VOL 3 (2019) NO 4

e-ISSN : 2549-9904
ISSN : 2549-9610

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

# The Use of Data Mining Techniques in Predicting the Noise Emitted by the Trailing Edge of Aerodynamic Objects

Abdusalam Abdulla Shaltooki[#], Mojtaba Jamshidi[*]

[#] *Department of Information Technology, University of Human Development, Sulaymaniyah, Iraq*
[*] *Department of Electrical, Computer and IT Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran*
*E-mail: salam.abdulla@uhd.edu.iq, jamshidi.mojtaba@gmail.com*

***Abstract*— Aerodynamic is a branch of fluid dynamics that evaluates the behavior of airflow and its interaction with moving objects. The most important application of aerodynamic is in aerospace engineering, designing and construction of flying objects. Reduction of noise emitted by aerodynamic objects is one of the most important challenges in this area and many efforts have been to reduce its negative effects. The prediction of noise emitted from these aerodynamic objects is a low-cost and fast approach that can partially replace the "fabrication and testing" phase. One of the most common and successful tools in prediction procedures is data mining technology. In this paper, the performance of different data mining algorithms such as Random Forest, J48, RBF Network, SVM, MLP, Logistic, and Bagging is evaluated in predicting the amount of noise emitted from aerodynamic objects. The experiments are conducted on a dataset collected by NASA, which is called "Airfoil Self-Noise". The obtained results illustrate that the proposed hybrid model derived from the combination of Random Forest and Bagging algorithms has better performance compared to other methods with an accuracy of 77.6% and mean absolute error of 0.2279.**

***Keywords*— Data mining, Classification, Combined model, Noise prediction, Aerodynamic objects.**

## I. INTRODUCTION

Aerodynamics is a branch of the gas dynamics, and in general, the fluid dynamics that studies the behavior of the airflow and its interaction with moving objects. The solution of an aerodynamic problem consists of the calculation of the velocity field, pressure, and air temperature around an object. For this purpose, the governing equations on fluid flow should be solved. Using the obtained solution, it is possible to calculate the force and momentum that are applied to the body [1].

The most important application of aerodynamic is in aerospace engineering, designing and construction of flying objects. Additionally, aerodynamics is used in automotive engineering to design an efficient body for the automobiles. The structural engineers also use aerodynamics to analyze the influence of wind flow on structures such as skyscrapers, bridges or towers. Hence, a structure or body that is designed and built on the basis of aerodynamic principles and rules should withstand the least possible force from the air or fluid around it. Furthermore, with the use of aerodynamics, the lifting force of the flying object against the gravity of the earth can be augmented [2, 3].

Many challenges and issues have been raised in this area that designers and constructors are trying to solve, among them, reduction of noise generated by aerodynamic objects to moderate the resulting negative effects can be considered as the most important issues.

Therefore, the determination of the noise levels in designed objects is mandatory. Predicting the noise amount emitted by these aerodynamic objects is a low-cost and fast way that can partly replace the "fabrication and testing" phase. Among different methods for prediction, data mining is one of the most common and successful tools [3, 4].

Airfoil self-noise is due to the interaction between an airfoil blade and the turbulence produced in its own boundary layer and near wake. It is the total noises produced when an airfoil encounters smooth nonturbulent inflow [2]. In recent years, engineers and scientists have been able to reduce aeroacoustic and vibroacoustic noise to such an extent that broadband sources are now limiting further noise reduction. This is particularly true for technology that utilizes airfoils and airfoil-like shapes that generate broadband noise at the trailing edge (TE).

In aerodynamic objects, prediction of TE noise has become a permanent challenge for engineers over the past 30 years due to the complexity of the turbulent fluid flow, which is considered as the source of the noise. The complex and random nature of turbulence has led to the development of methods that have used simplified turbulence models to calculate noise.

In this paper, the performance and efficiency of different data mining algorithms such as Random Forest, J48, RBF Network, SVM, MLP Neural Network, Logistic, and Bagging are evaluated in predicting the amount of noise emitted from aerodynamic objects [5].

The rest of this paper is organized as follows. Section II presents related work, Airfoil Self-Noise dataset, and the proposed model. Section III presents the simulation results. Finally, the paper is concluded in Section IV.

