UNIVERSITY OF TECHNOLOGY SYDNEY
Faculty of Engineering and Information Technology

STREAMING DATA REGRESSION

by

Hang Yu

A Thesis Submitted
in Fulfillment of the
Requirements for the Degree

Doctor of Philosophy

Sydney, Australia

November, 2020

Certificate of Authorship/Originality

I, Hang Yu declare that this thesis, is submitted in fulfillment of the requirements for the award of Doctor of Philosophy, in the Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Production Note:

Signature: Signature removed prior to publication.

Date: 10/11/2020

ABSTRACT

STREAMING DATA REGRESSION

Machine learning is a field of computer science that gives computers the ability to learn knowledge. Regression analysis is one of the most important tasks to address in the area of machine learning, and it is a form of predictive modeling technique that investigates the relationship between dependent and independent variables. However, most regression algorithms, whether against linear regression or nonlinear regression analysis, were designed based on batch datasets. Nowadays, technological advancements make it possible to access fast and potentially infinite data known as streaming data. In streaming data, the data is displayed in the form of sequences and can only be read once in a predetermined order, so batched regression algorithms cannot be used to process streaming data. The streaming algorithm is a new type of technique in machine learning. In streaming algorithms, data are processed sequentially as well and can be examined in only a few passes (typically just one).

However, as a novel learning technique, the streaming algorithm is still immature and imperfect for the regression problem. Firstly, most of the existing streaming regression algorithms only can address precise data; however, in many real-world applications, streaming data is generated under noisy environments. The noisy data impacts the learning process of many regression algorithms and thereby resulting in the performance of many algorithms decrease dramatically. Secondly, more studies on streaming data show that data distribution is nonstationary; it can change or evolve. Concept drift refers to this unpredictable change of data distribution in streaming data, and the performance of an algorithm becomes declines when concept drift occurs. Hence, concept drift in streaming data is also a factor that impacts the performance of streaming regression algorithms. Finally, in many real-world applications, the regression problem of streaming data becomes more complicated.

Two or more outputs instead of single output need to be predicted. However, multi-output regression, which corresponds to two or more outputs, has been discussed extensively for offline, static settings. Only a few works address how to solve this problem for streaming data. Motivated by this reasoning, our research on streaming data regression aims to conquer the aforementioned challenges.

In order to solve streaming data regression under a noisy environment, we propose a novel online regression algorithm, called online robust support vector regression (ORSVR). ORSVR is able to solve nonparallel bound functions simultaneously. Hence, the large quadratic programming problem (QPP) in classical v-SVR is decomposed into two smaller QPPs. An online learning algorithm then solves each QPP step-by-step. The results of a series of comparative experiments demonstrate that the ORSVR algorithm efficiently solves regression problems in streaming data, with or without noise, and speeds up the learning process. Furthermore, we also propose an online topology learning algorithm to filter noise data in the data preprocessing stage, called Gaussian membership-based self-organizing incremental neural network (Gm-SOINN). Gm-SOINN is an unsupervised learning algorithm and can learn a topology network to represent the data distribution accurately. The size of the topology network is much smaller than the size of the training data. In addition, Gm-SOINN utilizes the advantages of fuzzy logic, unlike other SOINN-based methods that allow only one node to be identified as a "winner" (the nearest node), Gm-SOINN allows for any node to be selected as the winner and uses a Gaussian membership to indicate the degree to which nodes are identified as winners.

In order to the streaming data regression problem under evolving environments, we propose continuous support vector regression (C-SVR) for nonstationary streaming data. Like an ensemble-based method, in C-SVR, a series of regression models are continuously learned in a series of time windows to determine the relationship between the input and output at different timestamps. Additionally, in contrast to algorithms that forget all learned knowledge, learning processes in different time windows are not independent in C-SVR. A similarity term added to the QPP carries some learned knowledge from the last model forward into the current model. The problem of evolving streaming data regression has been a topic of consistent research in the fuzzy systems community. Hence, a novel evolving-fuzzy-neuro system, called the topology learning-based fuzzy random neural network (TLFRNN), is proposed. In TLFRNN, we revised our proposed Gm-SOINN to self-organize each layer of TLFRNN. However, different from current EFN systems, TLFRNN learns multiple fuzzy sets to reduce the impact of noises on each fuzzy set, and a randomness layer is designed, which assigning the probability of each fuzzy set. Also, TLFRNN does not utilize TSK rules; instead uses a simple inference that considering fuzzy and random information of data simultaneously. More importantly, in TLFRNN, concept drift can be detected and adapted easily and rapidly.

