DETECTING FINANCIAL INFORMATION MANIPULATION BY USING SUPERVISED MACHINE LEARNING TECHNICS: SVM, PNN, KNN, DT

Osman Musa AYDIN[1], Ramazan AKTAŞ[2]

Abstract

Within the scope of this paper, traditional estimation algorithms and supervised machine learning methods are used to estimate the manipulation of financial information. Traditional estimation algorithms, such as logit, and supervised machine learning methods, which are support vector machine (SVM), probabilistic neural network (PNN), k-nearest neighbor (KNN) and decision tree (DT) algorithms, are utilized. According to previous studies, support vector machine and probabilistic neural network algorithms perform higher than traditional estimation ones in terms of the accuracy of financial information manipulation estimation. Comparative analysis is made to decide better algorithm for classification by applying all algorithms separately to the financial information manipulation dataset that is collected by skimming weekly bulletins of Capital Markets Board of Turkey and Borsa Istanbul between 2009 and 2018. Thus, it is determined which algorithms perform better in financial information manipulation by looking at performance of classification accuracy, sensitivity and specificity statistics. The obtained results show that KNN and SVM have better performance than the other algorithms and all utilized algorithms have high performance compared to the previous literature's results.

Keywords: *Financial Information Manipulation, Supervised Machine Learning, SVM, KNN, Beneish*
Jel Classification: *G14, G17*

DENETİMLİ MAKİNE ÖĞRENMESİ TEKNİKLERİNİ KULLANARAK FİNANSAL BİLGİ MANİPÜLASYONUNUN TESPİTİ: SVM, PNN, KNN, DT

Öz

Bu çalışma kapsamında, finansal bilgi manipülasyonunu tahmin etmek için geleneksel tahmin algoritmaları ve denetimli makine öğrenmesi yöntemleri kullanılmaktadır. Geleneksel tahmin algoritması olarak logit kullanılırken, denetimli makine öğrenmesi yöntemlerinden destek vektör makinesi (SVM), olasılıksal sinir ağı (PNN), k-en yakın komşu (KNN) ve karar ağacı (DT) algoritmaları kullanılmıştır. Önceki çalışmalara göre, destek vektör makinesi ve olasılıksal sinir ağı algoritmaları geleneksel tahmin algoritmalarından finansal bilgi manipülasyonunu doğru olarak tespit etmekte daha yüksek performans göstermektedir. Sermaye Piyasası Kurulu'nun ve Borsa İstanbul'un 2009-2018 yılları arasındaki haftalık bültenlerini gözden geçirerek toplanan verilere tüm algoritmalar ayrı ayrı uygulanmıştır. Hangi algoritmanın finansal bilgi manipülasyonunu tespitinde daha başarılı olduğuna karar vermek amacıyla karşılaştırmalı analiz yapılmıştır. Karşılaştırmalı analizde, algoritmaların duyarlılık ve özgünlük istatistiklerinin performansına bakılmıştır. Elde edilen sonuçlar, KNN ve SVM'nin diğer algoritmalardan daha iyi performansa sahip olduğunu ve kullanılan tüm algoritmaların önceki literatürün sonuçlarına kıyasla yüksek performansa sahip olduğunu göstermektedir.

Anahtar Kelimeler: *Finansal Bilgi Manipülasyonu, Denetimli Makine Öğrenmesi, SVM, KNN, Beneish*
Jel Sınıflandırması: *G14, G17*

[1] TOBB Ekonomi ve Teknoloji Üniversitesi, İşletme Bölümü, osmanmusaaydin@gmail.com ORCID: 0000-0002-6732-6609
[2] Prof. Dr. TOBB Ekonomi ve Teknoloji Üniversitesi, İşletme Bölümü, raktas@etu.edu.tr

DOI: ulikidince.748742
Makalenin Geliş Tarihi (Recieved Date): 6 /6/ 2020
Yayına Kabul Tarihi (Acceptance Date): 14 /8/ 2020

1. Introduction

As is known, two different types of manipulation are studied in financial literature. The first one is stock manipulation, where investors try to manipulate each other, and the other is financial information manipulation where firm management tries to manipulate other stakeholders. In this study, the preference was on the second type of manipulation. The reason for this is that this issue is a common working subject of both finance and accounting.