## II. Material And Method

In this section, some existing works are studied first. Then, the Airfoil Self-Noise dataset used in this study is introduced. Finally, the proposed model is presented.

### A. Related Work

In reference [3], a report was presented by NASA, which provides a comprehensive review of noise in aerodynamic objects as well as its prediction. The prediction methods for individual self-noise mechanisms are in fact semi-empirical and are based on previous theoretical studies and most comprehensive and available self-noise dataset. In this study, a series of acoustic and aerodynamic experiments were taken from a two-dimensional and three-dimensional aerodynamic blade guided in a wind tunnel without reflection. Data were collected from the blades of seven NACA 0012 aerodynamic devices.

The key to an accurate TE noise prediction is to estimate the turbulence properties correctly. Noise predictions were made by Brooks and Hodgson [6] using data obtained from simultaneous noise and surface pressure measurements. For cases where the exact surface pressure spectrum (i.e. the turbulent field) is not known, estimates are used [2] and predictions are poor at high frequencies.

Lutz et al. [7] employed a surface pressure formulation with a boundary layer numerical flow simulation to improve the estimate of the fluctuating surface pressure spectrum. Roger et al. [8] have proposed an extension of Amiet's original formulation of trailing-edge noise based on fully analytical derivations. Back-Scattering, leading edge correction is developed, yielding a modified chordwise distribution of the acoustic sources induced by the scattering mechanism.

A neural network prediction approach has been proposed to compute self-noise of airfoils typically used in wind turbines by Antonio [9]. The neural networks were trained using experimental data corresponding to tests of several different airfoils over a range of flow conditions.

The TE noise model for the turbulent boundary layer is presented in [10], which is more complicated compared to the semi-empirical method [2]. In this model, boundary layer parameters are used to estimate TE noise on both sides of an aerodynamic object; these parameters are calculated by a boundary layer prediction routine called XFOIL.

Lloyd et al. [11] have developed an immersed boundary approach for use with direct numerical simulations (DNS) employing high-order accuracy spatial schemes.

In [12], an investigation of the noise emitted from the trailing edge (TE) of a Somers S834 airfoil section with advanced experimental and numerical methods is presented. In [13], the double spatial derivative of the pressure at the source points was calculated and the reciprocal theorem was used to determine the tailored Green's function of the body. This approach relies on taking the double spatial derivative of the volume distribution of acoustic pressures to determine the tailored Green's function. Significant errors can be introduced by spatial discretization and differentiation on a numerical grid

In [14], an analytic trailing edge noise model is used to determine the unsteady pressure on the blade surface. In [15], a method has been developed to predict the self-noise generated by a flat plate immersed in low Mach number flow. A Reynolds Averaged Navier-Stokes (RANS) simulation is performed of the turbulent flow over the flat plate. The predicted flow field data, such as mean velocity, turbulent kinetic energy, and turbulent dissipation rate, is then processed using a statistical noise model and combined with a Boundary Element Method (BEM) model of the flat plate to predict the far-field sound.

### B. Airfoil Self-Noise Dataset and Pre-Processing

The dataset used in this research is “Airfoil Self-Noise” [2, 9], which is collected by NASA. This dataset contains 1502 samples and 6 attributes with no missing value. The Output field or classification variable in this dataset is “Scaled sound pressure level”. The attributes of this dataset are:

1. Frequency, in Hertz.
2. Angle of attack, in degrees.
3. Chord length, in meters.
4. Free-stream velocity, in meters per second.
5. Suction side displacement thickness, in meters.
6. Scaled sound pressure level, in decibels. (class attribute)

The essential statistical information about attributes is provided in the following; if necessary, the required pre-processing would be applied on them:

Frequency: This attribute is numerical and contains 21 distinct values in the entire dataset with no missing value. The vacation range of this attribute is between 200 and 20000, with the histogram presented in Fig. 1. As shown in Fig. 1, most of the samples in this attribute are drawn to the value of 200, so it's better to transform it by a simple algorithm ln(x+1). Through the use of this function for the intended attribute, as shown in Fig. 2, the vacation range would be transformed into 5.3 and 9.9. Afterward, by applying the desired data mining method and fitting models, the outputs are post-processed by the inverse of the ln(x+1) function.

