A Review of Online Sequential Extreme Learning Machines

ChangHui Deng¹, XiaoJun Wang¹,²,*, Jun Gu¹ and Wei Wang¹

¹College of Information Engineering, Dalian Ocean University, No.52 Heishijiao St., Shahekou District, Dalian, 116023, China
²School of Management Science and Engineering, Dongbei University of Finance and Economics, No.217 Jianshan St., Shahekou District, Dalian, 116025, China
Email: wxjjeslicx0903@126.com
*Corresponding Author

Abstract. One of the challenges for machine leaning on big data is the effective and efficient leaning of large-scale and on-going explosion data which is always with the concept drift problem. To meet the challenge, learning algorithms/techniques performed well on large-scale data and also with the evolvable property are desired. The OS-ELM family has strong potential as viable alternative techniques for the computation of large-scale and on-going explosion data in more fields of applications/tasks. This work reviews the most important and latest works in OS-ELM family. The review consists of two topics, one related to the improved version of OS-ELM which aims at overcoming the disadvantages of OS-ELM, and the other related to the extended version the goals of which is to add some specialties to OS-ELM. It is expected that the review will support a certain research in the future.

1. Introduction

With the rapid development of computer science, data acquisition and storage techniques, and so on, the era of big data has been coming. In 2001, Douglas Laney defines big data in terms of “3Vs” characteristics, i.e., variety, volume and velocity, in Meta Group (i.e., Gartner) [1]. In 2016, Mingmin Chi et al. give a further explanation in Proceedings of the IEEE [2], and mention that “value” indicates the quality of big data, not a defining characteristic. Now, big data become more and more common in various areas, such as society, commerce, science industry. The effective use of big data had made our lives easier and better.

As the combination of computer science and statistics, machine learning has become one of the most important parts for the effective utilization of big data. On the basic of machine leaning, the applications of big data rapidly and continuously appears in many fields, such as, remote sensing [2], air pollution [3], wind tunnel measurements [4], imaging genetics[5], and so on. However, big data create many challenges for machine learning [6-9]. One of these challenges is the effective and efficient leaning of large-scale and on-going explosion data. Besides the issue of large scale size, this kind of data is always with the problem of concept drift in many real applications/tasks. Concept drift means that the statistical properties of target variable changes if environment changes or time goes by [9-11]. To meet the challenge, learning algorithms/ techniques performed well on large-scale data and also with the evolvable property [9] are desired.

Online Sequential Extreme Learning Machine (OS-ELM) [12] as a famous recursive leaning algorithm can fast and accurately learn large-scale data flows and provide effective and efficient solutions to the problem of concept drift. It is an online version of Extreme Learning Machine (ELM) (proposed by Guang-Bin Huang et al.) [13-14]. OS-ELM can learn data one-by-one or chunk-by-chunk with fixed or varying chunk size. Inheriting all the advantages of ELM, such as, low
computational burden, strong nonlinear-fitting ability, without local minimum and easy to implement, OS-ELM and its improved or extended versions have drawn attractions in some applications of large-scale/big data learning [15-17]. Consequently, they have strong potential as viable alternative techniques for the computation of large-scale and on-going explosion data in more fields of applications/tasks.

This work reviews the most important and latest works in OS-ELM family, and it is expected that the review will support a certain research in the future. The review consists of two topics, one related to the improved version of OS-ELM which aims at overcoming the disadvantages or limitations of OS-ELM, and the other related to the extended version, the goals of which is to extend suitable fields of OS-ELM via adding some specialties. The organization of this paper is as follows. In Section 2, the mathematical description of OS-ELM is rewritten. The topics of the improved and the extended versions of OS-ELM are stated in Section 3. The conclusion is summarized in Section 4.

2. The Mathematical Description of OS-ELM

ELM is a simple Single-Layer Feed-Forward Neural networks (SLFNs). Given the number of hidden nodes \( J \), \( M \)-dimensional input data \( x \) will be mapping to a \( J \)-dimensional feature space via an active function \( g(a,b,x) \) where \( a \) and \( b \) are the parameters of hidden nodes. ELM randomly generates the parameters of hidden nodes. The mathematical description of the output of ELM is,

\[
f(x) = \sum_{i=1}^{J} \beta_i g(a_i, b_i, x)
\]

or

\[
f(x) = h\beta
\]

where \( h \) is the vector of hidden-layer outputs, and \( h = [g(a_1, b_1, x), g(a_2, b_2, x), \ldots, g(a_J, b_J, x)] \). \( \beta \) is the vector of hidden-layer weights, and \( \beta = [\beta_1, \beta_2, \ldots, \beta_J]^T \).

