UNIVERSITY OF WOLVERHAMPTON

ACADEMIC YEAR: 2020/2021

Discovery Coursework: Chemometrics to predict different properties

DENSITY PREDICTION OF [BMIM][PF6] USING ARTIFICIAL NEURAL NETWORK

NAME: MEMEL ANGE PATRICK THEOPHANE AGNIMEL

STUDENT NUMBER: 1902655

Table of Contents

Abstract

This study aims to develop an Artificial Neural Network based QSPR model to predict the density of 1-Butyl-3-methylimidazolium hexafluorophosphate [Bmim][PF6] at various temperatures based on an extensive database gathered from literature published works. The database includes 131 density data points at different temperatures and water content. The model was developed using MATLAB Deep Learning Toolbox. The developed model was trained using *the Levenberg Marquardt* algorithm. The predictions of the developed model were analysed by various methods, including both statistical and graphical approaches. Results show that the developed model accurately predicts the experimental data with an overall R^2 and MSE values of 0.996% and 9.237×10^{-7}, respectively. However, the developed ANN has limited prediction generalisation capability since it only applies to one ionic liquid.

Keywords: QSPR (Quantitative structure-activity relationship); QSAR (Quantitative structure-activity relationship); algorithm, ionic liquid; model; ANN (Artificial Neural Network); MSE (Mean squared error); R^2 (Coefficient of correlation)

Introduction

The model to be developed has for objective the prediction of the thermo-physical properties of 1-Butyl-3-methylimidazolium hexafluorophosphate for CO2 capture. 1-Butyl-3-methylimidazolium hexafluorophosphate [Bmim][PF6], and ionic liquids, in general, have been emerging solvents within carbon capture technologies as alternatives for amine-based solvents which have significant usability and practicality issues. The property investigated in this work would be the density. This work is complementary to papers submitted by Teka Bi T. and Kire, Cedric E. (2021), in which they developed models to predict the other thermo-physical properties such as Viscosity and Heat Capacity. This study is similar to one conducted by Ali Mazari & al. (2021), in which they used the Gaussian process regression (GPR) and Support vector machine (SVM) to build a prediction model for the same end purpose. In this series of work, a different approach shall be implemented to take advantage of the adaptivity of ANN-based models that find their foundation in artificial intelligence and machine learning. The steps in developing and building the ANN will be presented in this work. The result and the performance of the ANN will be analysed as well.

I. Literature reviews

Several prediction models for predicting the density of ionic liquids have been reported in the literature. Two of them will be the object of reflection in this section.

Ali Barati-Harooni et al. (2016) had created a Hybrid-ANFIS (adaptive neuro-fuzzy inference system) model to predict the density of ionic liquids for a range of temperatures. To develop their model, 602 experimental density data for 146 ionic liquids at different temperatures were used from several published works. Ali Barati-Harooni et al. were to build a model that uses machine learning algorithms (Intelligent models) as those have a solid foundation and are generally more accurate than other methods, making them more reliable. Still, those models are underexploited. Ali Barati-Harooni et al. built the initial structure of their model using the MATLAB genfis2 function, and then the model was trained using a hybrid training approach to provide more accurate performance. The training process consisted of 20 stages, each composed of 60 epochs. After establishing their model, Ali Barati-Harooni et al. evaluated the performance of their model by comparing them with experimental data through graphs and statistical tools. The experimental data were also compared with other literature correlations, which allow for comparing the performance of the developed model with those literature correlations. Results indicated that the model developed by Ali Barati-Harooni et al. expresses accurate and reliable results, surpassing the literature correlations' performance. Results show that the developed model accurately predicts the experimental data with an overall R^2 of 0.985. Since the developed model is based on a wide-ranging set of literature data covering as many as 146 Ionic liquids, it is very inclusive and can be in fields where high accuracy is required or no data is available.