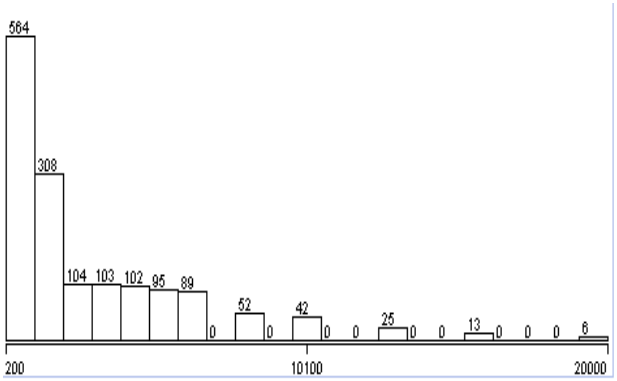


Fig. 1. A histogram of attribute "frequency" before pre-processing

**Angle of attack**: This attribute is of numerical type and contains 27 distinct values in the entire dataset with no missing value. The variation range of this attribute is between 0 and 22.2, with the histogram provided in Fig. 3. Due to the smooth trend of this graph, there is no need for pre-processing operations.

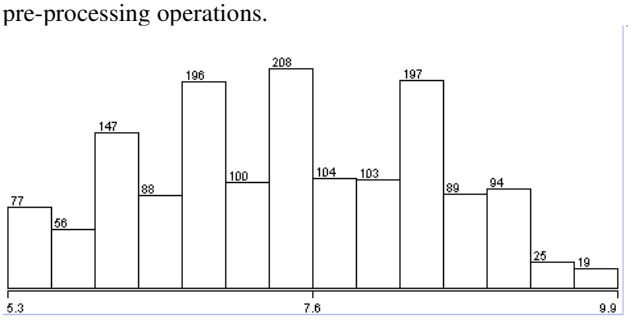


Fig. 2. The histogram of the attribute "frequency" after applying the function ln(x+1)

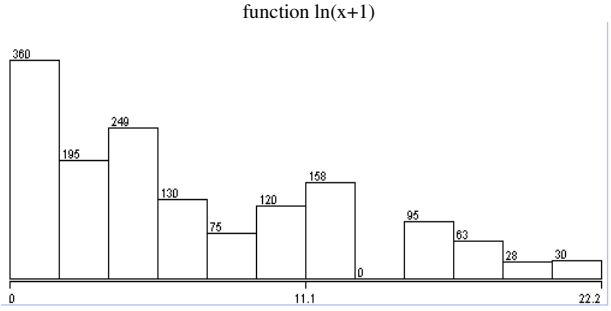


Fig. 3. A histogram of attribute "angle of attack" before pre-processing

**Chord length**: This attribute is also numerical with six distinct values and no missing value. The range of this attribute is between 0.025 and 0.305 with the histogram depicted in Fig. 4. Due to the smooth trend of this graph, there is no need for pre-processing operations.

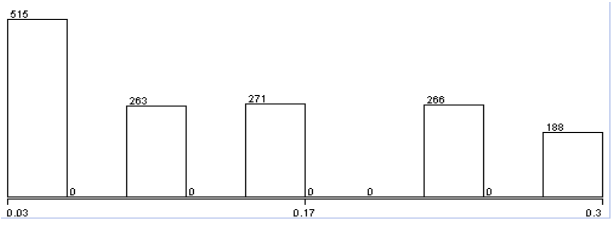


Fig. 4. A histogram of attribute "chord length" before pre-processing

**Free-stream velocity**: This attribute is also of numerical type with four distinct values and no missing value. For this attribute, the variation range is between 31.7 and 71.3 with the histogram presented in Fig. 5. As mentioned for the prior attribute, pre-processing is not needed due to the smooth trend of this graph.