Given the activation function (e.g., ‘RBF’, ‘Sigmoid’, or ‘Sine’), and random HPs (i.e., \( a \) and \( b \)), the goal of learning of \( N \) samples \( \{(x_1, t_1), (x_2, t_2), \ldots, (x_N, t_N)\} \) is to estimate \( \beta \), where \( t_n \) is the target corresponding to the input vector \( x_n \), and \( x_n = [x_{n,1}, x_{n,2}, \ldots, x_{n,M}] \). \( M \) is the dimension of input attributes. For classification tasks, \( t \in \{c_1, c_2, \ldots, c_q\} \), and for regression tasks, \( t \in \mathbb{R} \).

The estimation of \( \beta \) in ELM is the solution to

\[
\min ||H\beta - T||
\]

where \( H \) is the matrix of hidden-layer outputs, and \( H = [h_1, h_2, \ldots, h_N] \). \( T \) is the target vector, and \( T = [t_1, t_2, \ldots, t_N]^T \). In terms of the least square method, the estimation of \( \beta \) in ELM is the solution of (4),

\[
H\beta = T
\]

\( \beta \) can be calculated by the Moore-Penrose generalized inverse of \( H \), and the estimation of \( \hat{\beta} = H^+T \), and \( H^+ \) is the pseudo-inverse of \( H \). When \( HTH \) is nonsingular,

\[
H^+ = (H^T H)^{-1}H^T,
\]

and

\[
\hat{\beta} = (H^T H)^{-1}H^T T
\]
In order to learn a data flow, Liang et al. has proposed OS-ELM [12], which can learn the data one-by-one or chunk-by-chunk with fixed or varying chunk size. Given an initial chunk of data $\Omega_0$ with $N_0$ samples $\{(x_1,t_1),(x_2,t_2),\ldots,(x_{N_0},t_{N_0})\}$, the solution to (3) is given by

$$\hat{\beta}_0 = \mathbf{K}_0^{-1}\mathbf{H}_0\mathbf{T},$$

where

$$\mathbf{K}_0 = \mathbf{H}_0^\top\mathbf{H}_0$$

Suppose now that another chunk of data $\Omega_1$ with $N_1$ samples is given, and the problem becomes minimizing

$$\begin{bmatrix} \mathbf{H}_0 & 0 \\ \mathbf{H}_1 \\ \mathbf{T}_1 \end{bmatrix}$$

For sequential learning, $\hat{\beta}_1$ is expressed as a function $\hat{\beta}_0$. That is,

$$\hat{\beta}_1 = \hat{\beta}_0 + \mathbf{K}_1^{-1}\mathbf{H}_1^\top(\mathbf{T}_1 - \mathbf{H}_1\hat{\beta}_0),$$

where

$$\mathbf{K}_1 = \mathbf{K}_0 + \mathbf{H}_1^\top\mathbf{H}_1$$

Generalizing the previous arguments, when the $k$-th chunk of data $\Omega_k$ with $N_k$ samples is received, the solution to (3) is given by

$$\hat{\beta}_k = \hat{\beta}_{k-1} + \mathbf{K}_k^{-1}\mathbf{H}_k^\top(\mathbf{T}_k - \mathbf{H}_k\hat{\beta}_{k-1})$$

where

$$\mathbf{K}_k = \mathbf{K}_{k-1} + \mathbf{H}_k^\top\mathbf{H}_k$$

### 3. The Most Important and Latest Members of OS-ELM Family

Since proposed in 2006, OS-ELM has drawn much attraction in theory, techniques and applications. The OS-ELM family continues to expand. In term of the aims of the OS-ELM family members, the review of this paper is divided into two topics. One topic is related to the improved version of OS-ELM which aims at overcoming the disadvantages or limitations of OS-ELM. The other topic is related to the extended version of OS-ELM, the goal of which is to extend suitable fields of OS-ELM via adding some specialties.