One of the basic principles of corporate governance is public disclosure and transparency (Doğanay at al., 2009:1). This important principle states that all interested parties should receive timely and accurate information about the company and thus make rational decisions (OECD, 2004:22). Stakeholders can make rational decision based on information about company. They can reach information about companies with different ways. Publishing financial statements is one of them. Company's financial situation, performance, cash flows and related party transactions are mentioned in those statements. On the other hand, the financial statements should be comparable. In other words, financial statements should be prepared by using the same accounting principles. Moreover, financial statements should not contain any material that misguiding customers to serve companies' purposes. Misleading financial statements lead to deviation from basic principles of corporate governance (Aktaş et al., 2007:1). Moreover, accounting standards are violated with such kind of statements (Arens and Loebbecke, 2000:26). The financial situation and performance of the company are not presented in a fair manner in deviated tables. In order to make the companies' assets more favorable, companies manipulate information in the financial statements so that decision-makers who decide based on this information are misguided and deceived. Therefore, manipulation of financial information leads decision-makers not to make rational decisions. It is extremely important to detect and prevent the manipulation of financial information before companies are made public their manipulated financial statements. The purpose of this study is to determine manipulation of financial information by using some variables obtained from financial statements.

In this research, the sample data are obtained from the bulletins of Borsa Istanbul and Capital Markets Board of Turkey. Manipulation of financial information that is detected during audits or investigations by the Capital Markets Board of Turkey is published in these bulletins. These data will be analyzed separately by logit, k-nearest neighbor, decision tree algorithms, support vector machine and probabilistic neural network methods to predict the manipulation of financial information. The classification performances of the methods are compared according to the specificity, sensitivity and total classification accuracy statistics. Python is used as the software environment in which machine learning algorithms are developed and implemented.

Supervised machine learning methods have emerged as a powerful prediction tool. In such algorithms, data is split two parts as the training dataset and test dataset. After that, training and test dataset are loaded to the system. Machine learning algorithms label each data in learning period by using training dataset and thereby establish a relationship between the input and the output datasets. Algorithms learn the relationship from training data so that these are ready to detect manipulation. The trained algorithms are fed with the test data to detect financial information manipulation and their performance are tested by using confusion matrix. The main objective in this stage is to make effective estimates of the dataset to be investigated.

According to Aktaş et al. (2007), most of the research in this area is based on multivariate statistical methods. Although these multivariate approaches provide a useful tool by developing financial information manipulation models used to make decision about new cases, their accuracy for new cases is not satisfactory in distinguishing the manipulated cases from the nonmanipulated

ones. There are some endeavors to detect financial information manipulation with using support vector machine and probabilistic neural network methods. These are proposed in the following part of this paper.

In this study, not only Support Vector Machine (SVM) and Probabilistic Neural Network (PNN) methods are compared with multivariate statistical methods, but also K-Nearest Neighbor (KNN) and Decision Tree (DT) algorithms are tested for comparison to understand best algorithm. Promising results are obtained especially in SVM and KNN algorithms. In the following parts of the paper, firstly literature review about this topic is presented. After that the data collection and processing part are discussed briefly. Moreover, information about machine learning algorithms that is used in this study are also explained. Finally, the comparison results of all tested algorithms are presented.

2. Literature Review

There are two types of manipulations. One of them is price manipulation which is related to distort stock price. The other one is financial information manipulation. It is related to misstate financial statements deliberately. In this paper, it is focused on financial statement misreporting which violate financial standards. There are some researches about why companies make financial information distortion in their financial statement. On the other hand, some other papers investigate detecting financial information manipulation by using accrual-based methods (Doğanay et al., 2009:2). However, Beneish (1999:1) focused on detecting financial information manipulation by using multivariate statistical method which is probit. Spathis (2002:2) used multivariate statistical method which is logit in his work on seventy-six publicly traded company data in Greece. He implies that indicators such as the ratio of profit to total assets and the ratio of inventories to sales demonstrate good performance in determining financial information manipulation. Küçüksözen (2004), also tried to find financial information manipulation by using probit method with nine indicators.

