13 out of 13 risky projects are found as risky by prediction model.
- Accuracy of second row of column 3 is %100. Because 23 out of 23 not risky projects are found as not risky by prediction model.
- Total percent of accuracy is calculated as %100.
  So, ratio of true positive values are: %36.10 (13 projects out of 36 projects are true positive) and ratio of true negative values are: %63.90 (23 projects out of 36 projects are true negative) and the total ratio of true positive, true negative, false positive, false negative values are %100. (36.10 + 63.90)/100 = %100.

In the second part of Table 5, analysis of testing confusion matrix can be seen:

- Total percent of the column number 4 is %0. Which means 0 out of total 4 projects were predicted risky.
- Also, total percent of the column number 5 is %100. Which means 4 out of total 4 projects were predicted not risky.
- Accuracy of first row of column 6 is %NAN. Which means not a number. Because 0 out of 0 risky projects are found as risky by prediction model. But in the made of calculation 0/0 is not a valid number.
- Accuracy of second row of column 6 is %100. Because 4 out of 4 not risky projects are found as not risky by prediction model.
- Total percent of accuracy is calculated as %100.

So, ratio of true positive values are: %100 (4 projects out of 4 projects are true positive) and ratio of true negative values are: %0 (0 project out of 4 projects is true negative) and the total ratio of true positive, true negative, false positive, false negative values are %100. (100 + 0)/100 = %100.
In the third part of Table 5, analysis of validation confusion matrix can be seen:

- Total percent of the column number 7 is %0. Which means 0 out of total 4 projects were predicted risky.
- Also, total percent of the column number 8 is %100. Which means 4 out of total 4 projects were predicted not risky.

- Accuracy of first row of column 9 is %NaN. Which means not a number. Because 0 out of 0 risky projects are found as risky by prediction model. But in the made of calculation 0/0 is not a valid number.
- Accuracy of second row of column 9 is %100. Because 4 out of 4 not risky projects are found as not risky by prediction model.
- Total percent of accuracy is calculated as %100.
  So, ratio of true positive values are: %100 (4 projects out of 4 projects are true positive) and ratio of true negative values are: %0 (0 project out of 4 projects is true negative) and the total ratio of true positive, true negative, false positive, false negative values are %100. (100 + 0)/100 = %100.

In the fourth part of Table 5, analysis of all confusion matrix can be seen:

- Total percent of the column number 10 is %29.50. Which means 13 out of total 44 projects were predicted risky.
- Also, total percent of the column number 11 is %70.50. Which means 31 out of total 44 projects were predicted not risky.
- Accuracy of first row of column 12 is %100. Because 13 out of 13 risky projects are found as risky by prediction model.
- Accuracy of second row of column 12 is %100. Because 31 out of 31 not risky projects are found as not risky by prediction model.
- Total percent of accuracy is calculated as %100.

So, ratio of true positive values are: %29.50 (13 projects out of 44 projects are true positive) and ratio of true negative values are: %70.50 (31 projects out of 44 projects are true negative) and the total ratio of true positive, true negative, false positive, false negative values are %100. (29.50 + 70.50)/100 = %100.

4.1.2. Naive Gaussian Bayes Algorithm when it's Applied to 2016-2020 ES.LV Data Set

Figure 3 shows the result of 5-fold cross validation. The rows show the number of projects that are actually risky or not risky. The columns show the number of projects that are predicted as risky or not Risky.

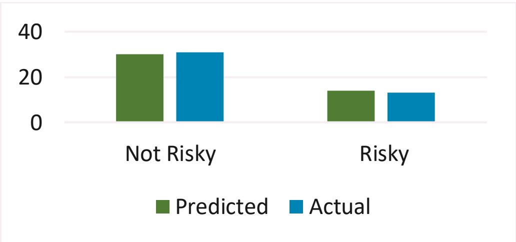

Figure 3. "Results of Five-Fold Cross Validation for Naive Gaussian Bayes Algorithm when it's Applied to 2016-2020 ES.LV Data Set" Graph

As shown in Figure 3, 37 (that is, 27+10) out of 44 projects can be predicted correctly. The predicting accuracy is thus %84.1.

When examining this matrix there is four types of data.

1. False Positive: If actual results are negative but predicted results are positive then the data that provides this condition are named as false positive data.