The second model was developed by Mohanad El-Harbawi et al. (2014). They have developed a new mathematical model for predicting the densities of ionic liquids using the quantitative structure-property relationship (QSPR) approach. They built the algorithms for their model using MATLAB software. The development of the model was based on molecular descriptors, which give the relationship between the density of each ionic liquid and its molecular structures. To elucidate the different parameters for the model, they wrote the code using a combination of the multiple linear regression (MLR) method and polynomial equation.
The model was developed based on a data set containing 918 experimentally measured density data for 747 pure ionic liquids, quoted from Shen et al. (2011). The model was trained using the least square error method. The MATLAB code was programmed to exploit 50% of the whole data set for training the algorithm and use the remaining 50% of the data with the obtained coefficients from the training data set to test the model's performance that has just been trained. The prediction of the developed model was validated against the experimental data and showed no substantial deviation. The results showed that the model could predict IL densities

with very high accuracy at a correlation of determination value R^2 value of 99.5 %. This model is also based on a broad range set of experimental data covering as many as 747 ionic liquids. It is inclusive and can be used in fields where high accuracy is needed or no data is available.

II. A.I and Machine Learning in QSPR Modelling

Artificial intelligence is described as intelligence demonstrated by machines. This term is used when a machine shows cognitive behaviour normally attributed to humans, such as learning or problem-solving. Machine learning is a subset of Artificial Intelligence, in which a computer can learn and make decisions through the use of data and algorithms. Artificial Intelligence and Machine Learning play a crucial role in modelling and discovery in science. Numerous applications in property or activity predictions like physicochemical and ADMET properties have recently emerged and reinforced the significance and importance of AI in QSPR/QSAR models. These are prime examples of AI-based models, particularly in artificial neural networks such as deep neural networks or recurrent networks. Other QSPR/QSAR modelling methods are available depending on the need and application. Those are summarised in table 1 below with their principal characteristics.

Table 1. QSPR/QSAR Models

Methods	Linear/ Non-Linear	Classification/ regression task	Advantages	Disadvantages
k-NN	Linear	Classification	Simple	Unstable and unreliable
MLR	Linear	Regression	Simple	Limitation for data with huge numbers of features
PLS	Linear	Both	Performs well on data with huge number of features	Linear model
ANN	Non-Linear	Both	performs well on non-linear data	Black-box method
SVM	Non-Linear	Both	Most powerful methods for both classification and regression	Black-box method
DT	Non-Linear	Classification	Highly interpretable	Requires a large number of training instances
RF	Non-Linear	Both	More confident estimate	Computationally intensive

Abbreviations: multiple linear regressions (MLR), partial least square (PLS), k-nearest neighbor (k-NN), artificial neural network (ANN), support vector machine (SVM), decision tree (DT), and random forests (RF).