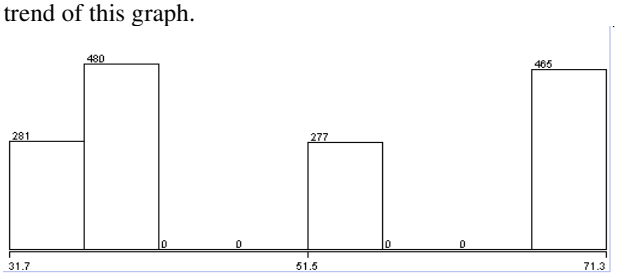


Fig. 5. A histogram of attribute "free-stream velocity" before pre-processing

**Suction side displacement thickness**: This attribute is also numerical and 105 distinct quantities in the entire dataset with no missing value. The variation range is between 0 and 0.058 with the histogram provided in Fig. 6. According to the large variation domain of this attribute, pre-processed is required. The values of this attribute are placed in 105 clusters, which reduce the accuracy of classification algorithms. On the other hand, the values of this attribute are very small decimal amounts (less than 0.058) with the precision of $10^{-7}$. If we reduce the accuracy of the values to $10^{-5}$, the number of clusters or individual samples will be reduced to 78, which will increase the accuracy of the classification algorithms.

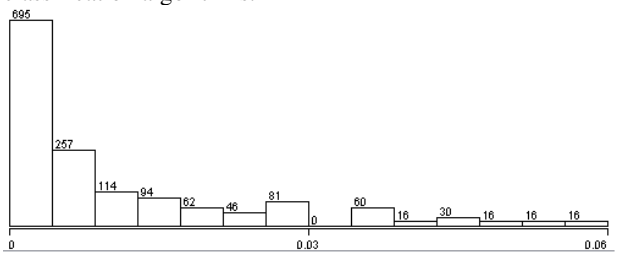


Fig. 6. A histogram of attribute "suction side displacement thickness" before pre-processing

**Scaled sound pressure level**: This output attribute (class) is a numerical type with the large variation range and does not has any missing value. Since this attribute is the output and the classification algorithms are applied to this attribute, some pre-processing operations are required. The variation range for this attribute is between 103.38 and 140.987 with the histogram of presented in Fig. 7.

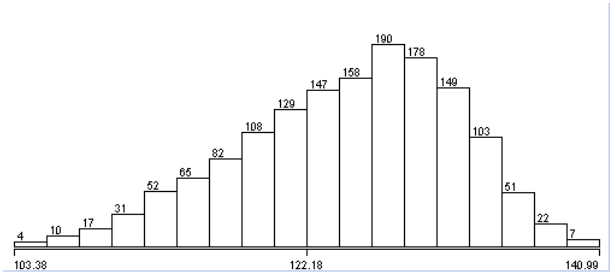


Fig. 7. A histogram of attribute "scaled sound pressure level" before pre-processing

In order to apply classification algorithms such as Support-Vector Machine (SVM), Bayesian Network, Random Forest, Decision Tree and etc. it is required that the output attribute is of nominal type, so it should be converted. But according to the numerical type of this attribute and uniqueness for all samples in the dataset (94% of the values are unique), prior to conversion, the grouping should be performed so that the number of classes is reduced. A simple way is to place adjacent values in a similar group. In this research, we use the standard deviation of the output attribute, which is equal to 6.899. Hence, the variations range, which is between 103.38 and 140.987, is divided to the classes with the length of 6.899 and the resulting values are fed to the Floor () function to obtain integer output values. Finally, six output classes with the histogram given in Fig. 8 would be generated. Similarly, in Fig. 9, the

histogram of all attributes is depicted in the classified form resulted from the pre-processing operation.

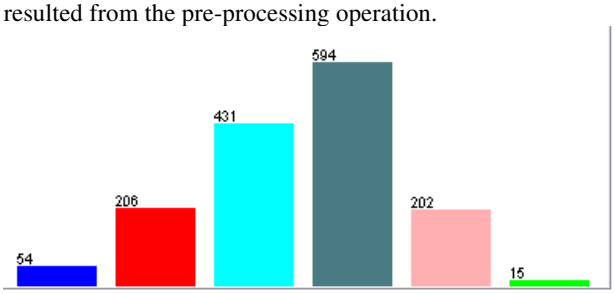


Fig. 8. A histogram of attribute "scaled sound pressure level" after pre-processing