2. False Negative: If actual results are positive but predicted results are negative then the data that provides this condition are named as false negative data.
3. True Positive: If actual results are positive and predicted results are also positive then the data that provides this condition are named as true positive data.
4. True Negative: If actual results are negative and predicted results are also negative then the data that provides this condition are named as true negative data.

For this matrix;

- Number of false positive data are 3.
- Number of false negative data are 4.
- Number of true positive data are 27.
- Number of true negative data are 10.

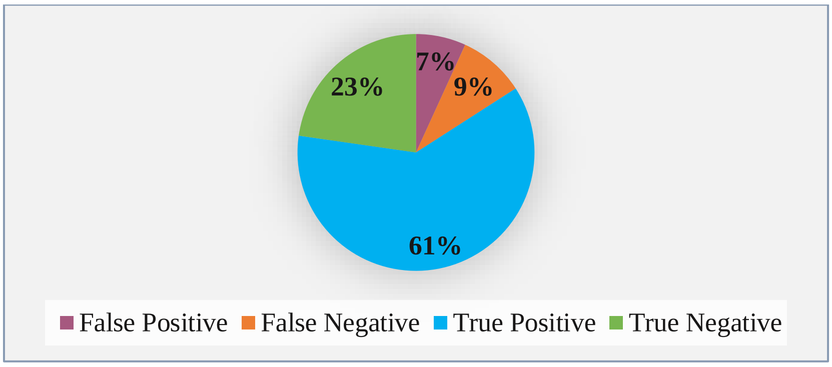

Figure 4. "Success Ratio of Estimation Result Types for Naive Gaussian Bayes Algorithm when it's Applied to 2016-2020 ES.LV Data Set" Chart

5. Second Part of the Study: Usage of 1996-1998 Omron Data Set

Data Collection: In the made of data set on Table 6; 40 projects, which were part of the projects performed from 1996 to 1998 by the SSBC are chosen. (The predictive model was constructed based on 32 projects from 1996 to 1997 and was then validated by 8 projects in 1998. [2] The SEPG (Software Engineering Process Group) distributed the questionnaires to the project managers or the project leaders of 40 target projects and explained the details of the questionnaire and the purpose of the trial. The responses to the questionnaire were collected by the SEPG after one month. Table 6 shows the collected responses. The definition of answers are explained on Table 2. And the result column's values shows the actual results of the projects and definition of how the results were categorized was explained on Table 3. [3]

Table 6. 1996-1998 OMRON Data Set [24]

Projects	**1: Requirements**					**2: Estimations**					**3: Planning**						**4: Organization**			**5: Management**			**Result**
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	6	1	2	3	1	2	3	
PR1	D	D	D	D	D	A	S	S	A	D	A	D	D	D	D	D	A	N	D	D	D	D	Not Risky
PR2	D	D	D	D	D	D	D	D	D	D	D	D	D	A	D	D	D	D	D	D	D	D	Not Risky