III. Artificial Neural Network Development and Structure

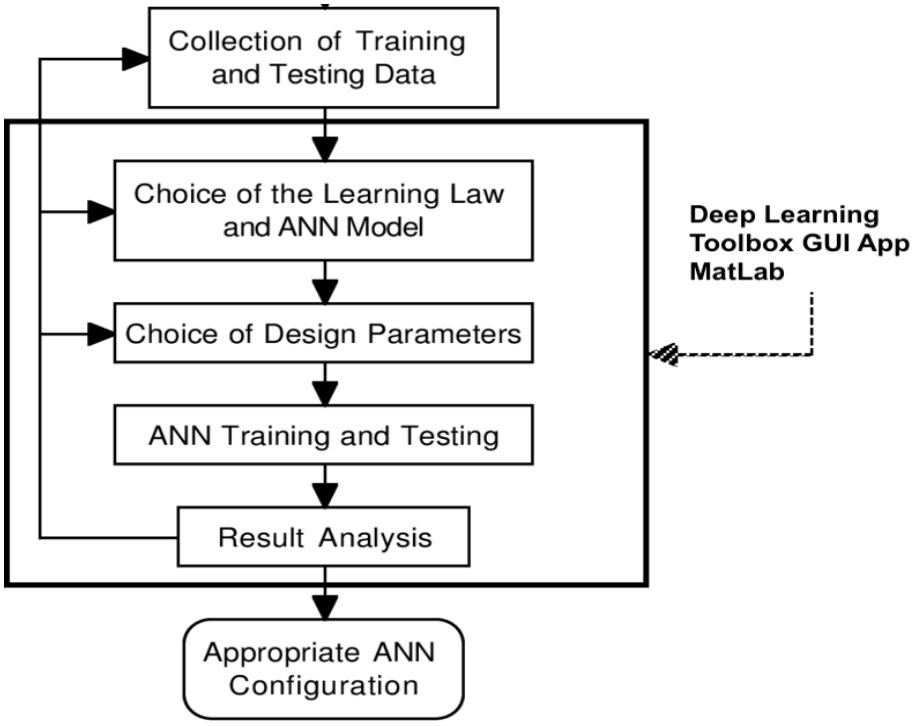

Figure 1. Artificial Neural Network Development Steps in MATLAB Deep Learning Toolbox

Collection of Training and Testing Data (Selecting and Dividing data)

The model was developed based on a data set containing 131 experimentally measured density data for 1-Butyl-3-methylimidazolium hexafluorophosphate at atmospheric pressure for different temperatures and water content published in several pieces of literature. Data collection is important when building a self-learning model. For instance, an ANN model trained with incorrect or/and incomplete data, shall also deliver incorrect or/and incomplete data. Misleading data will cause misleading results. And Individuals creating algorithms might not know that the data they feed is misleading until it is too, and their model has already caused damage. Data diversity and accuracy are important factors.

(See excel files for data)

Choices of Learning Law and ANN Model

The development of the Artificial Neural Network is based on simple numerical descriptors. The ANN is supposed to find the relationship between the density of [Bmin][PF6] and its physical conditions. Ideally, at a given water content and temperature, the ANN should provide a prediction for density. Therefore, the most appropriate learning law shall be the "Input-Output and Curve fitting" model. Thus, the MATLAB Neural Fitting Code "n*ftool*" was used.

Choices of Design Parameters

The ANN model was programmed to exploit 70% of the whole data set to train the algorithm and use the remaining 30% of the data for validation and testing; each using 15% of the data.

The structure of the ANN is presented below.

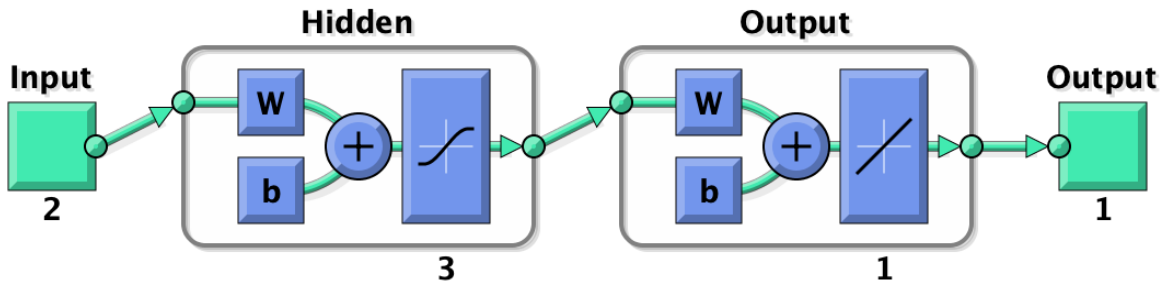

Figure 2. Graphical of The Neural Network

The ANN has the following characteristic

- One 1x2 matrix input with temperature and water content in each column
- Three hidden layers
- One output element which encloses the density

The question of how many hidden layers and hidden nodes is a pertinent issue that always rises for any classification/prediction tasks. However, there is no definitive answer as of today. The available methods rely on trial and errors, assumptions, and other educated guesses.