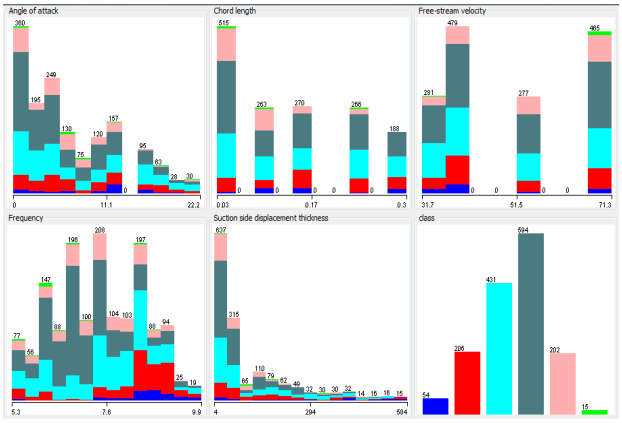


Fig. 9. Histogram of all attribute considering the class attribute

### C. *The Used Classification Algorithms*

In this section, we introduce the data mining algorithms that used to predict the noise level of aerodynamic devices [5, 16]:

**Bagging**: This algorithm was proposed in 1994 by Leo Breiman to improve the classification by combining randomly generated training sets. This methodology is a meta-algorithm designed to improve the stability and accuracy of machine learning algorithms used in statistical classification and regression. Variance is reduced and over-fitting is improved through the use of this algorithm. Although this method is used in the decision tree, it can be used in any kind of model. In fact, Bagging is a special case of model averaging approach.

**Random Forest**: Top-down decision trees are one of the most commonly used classification techniques, in which the samples are classified in such a way that the tree grows from the root to the bottom and eventually reaches the leaf nodes. The leaves of the decision tree are determined by a class and a set of solutions. To classify an input in this tree, the algorithm starts from the root and follows the branches according to the property of the input to reach a leaf. The output of the leaf value is considered as the class of the input element. The Random Forest algorithm is an example of decision tree algorithms with the advantage of high-precision classifier as well as conformity with a large number of inputs.

**J48**: This algorithm is the implementation of the C4.5 decision tree. In this algorithm, additional grafting branches are considered on a tree in a post-processing phase. The grafting process tries to capture some of the capabilities of ensemble methods such as Bagged and Boosted trees, while a single structure can be maintained. This algorithm identifies areas that are either empty or only contains misleading classified samples and explores another (alternative) class.

**RBF Network**: It is an artificial neural network that uses radial basis functions as activation functions. The output of this network is a linear combination of radial base functions for input parameters and neurons. This type of network is used in the approximation function, time series prediction, classification, and control systems, and is referred to as radial functions interpolation.

**Logistic**: This algorithm is another implementation for constructing and using a polynomial logistic regression model along with an edge estimator to protect against overfitting by penalizing large.

**Multi-Layer Perceptron (MLP)**: This algorithm is a class of feedforward artificial neural network which consists of at least three layers of nodes: an input layer, a hidden layer, and an output layer. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. The MLP utilizes a supervised learning technique called backpropagation for training.

**BayseNet**: The Bayesian classification assigns the membership value of each sample to the class with a probability, additionally, statistical concepts such as mean, standard deviation, or histogram of attributes are used for generating law. The Bayesian network is a graphical model that expresses the potential relationship between a set of variables. The structure of a Bayesian network is a directed graph without loops in which nodes represent random variables, and its edges represent a one-to-one relationship between variables. The implementation of this method is very simple and does not require complicated recursive parameter estimation procedures. That is, it can be used for a large dataset appropriately. It should be noted that this algorithm may not be the best possible classifier in a particular application, but its robustness and excellent performance are assured.

**Self-Organizing Map (SOM)**: This algorithm is a type of neural network model that is trained using unsupervised learning. The SOM maps the high-dimensional input vectors onto a two-dimensional grid of prototype vectors and orders them.

## III. Discussion and Simulation Results

One of the common tools used for evaluating classification algorithms is to employ the confusion matrix. As can be seen in Table 1, the confusion matrix includes results of predictions of the classifier algorithm in 4 different classes including True Positive, False Negative, False Positive and True Negative [17].