PR3	D	D	D	D	S	D	D	A	S	D	D	D	D	D	A	D	D	D	D	D	D	D	Not Risky
PR4	S	S	A	A	S	D	D	A	A	D	A	A	D	D	D	N	A	D	D	D	D	D	Not Risky
PR5	D	D	D	D	A	D	D	D	D	D	D	A	D	A	A	D	D	D	D	D	A	D	Not Risky
PR6	D	S	A	D	D	A	A	A	D	A	D	A	D	D	D	D	D	D	D	D	D	A	Not Risky
PR7	D	D	A	S	A	D	D	D	D	D	D	A	D	S	D	D	D	D	D	D	D	D	Not Risky
PR8	D	A	S	S	D	N	D	A	D	D	A	A	D	D	A	A	D	D	N	S	D	D	Not Risky
PR9	D	A	D	A	S	D	D	D	D	D	A	A	D	A	A	D	D	D	D	D	D	A	Not Risky
PR10	D	D	D	D	A	D	A	A	D	D	D	A	D	D	A	D	D	D	D	A	D	D	Not Risky
PR11	D	S	S	A	D	D	D	S	S	D	D	D	D	D	D	D	D	D	A	D	D	D	Not Risky
PR12	D	A	A	A	D	D	A	D	D	D	D	A	D	A	D	D	D	D	D	D	D	A	Not Risky
PR13	D	A	D	A	D	D	D	D	D	D	A	S	S	D	A	A	A	A	D	A	A	N	Not Risky
PR14	D	D	D	D	S	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	Not Risky
PR15	D	A	A	A	A	D	A	A	D	D	D	D	D	D	D	D	S	A	D	S	D	D	Not Risky
PR16	D	D	D	D	A	D	A	D	A	S	S	A	D	A	S	D	S	A	D	A	A	A	Not Risky
PR17	D	D	D	D	D	D	A	D	D	D	A	A	A	S	A	A	D	D	D	A	A	D	Not Risky
PR18	D	D	D	D	N	D	D	D	D	D	D	A	A	D	D	D	D	D	D	D	D	D	Not Risky
PR19	D	D	D	D	S	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	Not Risky
PR20	D	A	S	A	S	D	D	D	D	D	S	D	D	D	S	D	A	D	D	D	S	S	Not Risky
PR21	D	A	A	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	Not Risky
PR22	S	A	S	S	A	A	N	S	A	N	D	A	A	A	D	N	S	N	A	A	A	D	Not Risky
PR23	A	A	D	A	S	D	D	A	S	D	A	D	A	A	S	A	D	D	D	D	A	S	Risky
PR24	A	A	S	S	S	A	A	S	A	S	S	S	S	A	S	A	S	S	D	A	A	A	Not Risky
PR25	S	A	D	D	S	D	D	D	D	D	S	D	D	S	S	D	D	D	D	D	D	D	Not Risky
PR26	D	A	S	A	A	S	D	A	A	N	D	A	D	D	A	A	D	A	A	A	D	A	Not Risky
PR27	D	A	A	A	A	D	S	A	S	S	D	A	A	D	D	A	A	A	D	D	D	D	Not Risky
PR28	A	S	S	A	A	D	D	S	S	A	S	D	S	D	A	S	A	D	A	D	A	A	Not Risky

PR29	S	A	S	A	D	S	A	A	A	D	D	A	A	A	S	D	A	D	A	D	S	S	Not Risky
PR30	A	A	S	S	A	D	D	A	D	A	A	A	A	A	A	D	S	D	A	D	A	D	Not Risky
PR31	D	D	D	D	D	D	D	D	D	D	S	S	S	S	S	S	S	S	A	S	S	D	Not Risky
PR32	A	S	S	S	A	A	S	S	S	S	S	S	S	A	S	S	S	S	A	S	S	D	Not Risky
PR33	A	D	D	D	D	D	A	D	A	D	D	S	A	D	D	D	S	D	D	A	D	D	Risky
PR34	D	D	D	D	D	D	D	D	D	D	D	A	D	D	D	D	D	D	D	D	D	A	Risky
PR35	A	A	A	A	S	D	A	D	D	D	D	A	A	D	D	D	D	D	D	D	D	D	Risky
PR36	D	D	D	D	A	D	D	D	A	A	A	A	D	D	D	D	D	D	D	D	D	D	Risky
PR37	D	D	D	D	N	D	D	D	D	D	D	A	D	D	D	D	D	D	D	D	D	D	Risky
PR38	S	D	D	A	A	D	A	A	S	S	D	A	A	D	D	D	A	S	D	S	S	D	Not Risky
PR39	A	S	S	S	A	N	N	S	N	N	N	S	S	S	D	A	A	N	S	D	D	D	Not Risky
PR40	A	A	A	A	S	A	D	S	D	D	S	S	S	S	S	S	D	D	D	D	A	A	Not Risky

The authors would also like to remark that doing data cleaning or data validation on Omron's 1996-1998 Data Set (Table 6) was not needed because these steps were already made in that data set. Also, there was not any incompatible, incorrect or missing data.

5.1. Experimental Results and Analysis of Results

5.1.1. Scaled Conjugate Gradient Backpropagation Algorithm (Artificial Neural Network Model) when it's Applied to 1996-1998 Omron Data Set

Table 7 shows analysis of Confusion Matrix of Results of Scaled Conjugate Gradient Backpropagation Algorithm (Artificial Neural Network Model) when it's Applied to 1996-1998 Omron Data Set.