#### 3.1. The Improved OS-ELMs

Firstly, OS-ELM is based on the criterion of empirical risk as shown in (2), and with weak robustness on noisy data flows. Moreover, the least square solution of (2) sometimes obtains singular and ill-posed $\mathbf{H}\mathbf{T}$, and then leading to a bad or fail OS-ELM. To avoid singular and ill-posed issue and improve the robustness of OS-ELM, Huynh and Won (2011) [18] present Regularized OS-ELM (ReOS-ELM) which add structural risk to OS-ELM, and the estimation of $\beta$ in ReELM is the solution to

$$\min \|\mathbf{H}\beta - \mathbf{T}\|^2 + \lambda \|\beta\|^2$$

The second part of (13) denotes structural risk, and $\lambda$ is the Regularization Factor (RF), $\lambda \geq 0$. Given an initial chunk of data $\Omega_0$. The solution to (13) is given by
\[ \hat{\beta}_0 = \mathbf{K}_0^{-1} \mathbf{H}_0^T \mathbf{Y}_0 \]  

(14)

where

\[ \mathbf{K}_0 = \mathbf{H}_0^T \mathbf{H}_0 + \lambda \mathbf{I} \]  

(15)

and \( \mathbf{I} \) is an unit matrix with size \( J \times J \).

Compare (7) of OS-ELM to (14) of ReOS-ELM, the difference is that \( \lambda \mathbf{I} \) is added to \( \mathbf{K}_0 \) of OS-ELM. OS-ELM is a special case of ReOS-ELM with \( \lambda = 0 \). When \( \lambda > 0 \), ReOS-ELM has the same mathematical description of \( \hat{\beta}_0 \) as OS-ELM. Moreover, Gu (2014) et al. [19] propose Constraint OS-ELM (COS-ELM) which is also based on both empirical risk and structural risk. Differently from ReOS-ELM, COS-ELM adds a tuning parameter to empirical risk, not structural risk.

Secondly, OS-ELM randomly generates the Hidden-node Parameters (HPs) of the initial ELM, and dose not changes them during updating. Although speeding up learning, HPs brings instability. In order to enhance the stability of OS-ELM, Lan et al. (2009) presented Ensemble OS-ELM (EOS-ELM) [20] which constructs many OS-ELM sub-models with different HPs and combing their outputs to obtain an ensemble one. Huang et al. (2016) [16] proposed an ensemble OS-ELM framework supports any combination of bagging, subspace partitioning and cross validation. Further, they design a Parallel Ensemble of OS-ELM (PEOS-ELM) algorithm based on MapReduce for extending the data scale of learning.

### 3.2. The Extended OS-ELMs

Attracting by the advantages of OS-ELM, such as, low computational burden, strong nonlinear-fitting ability, without local minimum and easy to implement, many researchers from various applications has introduced and modified OS-ELM for realizing the tasks of their applications.

OS-ELM resolves the concept drift problem only via updating models from new acquired data, and cannot do well on a time-sensitive data flow which is common in in plenty of practical applications, such as weather forecast, stock forecast, and so on [21]. Time sensitive means that data often have timeliness and each chunk of data has a period of validity. To handle this issue, Zhao et al. (2012) [21] introduce OS-ELM with Forgetting mechanism (FOS-ELM) and its ensemble version via discarding old and validate data. As an alternative way, Ye and Dai (2018) [22] transformed invalidate data in validate ones, and propose OS-ELM with Kernels abbreviated as TrEnOS-ELMK, and its ensemble version for time series forecasting. Considering the differences of valid data, Zou et al. (2018) [23] propose Memory Degradation Based OS-ELM (MDO-ELM) which adjusts the weights of valid data by a self-adaptive memory factor, and at the same time discarding invalid data.

Mirza et al. (2013) [24] propose Weighted OS-ELM (WOS-ELM) for class imbalance and concept drift learning. WOS-ELM assigns higher weight to minority-class samples than to majority-class samples. Instead of adopting fixed weights, a weight selection strategy is used. Ding et al. (2018) [25] presented the WOS-ELM with kernels (WOS-ELMK) for online imbalance class learning via kernel mapping, avoiding the non-optimal hidden node problem in WOS-ELM. It can tackle both the binary class and multiclass imbalance problems.