TABLE I
CONFUSION MATRIX

| | | Predicted | |
|---|---|---|---|
| | | True | False |
| Observed | True | TP | FN |
| | False | FP | TN |

Considering the confusion matrix, the following measures can be defined and evaluated [17]:

- **True Positive** refers to the positive samples that were correctly labeled by the classifier.
- **True Negative** refers to the negative samples that were correctly labeled by the classifier.
- **False Positive** is an error in data reporting in which a test result improperly indicates the presence of a condition, such as a disease (the result is *positive*), when in reality it is not present.
- **False Negative** is an error in which a test result improperly indicates no presence of a condition (the result is *negative*) when in reality it is present.
- **Precision** is the fraction of retrieved instances that are relevant:

$$\frac{TP}{TP+FP} \tag{1}$$

- **Accuracy** is the proportion of true results (both true positives and true negatives) among the total number of cases examined:

$$\frac{TP+TN}{TP+TN+FP+FN} \tag{2}$$

- **Recall** is the fraction of relevant instances that are retrieved:

$$\frac{TP}{TP+FN} \tag{3}$$

- **F-Measure** combines precision and recall (harmonic mean):

$$\frac{2\times Recall\times Precision}{Recall+Precision}=\frac{2\times TP}{2\times TP+FP+FN} \tag{4}$$

- **Root Mean Squared Error (RMSE)** is a frequently used measure of the differences between values predicted by a model or an estimator and the values observed. The RMSD represents the square root of the second sample moment of the differences between predicted values ($p_i$) and observed values ($a_i$) or the quadratic mean of these differences.

$$\sqrt{\frac{(p_1-a_1)^2+\cdots+(p_n-a_n)^2}{n}} \tag{5}$$

- **Mean Absolute Error (MAE)** measures how far predicted values ($p_i$) are away from observed values ($a_i$).

$$\frac{|p_1-a_1|+\cdots+|p_n-a_n|}{n} \tag{6}$$

In this research, the Weka tool is used to perform pre-processing operations and construct the proposed predictive models. This software has been developed at Waikato University in New Zealand and is an open-source tool implemented by the object-oriented programming (OOP) language. This tool includes several machine learning and data mining algorithms such as regression, classification, clustering, exploring association rules, pre-processing tools (filters), and selection methods for attributes.

In addition, to train and test the proposed method, *K*-fold (*K*=10) method is employed. In this type of test, data are classified into *K* subsets. From these *K* subsets, a subset is used for test and *K*-1 subsets are used for training. This procedure is repeated *K*-times and all data are once used for test and once for training. Finally, an average of these *K* times test is selected as the final estimation. In the *K*-fold method, the ratio of each class in each subset and in the main set is the same [17].

The results of the experiments are presented in Fig. 10 to Fig. 15 in terms of the model construction time, precision, F-measure, kappa, mean absolute error, and root mean squared error.

According to Fig. 10, the time of constructing the model based on the MLP algorithm and the proposed hybrid model based on Bagging and Random Forest algorithms are higher compared to other models. Also, the results in Fig. 11 show that the proposed hybrid model obtained from the Bagging and Random Forest algorithms with an accuracy of 77.5% yields superior performance over other models.

Moreover, as it is obvious in Fig. 12 and Fig. 13, the proposed hybrid model is superior in terms of F-measure and Kappa criteria over other models. These values are 77.4% and 0.6768, respectively.

Furthermore, according to the results shown in Fig. 14, it can be concluded that the Random Forest has a lower mean absolute error than the other models; while the proposed hybrid model has better performance in terms of root mean squared error, which is equal to 0.2279.