Table 7. Confusion Matrix of Results of Scaled Conjugate Gradient Backpropagation Algorithm (Artificial Neural Network Model) when it's Applied to 1996-1998 Omron Data Set Analysis Table

Actual Status	Training			Testing			Validation			All		
	Predicted Status			Predicted Status			Predicted Status			Precited Status		
	Risk y	Not Risk y	Accur acy	Ris ky	Not Risk y	Accur acy	Ris ky	Not Risk y	Accur acy	Risk y	Not Risk y	Accur acy
Risky	11	0	100%	1	1	50%	0	0	NaN %	12	1	92.30 %
Not Risky	0	21	100%	0	2	100%	0	4	100%	0	27	100%
Total Percent	34.4 0%	65.6 0%	100%	25 %	75%	75%	0%	100 %	100%	30.0 0%	70.0 0%	97.50 %
Column Numbers:	1	2	3	4	5	6	7	8	9	10	11	12

In the first part of Table 7, analysis of training confusion matrix can be seen:

- Total percent of the column number 1 is %34.40. Which means 11 out of total 32 projects were predicted risky.
- Also, total percent of the column number 2 is %65.60. Which means 21 out of total 32 projects were predicted not risky.
- Accuracy of first row of column 3 is %100. Because 11 out of 11 risky projects are found as risky by prediction model.
- Accuracy of second row of column 3 is %100. Because 21 out of 21 not risky projects are found as not risky by prediction model.
- Total percent of accuracy is calculated as %100.
  This calculation was made by CCR formula. Accuracy (AC) is also called the correct classification rate (CCR). It is expressed as the percentage of the total number of predictions that were correct. It is calculated as follows:
  Accuracy= $\frac{TP+TN}{TP+FP+FN+TN}$
  So, ratio of true positive values are: %34.40 (11 projects out of 32 projects are true positive) and ratio of true negative values are: %65.60 (21 projects out of 32 projects are true negative) and the total ratio of true positive, true negative, false positive, false negative values are %100. (34.40 + 65.60)/100 = %100.

In the second part of Table 7, analysis of testing confusion matrix can be seen:

- Total percent of the column number 4 is %25. Which means 1 out of total 4 projects were predicted risky.
- Also, total percent of the column number 5 is %75. Which means 3 out of total 4 projects were predicted not risky.
- Accuracy of first row of column 6 is %50. Because 1 out of 2 risky projects are found as risky by prediction model.
- Accuracy of second row of column 6 is %100. Because 2 out of 2 not risky projects are found as not risky by prediction model.
- Total percent of accuracy is calculated as %75.
  So, ratio of true positive values are: %50 (2 projects out of 4 projects are true positive) and ratio of true negative values are: %25 (1 project out of 4 projects is true negative) and the total ratio of true positive, true negative, false positive, false negative values are %100. (50 + 25)/100 = %75.

In the third part of Table 7, analysis of validation confusion matrix can be seen:

- Total percent of the column number 7 is %0. Which means 0 out of total 4 projects were predicted risky.
- Also, total percent of the column number 8 is %100. Which means 4 out of total 4 projects were predicted not risky.
- Accuracy of first row of column 9 is %NaN. Which means not a number. Because 0 out of 0 risky projects are found as risky by prediction model. But in the made of calculation 0/0 is not a valid number.
- Accuracy of second row of column 9 is %100. Because 4 out of 4 not risky projects are found as not risky by prediction model.
- Total percent of accuracy is calculated as %100.
  So, ratio of true positive values are: %100 (4 projects out of 4 projects are true positive) and ratio of true negative values are: %0 (0 project out of 4 projects is true negative) and

the total ratio of true positive, true negative, false positive, false negative values are %100. (100 + 0)/100 = %100.

In the fourth part of Table 7, analysis of all confusion matrix can be seen:

- Total percent of the column number 10 is %30. Which means 12 out of total 40 projects were predicted risky.
- Also, total percent of the column number 11 is %70. Which means 28 out of total 40 projects were predicted not risky.
- Accuracy of first row of column 12 is %92.30. Because 12 out of 13 risky projects are found as risky by prediction model.
- Accuracy of second row of column 12 is %100. Because 27 out of 27 not risky projects are found as not risky by prediction model.
- Total percent of accuracy is calculated as %97.5.

So, ratio of true positive values are: %67.5 (27 projects out of 40 projects are true positive) and ratio of true negative values are: %30 (12 projects out of 40 projects are true negative) and the total ratio of true positive, true negative, false positive, false negative values are %100. (67.5 + 30)/100 = %97.5.

5.1.2. Naive Gaussian Bayes Algorithm when it's Applied to 1996-1998 Omron Data Set

Figure 5 shows the result of 5-fold cross validation. The rows show the number of projects that are actually risky or not risky. The columns show the number of projects that are predicted as risky or not risky.