In order to solve the multiple-output regression problem of streaming data, we present an online multi-output regression system, called MORStreaming, for streaming data. MORStreaming uses an instance-based model to make a prediction because this model can quickly adapt to change by only storing new instances or by throwing away old instances. However, learning instances in our regression system is constrained by online demand, and need to consider the relationship between outputs. Hence, MORStreaming consists of two main algorithms: 1) an online learning instances algorithm based on topology networks was designed to make MORStreaming robust to noise and determines the number of instances. 2) an online learning structured-outputs algorithm based on adaptive rules was designed for MORStreaming to learn the correlation between outputs automatically.

In summary, our thesis describes original research into streaming data regression, a problem that is important but relatively under explored. The original contribution is made in 3 aspects: (i) dealing with noisy streaming data; (ii) dealing with evolving streaming data; (iii) dealing with streaming data with multiple outputs.

Dissertation directed by Professor Jie Lu
Australian Artificial Intelligence Institute

Dedication

To my parents Tao Yu and Li Guo, and my girlfriend, Yiqun Jiang

Acknowledgements

It has been an exciting and memorable journey at the University of Technology Sydney (UTS) in pursuit of a Ph.D. degree over the past four years. I am sincerely grateful to the people who inspired and helped me in many ways.

I would like to express my earnest thanks to my principal supervisor, Distinguished Professor Jie Lu. Without her help, I would not have been able to obtain this Ph.D. degree. She led me into a new academic research field and guided me to pursue my own research interests. In addition, her professional knowledge enlightened me and inspired me to delve further and deeper into my research, especially when I become lost in my research direction. I felt extremely honored to be guided by such a distinguished researcher as well as an enthusiastic mentor. The skills and knowledge she imparted over the past four years has benefited my Ph.D. study and will be a great treasure throughout my life. Meanwhile, I would like to express my foremost and deepest gratitude to my co-supervisor, A./Professor Guangquan Zhang. He taught me how to think and work as a professional researcher. He also unconditionally supported me in pursuing my own research interests from the beginning. Without his patience and encouragement, I would have wasted my time on trivial research ideas. My discussions with him greatly improved the scientific aspect and quality of my research. His strict academic attitude and respectful personality benefited my PhD study and will be a great memory throughout my life.

During my Ph.D. candidature, I was fortunate to join the Decision Systems & e-Service Intelligence Lab (DeSI) in the Australian Artificial Intelligence Institute (AAII). I greatly enjoyed the pleasurable and plentiful research opportunities. It was a wonderful experience to spend four years with these dedicated researchers. I especially thank Dr Junyu Xuan and Dr Anjin Liu, who helped me greatly to deeply understand my research problem during my Ph.D. candidature; Dr Feng Liu, Dr Yiliao Song and other students in DeSI lab who have shared their opinions and comments with me. I also genuinely thank Michele Mooney, who proofread certain sections of this thesis and Robyn Barden for polishing the language of my thesis and publications. They were always patient in answering all my questions about academic writing. Hence, I have learned much about academic writing from them. I also thank all my wonderful friends, classmates, and colleagues for every enjoyable moment.

Last, I would like to express my heartfelt appreciation and gratitude to my family, friends, classmates, and colleagues for their love and support.

Hang Yu

Sydney, Australia, 2020.

List of Publications

Journal Papers

J-1. **H. Yu**, J. Lu, and G. Zhang, “Online Topology Learning by a Gaussian Membership-Based Self-Organizing Incremental Neural Network,” *IEEE Transactions on Neural Networks Learning Systems*, pp. 1-15, 2019, (DOI: 10.1109/ TNNLS.2019.2947658).

J-2. **H. Yu**, J. Lu, and G. Zhang, “An Online Robust Support Vector Regression for Data Streams,” *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–14, 2020, (DOI: 10.1109/TKDE.2020.2979967).

J-3. **H. Yu**, J. Lu, and G. Zhang, “Continuous Support Vector Regression for Nonstationary Streaming Data,” *IEEE Transactions on Cybernetics*, pp. 1–14, 2020, (DOI: 10.1109/TCYB.2020.3015266)

J-4. **H. Yu**, J. Lu, and G. Zhang, “Topology Learning-based Fuzzy Random Neural Network for Streaming Data Regression,” *IEEE Transactions on Fuzzy Systems*, pp. 1–14, 2020, (Under Review)

J-5. **H. Yu**, J. Lu, and G. Zhang, “MORStreaming: A Multi-Output Regression System for Streaming Data,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–14, 2020, (Under Review)

J-6. **H. Yu**, J. Lu, and G. Zhang, “Detecting Group Concept Drift From Multiple Data Streams,” *Pattern Recognition*, pp. 1–14, 2020, (Under Review)

Conference Papers

C-1. **H. Yu**, J. Lu, and G. Zhang, ”Learning a fuzzy decision tree from uncertain data,” *Proc. Int. Conf. on Intelligent Systems and Knowledge Engineering*, pp. 1-7, Nov. 24-26, 2017.