On the other hand, Fannign and Cogger (1998:1) tried to find financial information manipulation by using neural network with sixty-four indicators. They found that some of them are useful to find manipulation, such as ratio of fixed assets to total assets. Liou (2008:1), tried to detect financial information manipulation by using artificial neural networks. The data of his work were obtained from 3030 companies traded on the Taiwan stock exchange in 2004. Fifty-two indicators were used in this study and it was determined that indicators gave successful results in terms of detection.

Aktaş et al. (2007) also used multivariate statistical method and neural network algorithm to detect manipulation. In Aktaş et al. (2007:2) paper, indicators which are proposed by Beneish (1999) are used to analyze financial information manipulation in Turkey, but the performance of multivariate statistical methods are not good and neural network methods are not working properly for detection.

Apart from these, Doğanay et al. (2009:3) try to detect financial information manipulation by using support vector machine and probabilistic neural network. Doğanay et al. (2009:4) proposed that SVM and PNN are estimating better than multivariate statistical methods. Beneish's indicators are also used in this study.

Beneish (1999:2) proposed eight indicators to identify financial information manipulation. These indicators are calculated from financial statements. In these indicators, the year which happened distortion and the previous year are compared. These indicators are mentioned below:

$$X1 = \text{Sales in receivables indicator} = \frac{\text{Receivables}_t - \text{Sales}_t}{\text{Receivables}_{t-1} - \text{Sales}_{t-1}} \quad (1)$$

$$X2 = \text{Gross margin indicator} = \frac{\frac{\text{Sales}_{t-1} - \text{Cost of Goods Sold}_{t-1}}{\text{Sales}_{t-1}}}{\frac{\text{Sales}_t - \text{Cost of Goods Sold}_t}{\text{Sales}_t}} \quad (2)$$

$$X3 = \text{Asset quality indicator} = \frac{\frac{\text{CurrentAsset}_t - \text{Fix Asset}_t}{\text{TotalAsset}_t}}{\frac{\text{CurrentAsset}_{t-1} - \text{Fix Asset}_{t-1}}{\text{Total Asset}_{t-1}}} \quad (3)$$

$$X4 = \text{Sales indicator} = \frac{\text{Sales}_t}{\text{Sales}_{t-1}} \quad (4)$$

$$X5 = \text{Depreciation indicator} = \frac{\frac{\text{Depreciation}_{t-1}}{\text{Depreciation}_{t-1} + \text{Fix Assets}_{t-1}}}{\frac{\text{Depreciation}_t}{\text{Depreciation}_t + \text{Fix Assets}_t}} \quad (5)$$

$$X6 = \text{Sales, general and administration expenses indicator} = \frac{\frac{\text{Sales, general and administration expenses}_t}{\text{Sales}_t}}{\frac{\text{Sales, general and administration expenses}_{t-1}}{\text{Sales}_{t-1}}} \quad (6)$$

$$X7 = \text{Leverage indicator} = \frac{\frac{\text{Total Dept}_t}{\text{Total Assets}_t}}{\frac{\text{Total Dept}_{t-1}}{\text{Total Assets}_{t-1}}} \quad (7)$$

$$X8 = \text{Total accruals indicator} = \frac{\text{Total Accruals}_t}{\text{Total Assets}_t} \quad (8)$$

The manipulation attempt could be detected by looking at these indicators. Aktaş et al. (2007:2) and Doğanay et al. (2009:2) used these indicators for their study to identify correctly the distorted financial statement by using algorithms. We also use these indicators in this study to catch manipulation.

3. Data and Methodology

The research data in this paper is collected from Borsa Istanbul Daily Bulletins and Capital Markets Board of Turkey Weekly Bulletins. The manipulated financial statements are published in those bulletins. Capital Markets Board of Turkey uncover financial distortion and manipulation during the investigation or the audit. Seventy-nine financial sheets that include manipulation are identified between 2009 and 2018 period. One of the most important part of this research is finding and collecting these manipulated statements because neither Borsa Istanbul nor Capital Markets Board of Turkey propose any database that includes all of the manipulated financial statements. Therefore, we need to skim all of the weekly bulletins from 2009 to 2018 to collect the research data.

According to the financial regulations in Turkey, companies publish their financial statement every quarter and at the end of the year. In this study, it is not specifically focused on single period financial statement like year-end financial statement. Every periods of financial statements are taken into consideration when manipulated financial statements are identified.