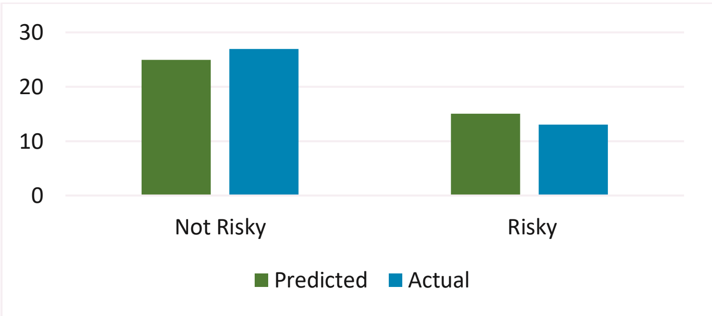

Figure 5. "Results of Five-Fold Cross Validation for Naive Gaussian Bayes Algorithm when it's Applied to 1996-1998 Omron Data Set" Graph

As shown in Figure 5, 36 (that is, 24+12) out of 40 projects can be predicted correctly. The predicting accuracy is thus %90.

For this matrix;

- Number of false positive data are 1.
- Number of false negative data are 3.
- Number of true positive data are 24.
- Number of true negative data are 12.

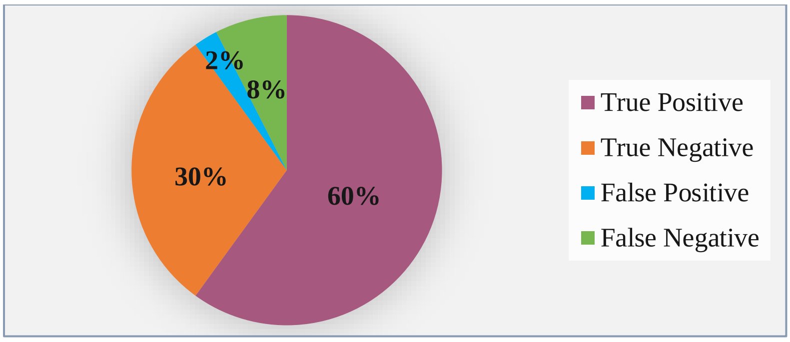

Figure 6. "Success Ratio of Estimation Result Types for Naive Gaussian Bayes Algorithm when it's Applied to 1996-1998 Omron Data Set" Chart

6. THIRD PART OF THE STUDY: COMPARISONS OF RESULTS

6.1. Results of Naive Gaussian Bayes Algorithm "when it's Applied to 2016-2020 ES.LV Data Set" and "when it's Applied to 1996-1998 Omron Data Set"

Table 8 shows the results that are returned by Naive Gaussian Bayes Algorithm when it is applied to data sets seperately. In the Table 8 results which are written in red shows the results that are not found correctly by relevant model and result which are written in grey means that number of projects does not exist in relevant data set. This difference was caused by difference in sizes of two data sets.

Table 8. Results of Naive Gaussian Bayes Algorithm "when it's Applied to 2016-2020 ES.LV Data Set" and "when it's Applied to 1996-1998 Omron Data Set"

Project id	Predicted values of 2016-2020 ES.LV Data Set by Naive Gaussian Bayes	Predicted values of 1996-1998 Omron Data Set by Naive Gaussian Bayes
Project 1	Risky	Not Risky
Project 2	Risky	Not Risky
Project 3	Not Risky	Not Risky
Project 4	Risky	Not Risky
Project 5	Not Risky	Not Risky
Project 6	Not Risky	Not Risky
Project 7	Risky	Not Risky
Project 8	Risky	Not Risky
Project 9	Risky	Not Risky
Project 10	Not Risky	Not Risky
Project 11	Not Risky	Not Risky
Project 12	Not Risky	Not Risky
Project 13	Risky	Risky
Project 14	Risky	Not Risky
Project 15	Not Risky	Not Risky
Project 16	Risky	Risky
Project 17	Not Risky	Not Risky
Project 18	Not Risky	Not Risky
Project 19	Risky	Not Risky
Project 20	Not Risky	Not Risky