The selection of the number of hidden layers was based on the complexity of the problem and trial errors. Above 3 layers there is little to no improvement in the performance of the ANN.

The MATLAB Algorithm automatically deduced the number of hidden nodes in each hidden layer.

However, if the ANN was meant to be built from scratch, the number of hidden nodes would have been deducted using one of the several methods mentioned in the literature.

One method assumes a backpropagation NN configuration is 1-m-n and makes deducting based on the number of nodes in the input and output layers.

Another method of selection through **trials and errors**. three approaches are cited in the literature.

Forward Approach: This approach begins by selecting a small number of hidden neurons. We usually begin with two hidden neurons. After that train and test the neural network. Then increased the number of hidden neurons. Repeat the above procedure until training and testing improved.

Backward Approach: This approach is the opposite of the Forward approach. In this approach, we start with a large number of hidden neurons. Then train and test the NN. After that, gradually decrease the number of hidden neurons and train and test the NN again. Repeat the above process until training and testing improved.

Two-phase method: In this method data set is divided into four groups. Among all four groups, two groups of data are used in the first phase to train the network and one group of remaining data set is used in the second phase to test the network. The last group of data set is used to predict the output values of the train network. This experiment is repeated for different neurons to get the minimum number of error terms for selecting the number of neurons in the hidden layer.

Another method which is also like trials and errors is **sequential orthogonal approach**. This approach is about adding hidden neurons one by one. Initially, increase *Nh (hidden nodes)* sequentially until the error is appropriately small. When adding a neuron, the new information introduced by this neuron is caused by that part of its output vector which is orthogonal to the space spanned by the output vectors of previously added hidden neurons. An additional advantage of this method is that it can be used to build and train neural networks with mixed types of hidden neurons and thus to develop hybrid models

There is also a **rule of thumb** method to help determine that number. The method suggests:

- The number of hidden neurons should be in the range between the size of the input layer and the size of the output
- The number of hidden neurons should be 2/3 of the input layer size, plus the size of the output layer
- The number of hidden neurons should be less than twice the input layer size

ANN Training and Testing

The artificial Neural Network was trained using *the Levenberg Marquardt* algorithm. The Levenberg-Marquardt algorithm was developed in the early 1960s to solve nonlinear least-squares problems. This algorithm typically requires more memory but less time. Training automatically stops when generalisation stops improving, as indicated by an increase in the mean square error of the validation samples. The Levenberg-Marquardt algorithm combines two numerical minimisation algorithms: the gradient descent method and the Gauss-Newton method. When the current solution is far from the correct one, the algorithm behaves like a steepest descent method: slow but guaranteed to converge. When the current solution is close to the correct solution, it becomes a Gauss-Newton method.

Result Analysis

After training the MATLAB algorithm provides the results regarding the training and performance of the ANN model. Those results will be summarised in section IV.

Appropriate ANN Configuration

A deployable solution of the trained was generated and the code was submitted along with this paper. The generated code is a MATLAB matric-Only Function. This deployable can then be tested in real applications using new data that were used not used to train the ANN. (see attached .m file)

IV. Results and ANN deployment

The developed ANN for predicting the density of 1-Butyl-3-methylimidazolium hexafluorophosphate IL has good correlating and predictive ability. The statistical parameters presented in table 2. support that statement. The training process was carried out using the Levenberg Marquardt algorithm and was composed of 32 epochs. The best validation performance was achieved at epoch 29 with an MSE value of 1.32×10^{-6}

Data division: Random
Training: Levenberg Marquardt
Performance: Mean Squared Error

Table 2. statistical parameters for ANN model

Dataset	Samples	RMSE	R^2
Train data	91	9.200×10^{-7}	0.972
Validation data	20	1.320×10^{-6}	0.996
Testing data	20	7.288×10^{-7}	0.978
Total data	131	9.237×10^{-7}	0.996