On the other hand, eighty-three nonmanipulated financial statements are specified to analyze in this study. Those are chosen from the companies which are in BIST-100 index. Moreover, those companies are mostly trusted by investors since they have not been subject to any investigation regarding financial information manipulation up to the date of this study. Year-end financial statement of those companies in 2010 is used to analyze as nonmanipulated statements. The year 2010 is chosen because along the manipulated financial statement, most of the manipulations were made in this year. The number of nonmanipulated financial statements can be increased by including more companies' financial statements but this does not affect the results significantly. Eighty-three nonmanipulated financial statements are enough to make rational results if the number of manipulated financial statements is taken into consideration.

The balance sheets are taken from Borsa Istanbul. The eight indicators of manipulation that is proposed by Beneish (1999:2), are calculated for manipulated and non-manipulated financial sheets by the help of the python software. These calculated indicators are written in excel sheet. The outliers of this sheet are imputed by the mean substitution technique. All of the proposed algorithms are tested to find manipulation by using this excel sheet.

PNN is one type of the artificial neural network. PNN is generally used for classification problem (Vincent and Kevin, 2002:2). It is feed forward neural network and it has four layers. These are input layer, pattern layer, summation layer and decision layer. First layer is used to feed values of predictor variable to pattern layer. In pattern layer, probability density function is developed (Vincent and Kevin, 2002:5). In the next layer, summing and averaging is made to compute weighted vote for each class. The probability density function of each class is calculated in summation layer. In decision layer, comparison of weighted votes is made. Classification is done by using Bayes' decision rule in this layer.

SVM is one of the supervised learning methods that is implemented by using quadratic programming. SVM is good for classification, regression, and density estimation (Cortes and Vapnik, 1995:1). It classifies groups by a separating hyperplane. Quadratic program minimizes misclassification by finding optimal hyperplane. The hyperplane shapes are decided by kernel functions. Kernel, gama and C are tune parameters for the SVM. Kernel function types are radial basis function, linear kernel function and polynomial kernel function. Gama is the kernel coefficient and C is penalty parameter of quadratic programming.

DT algorithm is used in machine learning for classification and regression. As its name implies it uses tree structure for decision making. In this structure internal nodes represent attributes; leaf nodes represent the result and branches represent decision rules. DT algorithm is very simple. The first step is selecting best attribute by using attribute selection measures such as gini index, information gain and gain ratio to split the data. The second step is making the selected attribute to a decision node and dividing data to smaller subsets. The other steps are building tree with repeating first two steps recursively until there is no attributes or instances (Liu et al., 2010:768).

KNN algorithm is also used in machine learning for classification and regression. It is nonparametric supervised machine learning algorithm. In this algorithm, it is assumed that similar things are close to each other. Since KNN is a type of lazy learning algorithm, the model is computed during classification. In KNN, an instance is assigned to most similar class among the instance similar to it (Moise et al., 2015:4).

4. Results

In this study, Python is used to apply machine learning algorithm and statistical methods. The prepared excel sheet which mentioned above includes eight indicators as input attributes for

manipulated and non-manipulated financial statement. Moreover, in this excel sheet, zero is used for non-manipulated financial statement as an output attribute and one is used for manipulated financial statement as an output attribute. The excel sheet is used in Python to test performance of proposed algorithms. In this excel sheet, there are eighty-three non-manipulated cases and seventy-nine manipulated cases. Randomly selected thirty percent of them are allocated for test data and the remaining seventy percent is used for training algorithms. In this study, the manipulative cases were limited to seventy-nine since only these firms are accessed as a result of the examination. On the other hand, why the number of non-manipulated firms are limited eighty-three could be explained by the fact that increasing the sample size is not creating any significant effect on the results.

In this study, 5-fold cross validation is used to find best smoothing parameters of tried algorithms. K-fold cross validation technique helps us to avoid underfitting problem. In K fold cross validation, data sets are exactly divided into k subsets (Kale et al., 2011:487). The tested algorithms are repeated k times. In each time, one of the subsets are test set and the other k-1 subsets are used to train algorithms. Since every data point becomes test sets' point once and training sets' point k-1 times, this process significantly leads to reduce bias. There is no rule for selection of the K variable but in general K that equals to 5 or 10 is used (Tibshirani, 2019:2).