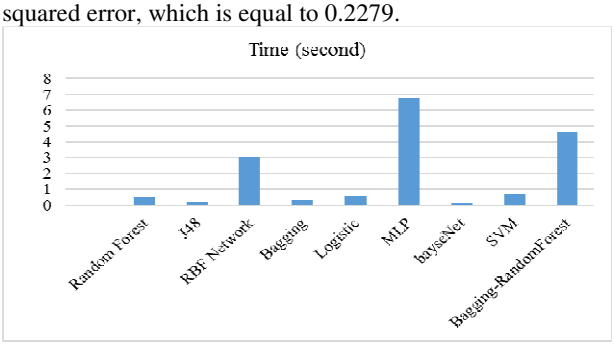


Fig. 10. Comparing the various predictive models in terms of the time needed to construct the model

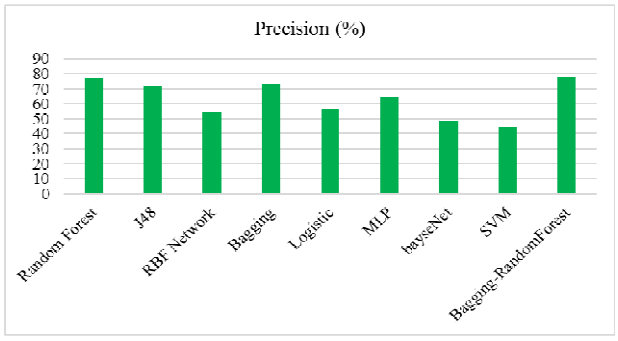


Fig. 11. Comparing the various predictive models in terms of precision

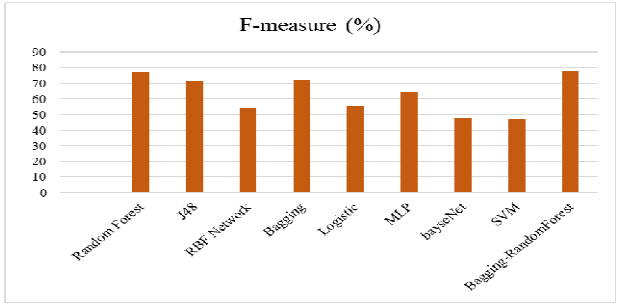


Fig. 12. Comparing the various predictive models in terms of precision

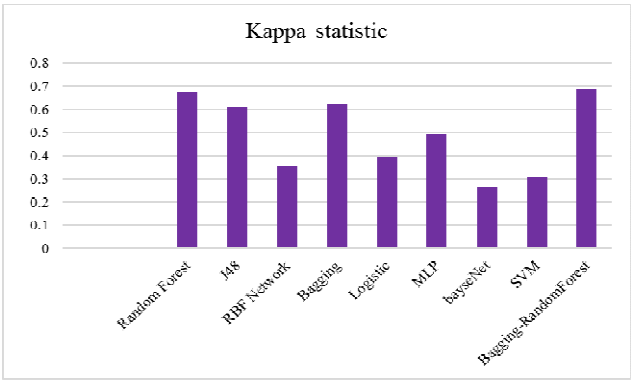


Fig. 13. Comparing the various predictive models in terms of precision

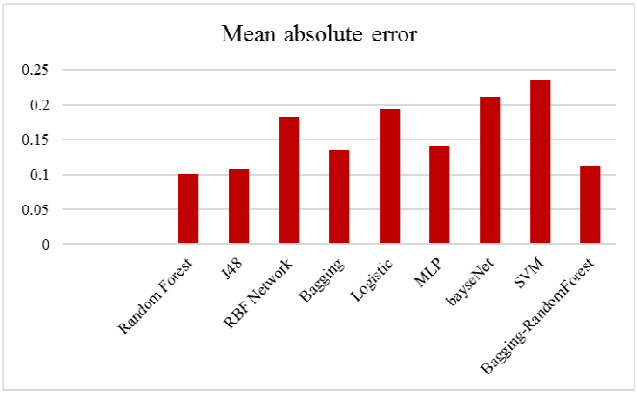


Fig. 14. Comparing the various predictive models in terms of precision

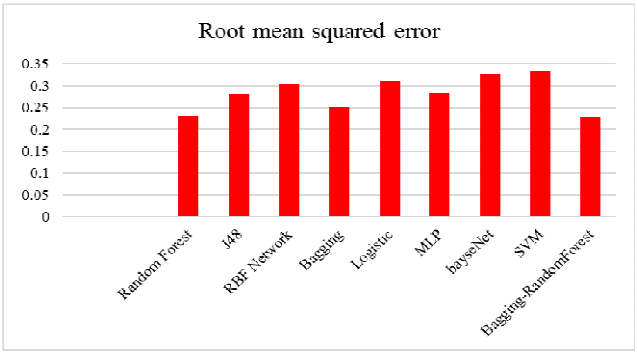


Fig. 15. Comparing the various predictive models in terms of precision

## IV. Conclusion

In this paper, data mining algorithms were used to predict the amount of noise emitted from aerodynamic objects. The dataset used in this study was named "Airfoil Self-Noise", which was collected by NASA. First, the required pre-processing was applied to this dataset. Then, using common classification methods, predictive models were generated and evaluated in Weka tool. The results showed that the proposed hybrid model derived from the Bagging and Random Forest algorithm is superior performance compared to other algorithms with an accuracy of 77.5%.

## References


[1] Amiet, R.K., 1975. Acoustic radiation from an airfoil in a turbulent stream. *Journal of Sound and vibration*, *41*(4), pp.407-420.

[2] Brooks, T.F., Pope, D.S. and Marcolini, M.A., 1989. Airfoil Self-Noise and Prediction. NASA Reference Publication, NASA-RP-1219.

[3] Schlinker, R. and Amiet, R., 1981. Helicopter Rotor Trailing Edge Noise. NASA CR-3470.

[4] Barnes, J. and Gomez. R., 2007. A variety of wind turbine noise regulations in the United States. Second International Meeting on Wind Turbines Noise, Lyon, France.

[5] Jiawei, H. and Micheline, K., 2006. Data Mining: Concepts and Techniques, Second Edition, Morgan Kaufmann Publishers, Publisher's name:Diane Cerra, Elsevier.

[6] Brooks, T.F. and Hodgson, T.H., 1981. Trailing edge noise prediction from measured surface pressures. *Journal of sound and vibration*, *78*(1), pp.69-117.