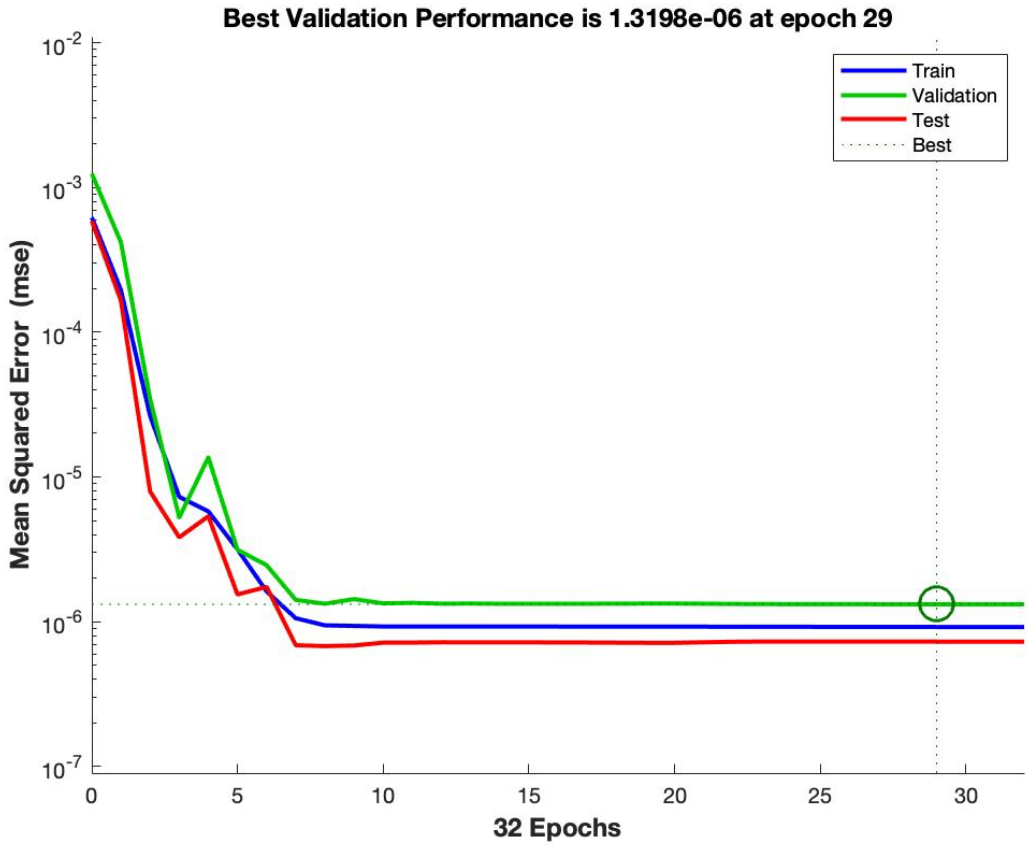

Figure 3. Performance plot, MSE vs. epochs

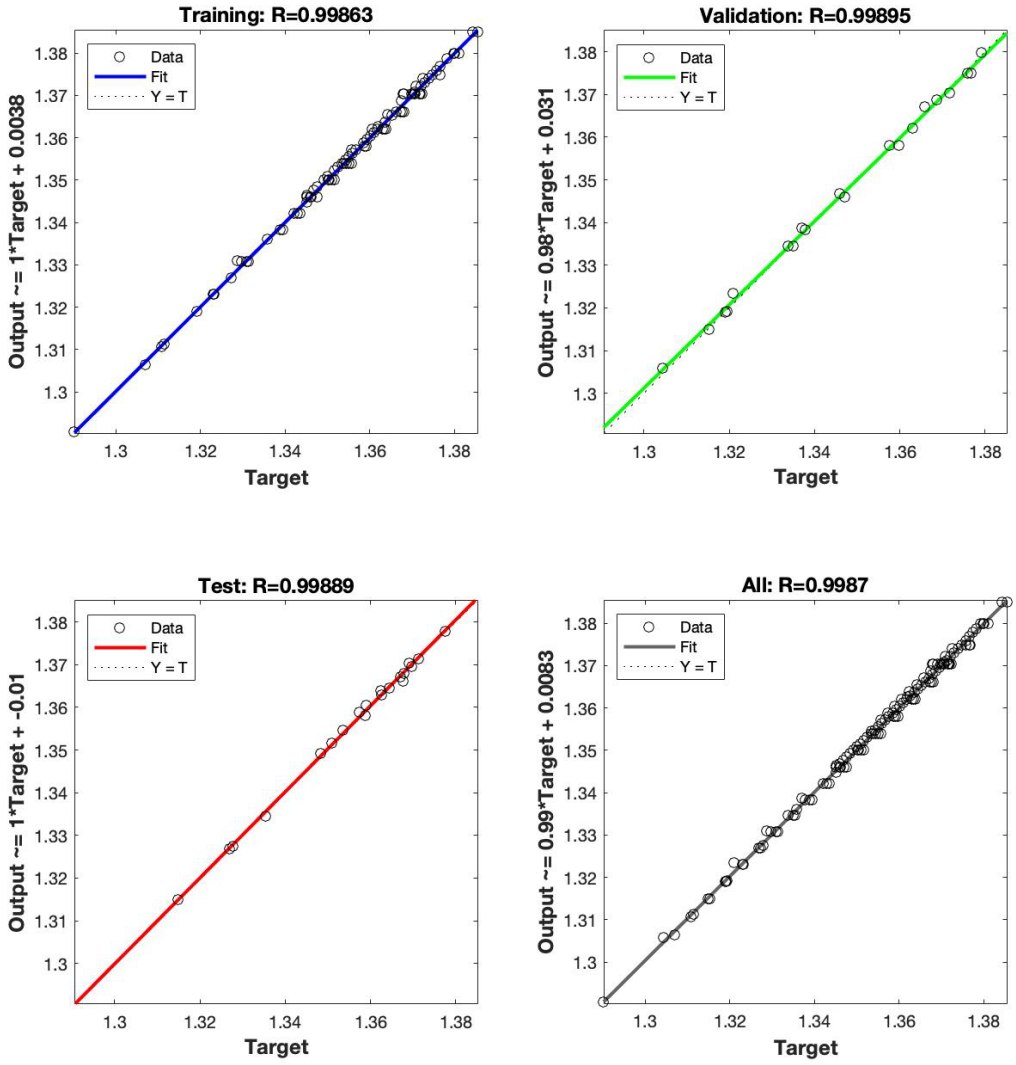

Figure 4. Cross plot of predictions

Figure 4. shows the cross plots of the model constructed by plotting the predicted density data against the experimental values. This figure indicates an appropriate uniformity between the predicted data and target density values because the data points are distributed close to the Y = X line which indicates good agreement between the target data and the predictions of the ANN model.