Performance of the algorithms are depended on smoothing parameters. For DT, attribute selection measure is specified. Gini index is used to select best attributes. Gini index is calculated by following equation:

$$\mathrm{Gini(D)} = 1 - \sum \mathrm{p_{j^2}} \tag{9}$$

Gini calculates how often an element is incorrectly labeled. Thus, it shows impurity level of the sets. If the sets are equally distributed, Gini gets its higher value (Rokach and Maimon, 2005:62). The other smoothing parameter of DT is maximum depth of the tree. We tried max depth of tree from 1 to 8 with increasing 1. We found that 5 gives the best result. The performance of DT is presented in Table 1. Moreover, the confusion matrix of the DT algorithm's result is also presented in Table 2.

In KNN, distance is used as a measure to weight each function in such a way that each distance function weights points by the inverse of their distance. Therefore, closer neighbors have more weights than the far away neighbors (Maillo et al., 2015:167). There are popular distance functions such as Euclidean distance, Hamming distance and Manhattan distance. Hamming distance function evaluates distance between binary points and Manhattan distance function evaluate the distance by summing the absolute difference. Calculation of the distance by Euclidean distance function is demonstrated in the equation 10.

$$\mathrm{d(p,q)} = \sqrt{(\mathrm{p_1} - \mathrm{q_1})^2 + (\mathrm{p_2} - \mathrm{q_2})^2 + \cdots . + (\mathrm{p_n} - \mathrm{q_n})^2} = \sqrt{\sum_{\mathrm{i=1}}^{\mathrm{n}} (\mathrm{p_i} - \mathrm{q_i})^2} \tag{10}$$

Euclidean distance function fits more to properties of the data used in this study. Therefore, for the distance metric of KNN, Euclidean metric is used. On the other hand, in order to get better performance from KNN algorithm, number of neighbors need to be adjusted. Moreover, in order to find best value of number of neighbors, sequence of number from 1 to 80 by 1 interval is tried. We found that the best performance comes with the number 3. We found that the performance of KNN that can be seen in below Tables is very good for this problem.

PNN has standard deviation parameter (std) for Probability Density Function that is approximated by the kernel in pattern layer. Most widely Gaussian kernel is used in PNN (Rutkowski, 2004:2). Therefore, Gaussian kernel is used in this study. Moreover, std parameter needs to be tuned to get better results (Specht, 1988:530). By using 5-fold cross validation different

numbers from 0.1 to 2 by 0.1 intervals are used as std to tune PNN. The best performance is reached when std is equal to 0.4. The result of PNN with best smoothing parameter is equal to DT performance. The results can be seen in below Tables.

SVM also shows good performance for financial information manipulation detection. To reach optimal performance of SVM, smoothing parameters are adjusted accordingly. Since Doğanay et al. (2009) mentioned in their paper that Radial Basis Function (RBF) has better performance than sigmoid kernel and linear kernel, RBF kernel is chosen in this study. RBF kernel function is given in the following equation:

$$K(x, x') = e^{-\gamma\|x-x'\|^2} \quad (11)$$

where $\|x - x'\|$ is Euclidean distance function. SVM makes its classification by following equation.

$$f(y) = \text{sign}(\sum_{i=1}^{n} y_i a_i K(x, x_i) + b) \quad (12)$$

where sign is sign function and b is the bias term. Equation 12 shows that kernel function directly affects the classification result of the SVM therefore gamma parameter has also impact on the result. On the other hand, there is an error parameter, that is C, affecting the parameter b value therefore C also influences the classification result (Hsu et al., 2004:4). Moreover, C and gamma parameters need to be tuned to get better performance. Grid search technique proposed by Hsu et al. (2004:5) is used to determine C and gamma. The combination of C and gamma is searched in this technique. For C parameters, we try numbers from 1 to 80 by 15 intervals. For gamma parameter we try numbers from 0.005 to 0.4 by 0.005 intervals. With using 5-fold cross validation, every combination of gamma and C are tried, and optimal combination is specified. The best performance is reached when C equal to 30 and gamma equal to 0,24. According to our results, the performance of SVM is better than PNN, DT and Logit and it is equal to performance of KNN. These can be seen in below Tables.