### 4. Conclusions

The OS-ELM family has important potential for big data applications and academic research. From the application perspective, OS-ELM has a potential to be used as a learning tool of data flows with concept drift problems to improve generalization, robustness, stability, efficiency and output quality in applications. From the academy/research perspective, OS-ELM can be stated as a multidisciplinary topic of research that encompasses several areas of study, such as machine learning and artificial intelligence. Big data era bring much unprecedented opportunities and challenges to the OS-ELM family. It is expected that the short review of this paper will support a certain research in the future.

### 5. Acknowledgements

We would like to thank anonymous reviewers for their careful reading of the manuscript.
6. Funds
This work is supported by the National Natural Science Foundation of China (No. 61702070) and the Research Projects of Liaoning Marine Fisheries Office (No. 201512).

7. References
[1] Laney D (2001). 3D data management: Controlling data volume, velocity and variety, META Group, Tech. Rep.
[2] Chi M, Plaza , Benediktsson J A, Sun Z, Shen J, Zhu Y (2016). Proceedings of the IEEE, 104(11):2207-2219.
[3] Zhou Y, Chang F-J, Chang L-C, Kao I-F, Wang Y-S, Kang C-C (2019). Science of the Total Environment, 651:230-24.
[4] Wang X-J, Liu Y, Yuan P, Zhou C-J, Zhang L (2019). IEEE Access, 7:7368-7380.
[5] Ulfarsson M O, Palsson F, Sigurdsson J, Sveinsson J R (2016). Proceedings of the IEEE, 104(11):2137-2154.
[6] Wu X, Zhu X, Wu G-Q, Ding W (2017). IEEE Transactions on Knowledge and Data Engineering, 26(1):97-107.
[7] Zhou L, Pan S, Wang J, Vasilakos A. V (2017). Neurocomputing, 237:350-361.
[8] Zhou Z-H (2018). National Science Review, 5(1):54-58.
[9] Zhou Z-H (2016). Frontiers of Computer Science, 10(4): 589-590.
[10] Zliobaite I. Learning under concept drift: an overview, CoRR abs/1010.4784.
[11] Souza F A A, Araújo R, Mendes J (2016). Chemometrics and Intelligent Laboratory Systems, 152:69-79.
[12] Liang N-Y, Huang G-B, Saratchandran P, Sundararajan N (2006). IEEE Transactions on Neural Networks, 17(6):1411-1423.
[13] Huang G B, Chen L., Siew C K (2006). IEEE Transactions on Neural Networks, 17(4):879-892.
[14] Huang G-B, Zhou H, Ding X, Zhang R (2012). IEEE Transactions on Systems, Man and Cybernetics: System, 42(2):513-529.
[15] Liu Y, He B, Dong D, Shen Y, Yan T, Nian R, Lendasse A (2014). Proceedings of ELM-2014. 1:325-344.
[16] Huang S, Wang B, Qiu J, Yao J, Wang G, Yu G (2016). Neurocomputing, 174(A):352-367.
[17] Ding S, Mirza B, Lin Z, Cao J, Sepulveda J (2018). Neurocomputing, 277:139-148.
[18] Huynh H T, Won Y (2011). Pattern Recognition Letters, 32:1930-1935.
[19] Gu Y, Liu J, Chen Y, Jiang X (2017). 2014 International Joint Conference on Neural Networks (IJCNN), pp.732-738, July 6-11, 2014, Beijing, China.
[20] Lan Y, Soh Y C, Huang G-B (2009). Neurocomputing, 72:3391-3395.
[21] Zhao J, Wang Z, Park D S (2012). Neurocomputing, 87:79-89.
[22] Ye R, Dai Q (2018). Knowledge-Based Systems, 56:74-99.
[23] Zou Q-Y, Wang X-J, Zhou C-J, Zhang Q (2018). Neurocomputing, 275:2864-2879.
[24] Mirza B, Lin Z, Toh K-A (2013). Neural Process Lett, 38:465-486.
[25] Ding S, M Bilal, Lin Z, Cao J, Lai X, Nguyen T V, Sepulveda J (2018). Neurocomputing, 277:139-148.