[7] Lutz, T., Herrig, A., Würz, W., Kamruzzaman, M. and Krämer, E., 2007. Design and wind-tunnel verification of low-noise airfoils for wind turbines. *AIAA journal*, *45*(4), pp.779-785.

[8] Roger, M., Moreau, S. and Wang, M., 2002. An analytical model for predicting airfoil self-noise using wall-pressure statistics. *Annual Research Brief, Center for Turbulence Research, Stanford University*, *2002*, pp.405-414.

[9] Errasquin, L., 2009. Airfoil self-noise prediction using neural networks for wind turbines (Doctoral dissertation, Virginia Tech).

[10] Moriarty, P., Guidati, G. and Migliore, P., 2005. Prediction of turbulent inflow and trailing-edge noise for wind turbines. In *11th AIAA/CEAS Aeroacoustics Conference* (p. 2881)

[11] Jones, L. and Sandberg, R., 2009. Direct numerical simulations of noise generated by the flow over an airfoil with trailing edge serrations. In *15th AIAA/CEAS Aeroacoustics Conference (30th AIAA Aeroacoustics Conference)* (p. 3195)

[12] Gerhard, T. and Carolus, T., 2014. INVESTIGATION OF AIRFOIL TRAILING EDGE NOISE WITH ADVANCED EXPERIMENTAL AND NUMERICAL METHODS. In *The 21st International Congress on Sound and Vibration*

[13] Wang, M., Moreau, S., Iaccarino, G. and Roger, M., 2009. LES prediction of wall-pressure fluctuations and noise of a low-speed airfoil. *International journal of aeroacoustics*, *8*(3), pp.177-197.

[14] Lee, S., Lee, S. and Lee, S., 2013. Numerical modeling of wind turbine aerodynamic noise in the time domain. *The Journal of the Acoustical Society of America*, *133*(2), pp. EL94-EL100.

[15] Croaker, P., Kessissoglou, N., Karimi, M., Doolan, C. and Chen, L., 2014. Self-noise prediction of a flat plate using a hybrid RANS-BEM technique. Inter-noise, Melbourne, Australia.

[16] Tan, P.N., 2018. *Introduction to data mining*. Pearson Education India.

[17] Poor, S.S.A. and Shiri, M.E., 2017. A Genetic Programming based Algorithm for Predicting Exchanges in Electronic Trade using Social Networks' Data. *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS*, *8*(5), pp.189-196.