On the other hand, in order to increase performance of the Logit, recursive feature elimination (RFE) technique is used. RFE method recursively eliminate features and recalculate accuracy with remaining features (Marquand et al., 2010:13). By using RFE, we found the optimal feature that give best result. After feature elimination, emerging logit model is given in the following equation:

$$y = 0.0992055x_3 + 0.6090048x_4 + 0.3938435x_6 - 0.9592146x_7 - 0.5203768x_8 \quad (13)$$

While finding this Logit model, Beneish (1999:2) model was taken as basis and only the variables (x_3, x_4, x_6, x_7 and x_8) were found statistically significant at the level of 0.05. As it is seen, the features found by utilizing RFE do not include three index that is proposed by Beneish (1999:2). These are trade receivables index, gross profit margin index and depreciation index. In other words, it can be concluded that the effect of these properties on logit regression is less than other indices. In Beneish's (1999:3) study, it was found that the effect of marketing, sales, distribution and general administrative expenses index, depreciation index and leverage index have less impact on detection of financial information manipulation. In this context, it was observed that the depreciation index had little effect both in this study and in Beneish's (1999:3) study. Moreover, we also reached that the performance of the Logit is worse than all the other algorithm like Doğanay et al. (2009:4) mentioned in their paper.

Performance of algorithms are compared based on statistics mentioned by Doğanay et al. (2009). These are specificity, sensitivity and total classification accuracy.

$$\text{Specificity} = \frac{\#\text{ Correctly Classified Nonmanipulated instances}}{\#\text{ Total Nonmanipulated instances}} \quad (14)$$

$$\text{Sensitivity} = \frac{\#\text{ Correctly Classified Manipulated instances}}{\#\text{ Total Manipulated instances}} \quad (15)$$

$$\text{Total Clasification Accuracy} = \frac{\#\text{ Total Correctly Classified instances}}{\#\text{ Total instances}} \quad (16)$$

According to these statistics performance of the algorithm is specified, and following Tables are prepared.

The results of the mentioned algorithms are presented in Table 1. The statistical performance values of the algorithms calculated based on these results are giving in the Table 2.

Table 1: **Confusion Matrices of Algorithms**

Algorithms	True Data	Predicted Data	
		Manipulated	Non-Manipulated
Logit	Manipulated	19	5
	Non-Manipulated	10	15
PNN	Manipulated	23	1
	Non-Manipulated	9	16
DT	Manipulated	17	7
	Non-Manipulated	3	22
KNN	Manipulated	23	1
	Non-Manipulated	5	20
SVM	Manipulated	21	3
	Non-Manipulated	3	22

Table 2: **Test Performance Statistics of Algorithms**

Algorithms	Specificity (%)	Sensitivity (%)	Total Accuracy (%)
Logit	60	79	69
PNN	64	96	80
DT	88	71	80
KNN	80	96	88
SVM	88	88	88

These findings support the view put forward at the beginning of the study. Although the logit is giving a useful model which is not possible with machine learning algorithms, its performance is not better than the methods based on machine learning in detecting cases that were either manipulated or not manipulated. On the other hand, SVM and KNN algorithms came out as expected among the models based on machine learning.

5. Conclusion

In this research, we try to detect financial information manipulation by using supervised machine learning algorithms and multivariate statistical method. For this purpose, we use Logit,

KNN, PNN, SVM and DT. Comparative analysis is made to decide better algorithm for detecting manipulation by using contemporary data obtained from Borsa Istanbul and the Capital Markets Board of Turkey. KNN and SVM shows better performance among the other algorithms. Their performances are satisfactory.

This study can help market regulatory institutions as an early warning model in determining the manipulation of financial information cases as well as guide investors in the accuracy of financial information when making decisions regarding stock investments. In addition, audit firms can use this model to detect the accuracy of financial statements of their customer firms.

In machine learning methods, a large amount of data will improve performance. In this study, the data of the companies are examined in t and t-1 years. Therefore, increasing this range for future studies will increase the accuracy of the algorithms. Moreover, including more companies' data in research will improve the learning performance of algorithms and increase their success in future studies.