Project 21	Not Risky	Not Risky
Project 22	Not Risky	Risky
Project 23	Risky	Risky
Project 24	Not Risky	Risky
Project 25	Risky	Not Risky
Project 26	Not Risky	Risky
Project 27	Not Risky	Risky
Project 28	Risky	Risky
Project 29	Not Risky	Risky
Project 30	Not Risky	Risky
Project 31	Not Risky	Risky
Project 32	Not Risky	Risky
Project 33	Not Risky	Not Risky
Project 34	Not Risky	Not Risky
Project 35	Not Risky	Not Risky
Project 36	Not Risky	Not Risky
Project 37	Not Risky	Not Risky
Project 38	Not Risky	Risky
Project 39	Risky	Risky
Project 40	Not Risky	Risky
Project 41	Not Risky	Does Not Exist
Project 42	Not Risky	Does Not Exist
Project 43	Not Risky	Does Not Exist
Project 44	Not Risky	Does Not Exist

As it can be seen on Table 8, while 7 out of 44 project results were found incorrect by Naive Gaussian Bayes Algorithm when it's applied to 2016-2020 ES.LV Data Set and 4 out of 40 projects were found incorrect by Naive Gaussian Bayes Algorithm when it's applied to 1996-1998 Omron Data Set. So while Naive Gaussian Bayes Algorithm's success ratio on 1996-1998 Omron Data Set is %90, same algorithm's success ratio on 2016-2020 ES.LV Data Set is %84.1.

6.2. Results of Scaled Conjugate Gradient Backpropagation Algorithm "when it's Applied to 2016-2020 ES.LV Data Set" and "when it's Applied to 1996-1998 Omron Data Set"

Table 9 shows the results that are returned by Scaled Conjugate Gradient Backpropagation Algorithm when it is applied to data sets seperately. In the Table 9 results which are written in red shows the results that are not found correctly by relevant model and result which are written in grey means that number of projects does not exist in relevant data set. This difference was caused by difference in sizes of two data sets.

Table 9. Results of Scaled Conjugate Gradient Backpropagation Algorithm "when it's Applied to 2016-2020 ES.LV Data Set" and "when it's Applied to 1996-1998 Omron Data Set"

Project id	**Predicted values of 2016-2020 ES.LV Data Set by Neural Network**	**Predicted values of 1996-1998 Omron Data Set by Neural Network**
Project 1	Risky	Not Risky
Project 2	Risky	Not Risky
Project 3	Not Risky	Not Risky
Project 4	Risky	Not Risky

Project 5	Not Risky	Not Risky
Project 6	Not Risky	Not Risky
Project 7	Risky	Not Risky
Project 8	Risky	Not Risky
Project 9	Not Risky	Not Risky
Project 10	Not Risky	Not Risky
Project 11	Not Risky	Not Risky
Project 12	Not Risky	Not Risky
Project 13	Not Risky	Not Risky
Project 14	Risky	Not Risky
Project 15	Not Risky	Not Risky
Project 16	Risky	Not Risky
Project 17	Not Risky	Not Risky
Project 18	Not Risky	Not Risky
Project 19	Risky	Not Risky
Project 20	Not Risky	Not Risky
Project 21	Not Risky	Not Risky
Project 22	Not Risky	Not Risky
Project 23	Not Risky	Risky
Project 24	Not Risky	Risky
Project 25	Risky	Risky
Project 26	Not Risky	Risky
Project 27	Not Risky	Not Risky
Project 28	Not Risky	Risky
Project 29	Not Risky	Risky
Project 30	Not Risky	Risky
Project 31	Risky	Risky
Project 32	Not Risky	Risky
Project 33	Risky	Not Risky
Project 34	Not Risky	Not Risky
Project 35	Not Risky	Not Risky
Project 36	Not Risky	Not Risky
Project 37	Not Risky	Not Risky
Project 38	Not Risky	Risky
Project 39	Risky	Risky
Project 40	Not Risky	Risky
Project 41	Not Risky	Does Not Exist
Project 42	Not Risky	Does Not Exist
Project 43	Not Risky	Does Not Exist
Project 44	Risky	Does Not Exist

As it can be seen on Table 9 while 0 out of 44 project results were found incorrect by Scaled Conjugate Gradient Backpropagation Algorithm (Artificial Neural Network Model) when it's applied to 2016-2020 ES.LV Data Set and 1 out of 40 projects were found incorrect by Scaled Conjugate Gradient Backpropagation Algorithm (Artificial Neural Network Model) when it's applied to 1996-1998 Omron Data Set. So, while Scaled Conjugate Gradient Backpropagation Algorithm's (Artificial Neural Network Model's) success ratio on 1996-1998 Omron data set is %97.5, same algorithm's success ratio on 2016-2020 Data Set is %100.