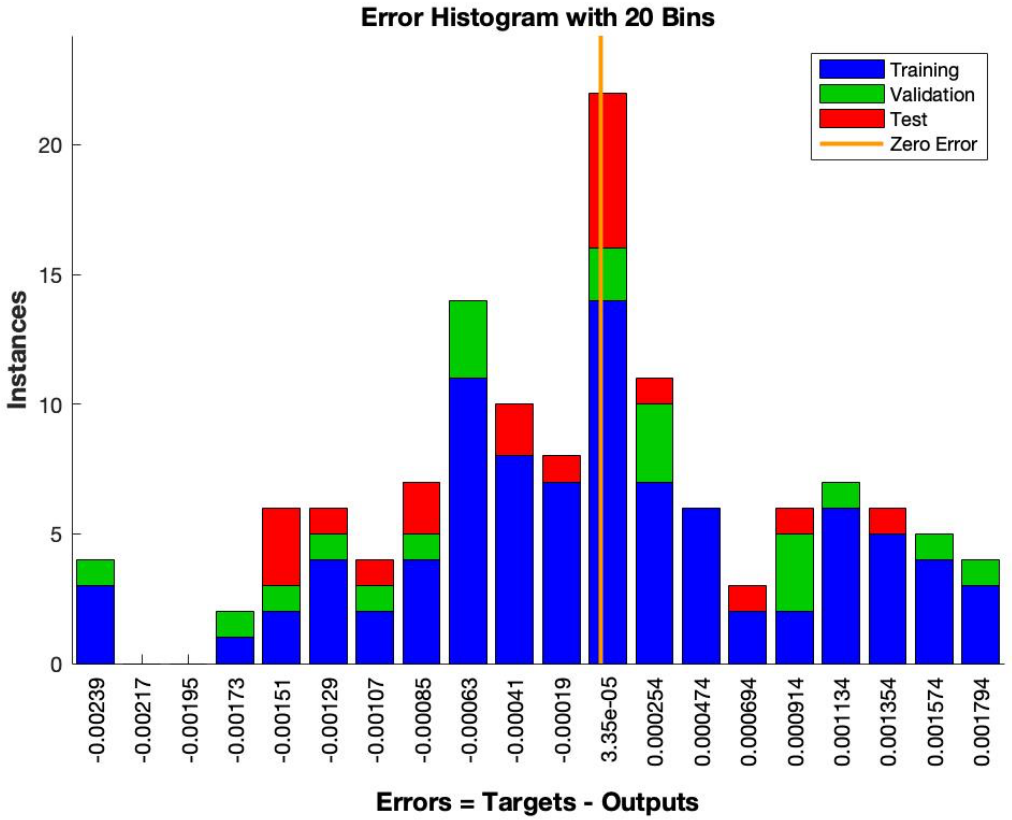

Figure 5. Error histogram

Figure 5. shows the error histogram of the ANN model in which the vertical axis denotes the instances of data point, and the horizontal axes represent the difference between the target (literature) and output data (predicted). This figure also shows that the exhibited errors by the model are mostly distributed near the zero error line which overall indicates the reliability and accuracy of the developed model. The maximum error of the developed model is no more than 1%

Conclusion

In this work, an ANN model was developed to predict the density of 1-Butyl-3-methylimidazolium hexafluorophosphate [Bmim][PF6] at various temperatures based on an extensive database gathered from literature published works. The model was developed using the MATLAB Deep Learning Toolbox. The structure of ANN was trained using *the Levenberg Marquardt* algorithm. Different statistical and graphical methods were utilised to evaluate the performance and predictions of the developed model. Results showed that the developed model presents accurate and dependable results as the developed model accurately predicts the experimental data with an overall R^2 and MSE values of 0.996%, and 9.237×10^{-7} and respectively., the developed ANN has limited predicted generalisation capability since it only applies to one ionic liquid and for limited condition. Further improvement could implement by building the ANN without the help of the MATLAB Deep Learning Toolbox, as it will allow more flexibility in designing the ANN learning algorithm, and structure.

References

Hessler, G., & Baringhaus, K. H. , 2018. Artificial Intelligence in Drug Design. *Molecules (Basel, Switzerland)*, *23*(10), 2520. https://doi.org/10.3390/molecules23102520

Foram S. Panchal *et al*, 2014. Review on Methods of Selecting Number of Hidden Nodes in Artificial Neural Network International Journal of Computer Science and Mobile Computing, Vol.3 Issue.11, November- 2014, pg. 455-464

Prachayasittikul, Veda & Worachartcheewan, Apilak & Shoombuatong, Watshara & Songtawee, Napat & Simeon, Saw & Prachayasittikul, Virapong & Nantasenamat, Chanin, 2015. Computer-Aided Drug Design of Bioactive Natural Products. Current Topics in Medicinal Chemistry. 15. 1780-1800. 10.2174/1568026615666150506151101.