As a result, it was confirmed that machine learning methods performed better than statistical methods in the detection of manipulation in accordance with the results of previous studies performed in the field (Doğanay et al., 2009:5). However, while statistical methods form a formula for the detection of financial manipulation, this is not the case with machine learning methods. While the formula obtained from logit can be used to test new data, the algorithms in machine learning methods need to be recalculated with these new data. Therefore, it can be said that statistical methods are more useful when the system is evaluated with new data.

References

Aktaş, R., Alp, A. and Doğanay, M. M. (2007). Towards Predicting Financial Information Manipulation. *The ICFAI Journal of Applied Finance*, 13(7), 39–52.

Arens, A. A. and Loebbecke, J. K. (2000). *Auditing: An Integrated Approach (Internatial ed.)*. New Jersey: Prentice Hall International, Inc.

Beneish, Messod D. (1999). The Detection of Earnings Manipulation. *Financial Analysts Journal*, 55(5), 24–36.

Cortes, Corinna., Vapnik, Vladimir N. (1995). Support-Vector Network. *Machine Learning*, 20, 273-297.

Doğanay, M., Aktaş, R., Alp, A. and Öğüt H. (2009*).* Prediction of Financial Information Manipulation by Using Support Vector Machine and Probabilistic Neural Network. *Expert Systems with Applications*, 36, 5419–5423.

Fanning, K., M. and Cogger, K., O. (1998). Neural Network Detection of Management Fraud Using Published Financial Data, International Journal of Intelligent Systems in Accounting, *Finance & Management*, 7(1), 21-41.

Hsu, C.-W., Chang, C.-C. and Lin, C.-J. (2004). A Practical Guide to Support Vector Classification, Technical Report. Department of Computer Science and Information Engineering, National Taiwan University.

Kale, S., Kumar, R. and Vassilvitskii, S. (2011). Cross-Validation and Mean-Square Stability. *The Second Symposium on Innovations in Computer Science*, 487-495, Sunnyvale, CA.

Küçüksözen, C. (2004). Financial Information Manipulation: Causes, Methods, Objectives, Techniques, Results and An Empirical Study on IMKB Companies. (Unpublished Doctoral Dissertation). Ankara University Social Science Institute, Management, Ankara.

Liou, F. M. (2008). Fraudulent Financial Reporting Detection and Business Failure Prediction Models: A Comparison. *Managerial Auditing Journal*, 23(7), 650-662.

Liu, W., Chawla S., Cieslak D. and Chawla N. (2010). A Robust Decision Tree Algorithm for Imbalanced Data Sets. *SIAM International Conference on Data Mining*, 766-777, Ohio.

Maillo, J., Triguero, I. and Herrera, F. (2015). A MapReduce-based K-Nearest Neighbor Approach for Big Data Classification. *IEEE Trustcom/BigDataSE/ISPA Conference,* 167-172, Helsinki.

Marquand, A., Simoni, S., O'Daly, O., Metha, M. and Mourao-Miranda, J. (2010). Quantifying the Information Content of Brain Voxels Using Target Information, Gaussian Processes and Recursive Feature Elimination. *First Workshop on Brain Decoding: Pattern Recognition Challenges in Neuroimaging*, 13-16, Istanbul.

Moise, I., Pournaras, E. and Helbing, D. (2015). *K-Nearest Neighbour Classifier*.

OECD (2004). OECD Principles of Corporate Governance.

Rokach, L. and Maimon, O. (2005). Top-Down Induction of Decision Trees Classifiers—A Survey. *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev*, 35(4), 476–487.

Rutkowski, L. (2004). Adaptive Probabilistic Neural Networks for Pattern Classification in Time-Varying Environment. *IEEE Transactions on Neural Networks*, 15, 811–827.

Specht, D. (1988). Probabilistic neural networks for classification, mapping, or associative memory, *IEEE 1988 International Conference on Neural Networks*, 525-532, San Diego.

Spathis, C. T. (2002). Detecting False Financial Statements Using Published Data: Some Evidence from Greece. *Managerial Auditing Journal*, 17(4), 179-191.

Tibshirani, R. (2019). K-Fold Cross-Validation. Cross-Validation and Bootstrap. Retrieved from statweb.stanford.edu/~tibs/sta306bfiles/cvwrong.pdf.

Vincent, C., Kevin, C. (2002). *An introduction to Probabilistic Neural Networks*.