7. DISCUSSION

Table 10 was made so that comparison between Classification and Neural Network methods and2016-2020 ES.LV and 1996-1998 Omron data sets can be made.

Table 10. Comparisons of Success Ratio

	Success Ratio of Naive Gaussian Bayes Algorithm (Classification)	Success Ratio of Scaled Conjugate Gradient Backpropagation Algorithm (Neural Network)
1996-1998 Omron Data Set	%90	%97.5
2016-2020 ES.LV Data Set	%84.1	%100

As it is seen on the Table 10, in both data sets Scaled Conjugate Gradient Backpropagation Algorithm (Neural Network Model) has given more accurate results. Also, while Naive Gaussian Bayes Algorithm worked better on 1996-1998 Omron Data Set between two data sets, Scaled Conjugate Gradient Backpropagation Algorithm (Neural Network Model) worked better on 2016-2020 ES.LV Data Set between two data sets.

This study shows that success ratio in statistical studies might be caused by:

- Model that is chosen. (For this study Neural Network vs Naive Gaussian Bayes)
- The years that data in the data set (For this study projects that were answered about) was made. (For this study 1996-1998 vs 2016-2020)
- The location data on that data set was collected. (For this study Turkey vs Japan)
- Difference in size of data set. (For this study 44 projects vs 40 projects)
- How rigorous data cleaning and data validation was made.
- The way data was collected. (For this study face to face vs online questionnaire)

Even though Naive Gaussian Bayes is a powerful algorithm for predictive modeling it is observed that Backpropagation (Artificial Neural Network Model) is more suitable, effective and giving correct results when applied to these data sets. As a result, the usage of Backpropagation Algorithm (Artificial Neural Network Model) is decided and applied.

8. CONCLUSION

In this study the most efficient method to determine risky software projects is investigated. It is observed that Backpropagation (Artificial Neural Network Model) is more suitable, effective and giving correct results than Naive Gaussian Bayes when applied to different data sets. This study gave us an opportunity to see; how method that is chosen can affect the success ratio of the study, how same method works with different success ratio on different data sets, compare how conditions' effects on project’s success changed through years, how different data sets effect success ratio of study. Finally, the goal of this study which was; "DETERMINING THE RISKY SOFTWARE PROJECTS USING ARTIFICIAL NEURAL NETWORKS" was achieved.