Ali Barati-Harooni, Adel Najafi-Marghmaleki, Amir H Mohammadi, 2016. ANFIS modeling of ionic liquids densities. Journal of Molecular Liquids,Volume 224, Part A, Pages 965-975, ISSN 0167-7322, https://doi.org/10.1016/j.molliq.2016.10.050.

l-Harbawi, M., Samir, B.B., Babaa, MR. *et al, 2014.* A New QSPR Model for Predicting the Densities of Ionic Liquids. *Arab J Sci Eng* **39,** 6767–6775. https://doi.org/10.1007/s13369-014-1223-3

SALGADO, J., REGUEIRA, T., LUGO, L., VIJANDE, J., FERNÁNDEZ, J. and GARCÍA, J., 2014. Density and viscosity of three (2,2,2-trifluoroethanol + 1-butyl-3-methylimidazolium) ionic liquid binary systems. Journal of Chemical Thermodynamics, 70, pp. 101-110.

TRONCOSO, J., CERDEIRIÑA, C.A., SANMAMED, Y.A., ROMANÍ, L. and REBELO, L.P.N., 2006. Thermodynamic properties of imidazolium-based ionic liquids: Densities, heat capacities, and enthalpies of fusion of [bmim][PF6] and [bmim][NTf2]. Journal of Chemical and Engineering Data, 51(5), pp. 1856-1859.

KUMAR, A., 2008. Estimates of internal pressure and molar refraction of imidazolium based ionic liquids as a function of temperature. Journal of Solution Chemistry, 37(2), pp. 203-214.

ALMEIDA, H.F.D., LOPES, J.N.C., REBELO, L.P.N., COUTINHO, J.A.P., FREIRE, M.G. and MARRUCHO, I.M., 2016. Densities and viscosities of mixtures of two ionic liquids containing a common cation. Journal of Chemical and Engineering Data, 61(8), pp. 2828-2843.

JACQUEMIN, J., HUSSON, P., PADUA, A.A.H. and MAJER, V., 2006. Density and viscosity of several pure and water-saturated ionic liquids. Green Chemistry, 8(2), pp. 172-180.

KUMEŁAN, J., KAMPS, A.P.-., TUMA, D. and MAURER, G., 2005. Solubility of CO in the ionic liquid [bmim][PF6]. Fluid Phase Equilibria, 228-229, pp. 207-211.

PEREIRO, A.B., LEGIDO, J.L. and RODRÍGUEZ, A., 2007. Physical properties of ionic liquids based on 1-alkyl-3-methylimidazolium cation and hexafluorophosphate as anion and temperature dependence. Journal of Chemical Thermodynamics, 39(8), pp. 1168-1175.

MONTALBÁN, M.G., BOLÍVAR, C.L., DÍAZ BAÑOS, F.G. and VÍLLORA, G., 2015. Effect of Temperature, Anion, and Alkyl Chain Length on the Density and Refractive Index of 1-Alkyl-3-methylimidazolium-Based Ionic Liquids. Journal of Chemical and Engineering Data, 60(7), pp. 1986-1996.

HUO, Y., XIA, S. and MA, P., 2007. Densities of ionic liquids, l-butyl-3-methylimidazolium hexafluorophosphate and l-butyl-3-methylimidazolium tetrafluoroborate, with benzene, acetonitrile, and 1-propanol at T = (293.15 to 343.15) K. Journal of Chemical and Engineering Data, 52(5), pp. 2077-2082. R77R

KRISHNA, TS, RAJU, KTSS, GOWRISANKAR, M., NAIN, AK, MUNIBHADRAYYA, B. and MOL, J., 2016. Liquid, 216, pp. 484-495.