C-2. **H. Yu**, J. Lu, and G. Zhang, "An incremental dual nu-support vector regression algorithm," *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 522-533, Jun. 3-6, 2018.

C-3. **H. Yu**, J. Lu, G. Zhang, and D. Wu, "A dual neural network based on confidence intervals for fuzzy random regression problems," *Proc. IEEE Int. Conf. on Fuzzy Systems*, pp. 1-8, Jul. 8-13, 2018.

C-4. **H. Yu**, J. Lu, and G. Zhang, "A Hybrid Incremental Regression Neural Network for Uncertain Data Streams," *Proc. IEEE Int. Joint. Conf. on Neural Networks*, pp. 1-8, Jul. 14-19, 2019.

Note: Chapeter 3 relates J-1 and J-2, Chapter 4 relates J-3 and J-4, Chapter 5 relates J-5

Contents

List of Figures

List of Tables

Abbreviation

SVR - Support Vector Regression

ORSVR - Online Robust Support Vector Regression

AONSVR - Accurate Online Support Vector Regression

INVSVR - Incremental nu-Support Vector Regression

TSVR - Twin Support Vector Regression

QPP - Quadratic Programming Problem

KKT - Karush-Kuhn-Tucker

RKHS - Reproducing Kernel Hilbert Space

SOINN - Self-Organizing Incremental Neural Network

E-SOINN - Enhanced SOINN

LB-SOINN - Load-balancing SOINN Gm-SOINN - Gaussian membership-based SOINN

KDE-SOINN - Kernel Density Estimation SOINN

LD-SOINN - Local Distribution SOINN GNG - Growing Neural Gas

TN - Topology Network

ART - Adaptive Resonance Theory

MAP - Mean Accumulated Point

eVQ - evolving Vector Quantization

eGMM - Evolving Gaussian Mixture Model

C-SVR - Continuous Support Vector Regression EFS - Evolving Fuzzy System

TSK - Takagi-Sugeno-Kang

eFN - Evolving-Fuzzy-Neuro

FCM - Fuzzy c-means

CB - Case-base

MVR - Mean-Variance-Ratio

ARMSE - Average Root Means Square Error

FLEXFIS - Flexible Fuzzy Inference System

DENFIS - Dynamic Evolving Neural-Fuzzy Inference System

RLS - Recursive Least Squares

FIMT-DD - Fast Incremental Model Trees Drift Detection

BSGD - Budgeted Stochastic Gradient Descent

AMRules - Adaptive Model Rules

IBLStreams- Instance-based Learning System for Streaming Data

ORTO - Online Option Trees For Regression

ARF-Reg - Adaptive Random Forest

Learn++.NSE- Adaptive batch-based ensembles

AddExp.C - Adaptive datum-based ensembles

MOR - Multiple Output Regression

Nomenclature and Notation

ε	is the ε-insensitive loss function and defined as $\|Y - f(X)\|\varepsilon = \max\{0, \|Y - f(X)\| - \varepsilon\}$ for a predicted value $f(X)$ and a true output Y, which does not penalize errors below some $\varepsilon > 0$, chose a prior. Thus, the region of all samples with $\{\|Y - f(X)\| \leq \varepsilon\}$ is called ε-tube.
K	is the kernel function
$\alpha_i - \alpha_i^*$	is the weight of $K(X_i, X_j)$
$*_{1i}$	is the ith * variable in the upper function
$*_{2i}$	is the ith * variable in the lower function
Δ	is the amount of the change of each variable
Q'_{ij}	is the sub-matrix of Q_{ij} after initial adjustment
$*'_{1i}$	is the ith * variable in the upper function after initial adjustment
$*'_{2i}$	is the ith * variable in the lower function after initial adjustment
$Q'_{s_s S_S}$	is the sub-matrix of Q' with the rows and columns indexed by S_S
$Q_{n_n S_S}$	is the sub-matrix of Q' with the rows and columns indexed by n_n
λ	is the number of samples during one learning period
$age_{\max}$	is the lifetime of each edge
AN	is set of all nodes
$\|\|\|\|$	is the Euclidian distance (L_2-norm)
W_i	is the n-dimensional weights vector of node i
nei_i	is the neighbour nodes of node i, i.e., the nodes that directly connect to node i

L_i	is the learning time of node i
ε	is the learning rate and usually be set to 100
p_i	is the point of node i
sp_i	is the sum of points of node i during a learning period
h_i	is the mean accumulated point (MAP)
Num_A	is the number of nodes in subclass A
$mean_A$	is the mean density of the nodes in subclass A
t	is the order in which the sample X inputted
$\mu_i(t)$	is the Gaussian membership of input sample $X(t)$ belongs to node i
G	is a Gaussian model
mv_G	is the mean vector of the G
Cov_G	is the covariance matrix of the G
num_G	is the winner times of the G
r_G	is a vigilance parameter to decide whether an input data belongs to the G