REFERENCES

[1] Khaled El Emam and Akif Gunes Koru, “A Replicated Survey of IT Software Project Failures” IEEE Software, Vol. 25, No. 5, 2008, pp. 84-90.
[2] Wen-Ming Han, Discriminating Risky Software Project Using Neural Networks, Computer Standards Interfaces, Volume 40, 2015, Pages 15-22, ISSN 0920-5489 .
[3] Osamu Mizuno, Takanari Hamasaki, Yasunari Takagi, Tohru Kikuno, An Empirical Evaluation of Predicting Runaway Software Projects Using Bayesian Classification, Product Focused Software Process Improvement, 2004, Volume 3009, ISBN: 978-3-540-21421-2
[4] https://www.mathworks.com/products/matlab.html .
[5] Fundamentals of Neural Networks: Architectures, Algorithms and Applications, Laurene V. Fausett, Pearson; 1st Edition, Chapter 1, Page 3 - 37 (December 19, 1993)
[6] The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting (Adaptive and Cognitive Dynamic Systems: Signal Processing, Learning, Communications and Control) by Paul John Werbos, Jan 1, 1994.
[7] McCallum, Andrew; Nigam, Kamal. A Comparison of Event Models for Naive Bayes Text Classification. 1998.
[8] Metsis, Vangelis; Androutsopoulos, Ion; Paliouras, Georgios. Spam Filtering with Naive Bayes—Which Naive Bayes. CEAS 2006 - The Third Conference on Email and Anti-Spam, July 27-28, 2006, Mountain View, California, USA.
[9] Hestenes, Magnus R.; Stiefel, Eduard. Methods of Conjugate Gradients for Solving Linear Systems. Journal of Research of the National Bureau of Standards Vol. 49, No. 6, December 1952 Research Paper 237.
[10] https://www.google.com/intl/tr_tr/forms/about/
[11] Bannerman, P. L. (2008). Defining project success: a multilevel framework. Paper presented at PMI® Research Conference: Defining the Future of Project Management, Warsaw, Poland. Newtown Square, PA: Project Management Institute.
[12] Scanlan, J. (2007). Project failure: the planning paradox. Paper presented at PMI® Global Congress 2007—North America, Atlanta, GA. Newtown Square, PA: Project Management Institute.
[13] Obadia, Bett. (2018). Why Projects Fail. Project Management Journal. Vol 1. 7.
[14] 2013 Project Management Institute. A Guide to the Project Management Body of Knowledge (PMBOK 309 ® Guide) – Fifth Edition
[15] Hillson, D. (2014). Managing overall project risk. Paper presented at PMI® Global Congress 2014—EMEA, Dubai, United Arab Emirates. Newtown Square, PA: Project Management Institute.
[16] Advances in Friction-Stir Welding and Processing Woodhead Publishing Series in Welding and Other Joining Technologies 2014, Pages 543-599
[17] Wlodzislaw, Duch & Jankowski, Norbert. (1999). Survey of Neural Transfer Functions. Neural Computing Surveys. 2. 163-212.
[18] Limpert E, Stahel WA (2011) Problems with Using the Normal Distribution – and Ways to Improve Quality and Efficiency of Data Analysis. PLoS ONE 6(7): e21403.
[19] Weglarczyk, Stanislaw. (2018). Kernel density estimation and its application. ITM Web of Conferences. 23. 00037. 10.1051/itmconf/20182300037.
[20] Naderi, Mehrdad & Arabpour, Alireza & Jamalizadeh, A.. (2017). Multivariate normal mean-variance mixture distribution based on Lindley distribution. Communications in Statistics - Simulation and Computation. 10.1080/03610918.2017.1307400.
[21] Berrar, Daniel. (2018). Cross-Validation. 10.1016/B978-0-12-809633-8.20349-X.
[22] Gail A Carpenter, Neural network models for pattern recognition and associative memory, Neural Networks, Volume 2, Issue 4, 1989, Pages 243-257, ISSN 0893-6080.
[23] Nasr, G.E. & Badr, E. & Joun, C.. (2002). Cross Entropy Error Function in Neural Networks: Forecasting Gasoline Demand.. 381-384.
[24] https://se.is.kit.ac.jp/pman/pman3.cgi?DOWNLOAD=48
[25] Iso/iec 16085:2006, information technology – software lifecycle processes – risk management, iso (international organization for standardization), 2006.
[26] D.w. karolak, software engineering risk management, ieee computer society press, 1996
[27] A. tiwana, m. keil, the one-minute risk assessment tool, commun. acm 47 (11) (2004) 73–77.
[28] Y. takagi, o. mizuno, t. kikuno, an empirical approach to characterizing risky software projects based on logistic regression analysis, empir. softw. eng. 10 (4) (2005) 495–515.

[29] S. amasaki, y. hamano, o. mizuno, t. kikuno, characterization of runaway software projects using association rule mining, lect. notes comput. sci 4034 (2006) 402–407

AUTHORS

Etkin Sakucoglu

Etkin Sakucoglu received her B.S. degree in computer engineering from Yıldız Technical University in 2021. She is currently working as a software developer. She has studies on artificial neural network, web development and machine learning.

Laman Valizada

Laman Valizada completed her B.S. degree in computer engineering from Yıldız Technical University in 2021. She has studies on artificial neural network and web development. She is interested in artificial intelligence and web development.

Ayse Buharali Olcaysoy

Ayse Buharali Olcaysoy completed her PhD in computer engineering in 2016. She has presentations at the national and international conferences on software engineering. She is one of the authors of "System Analysis and Design in Computer Science" of which 4th edition had been published in Turkish in 2019. And also she has been studying as a project & program manager at several projects in the telecommunication and retail sectors since 2004. She has got the PMP certification since 2006.

Oya Kalipsiz

Oya Kalipsiz received her M.S. degree in system analysis from Istanbul Technical University.

She received her PhD degree from Istanbul University with a study on Hospital Information Systems.

She is currently working as a professor in Computer Engineering Department of Yildiz Technical University. Her main research interests are software engineering, database systems, data mining, system analysis, and management information system.