IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 14, 2022, accepted April 17, 2022, date of publication April 22, 2022, date of current version May 3, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3169785

Learning Under Concept Drift for Regression—A Systematic Literature Review

MARÍLIA LIMA[1], MANOEL NETO[1], TELMO SILVA FILHO[2], AND ROBERTA A. DE A. FAGUNDES[1], (Member, IEEE)

[1]Department of Computer Engineering, University of Pernambuco, Recife, Pernambuco 50720-001, Brazil
[2]Department of Statistics, Federal University of Paraíba, João Pessoa, Paraíba 58051-900, Brazil

Corresponding authors: Marília Lima (e-mail: mncal@ecomp.poli.br), Manoel Neto (e-mail: maan@ecomp.poli.br), Telmo Silva Filho (e-mail: telmo@de.ufpb.br), and Roberta A. de A. Fagundes (e-mail: roberta.fagundes@upe.br)

This work was supported in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Brazil, under Grant 001.

ABSTRACT Context: The amount and diversity of data have increased drastically in recent years. However, in certain situations, the data to which a trained Machine Learning model is significantly different from testing data, a problem known as Concept Drift (CD). Because CD can be a serious issue, there has been a wealth of research on how to detect and work around it. However, most of the literature focuses on classification tasks. Objective: Making a Systematic Literature Review (SLR) for CD in the context of regression. Research questions: How to detect CD and how to build CD techniques for regression problems using machine learning? Method: We ran an automatic search process on reference databases, selecting papers from 2010 to August 2020, following the methodological process proposed by (*Kitchenhame* and *Charters*) (2007). Results:We selected 41 papers. Drift Detection Methods based on ensembles and neural networks with highlight OS-ELM were the most frequent in the selected papers with superior performance. However, only two papers confirm such superiority statistically. Furthermore, identify CD problems as the batch size, drift points, and where drift happens. Conclusions: SLR focuses on highlighting the existing literature on CD applied to regression.

INDEX TERMS Concept drift, data stream, ensemble, regression, non-stationary environments, systematic literature review.

I. INTRODUCTION

In the real world, contexts frequently change, causing a dynamic behavior in the data generated everyday. In the same way, the amount of data generated per day by big companies and the government is getting bigger. This makes machine learning a very useful tool to analyze these data effectively, helping with data-driven decision making [1]. For example, in the context of the Internet of Things (IoT), the authors Zhang *et al.* (2021) [2] consider the dynamics of traffic data as a scenario of temporal evolution. They propose an ensemble learning in the scenario of connected Internet of vehicles. The numerical results indicate that the proposed model improves prediction accuracy by 30% to 40% compared to several basic methods. Whereas, the authors Guo and Wang (2020) [3], considering the overload of information currently present in the IoT context, they propose the use of a deep graph neural network-based social recommendation framework exploring the correlation between data characteristics. The results for this work show that the proposed model is able to perform better than the conventional model in most experiments. However, when data change dynamically, traditional machine learning techniques, which use static data to fit models, may not adapt to new concepts and environments. Such scenarios give rise to Concept Drift (CD), whose study aims to propose and use techniques for solving complex dynamically evolving problems.

The idea behind CD is that new instances arriving after deployment of a model do not follow the same distribution as the training dataset, and the characteristics of the output ($\hat{y}$) change periodically making the problem hard to model [4], [5]. As a result, models lose generalization performance with time. Thus, the CD problem occurs when the data at period t_0 does not have the same distribution at time t_{+1}. Therefore, the distribution of the data is different at $t_0 \neq t_{+1}$ that causes the degradation of the machine learning model.

The associate editor coordinating the review of this manuscript and approving it for publication was Vivek Kumar Sehgal.

It happens since the information that was used to train the model does not have the same as the data used to test it [4]. A mathematical definition of CD might look like this: considering time t_0 and t_1, $P_{t_0}(X, y) \neq P_{t_1}(X, y)$, where X is the independent variable, y the independent variable, and P the joint probability at that point. This kind of problem is well known in fraud detection [6], healthcare [7], financial distress [8], fake news detection [9] and, electric power consumption [10]

CD has garnered attention from various authors who have proposed techniques and frameworks to solve this problem. For example, Widmer and Kubat (1996) [11] proposed a Floating Rough Approximation (FLORA) framework, which uses a strategy that stores sets of concept descriptors, using dynamic size sliding windows to select the most suitable descriptor for each context. Minku and Yao (2011) [12] proposed Diversity for Dealing with Drifts (DDD), an approach based on online ensemble learning with different levels of diversity. Brzezinski and Tefanowski (2013) introduced Accuracy Updated Ensemble (AUE2) which is a data stream classifier with the objective of reacting equally to different types of drift, i.e. it combines precision-based weighting mechanisms based on batches with the incremental nature of Hoeffding Trees [13]. Gonçalvez *et al.* (2014) [14] presented a comparative study of CD detectors using the Naive Bayes method as a base learner, and concluded that the best detector depends on the type of drift. Guo et al (2021) [15] used an adaptive online deep neural network technique based on ensembles to adapt models to CD problems. In their work, the method groups surface features with deep features and dynamically fits the data stream in the network considering possible drifts. Finally, Abbasi *et al.* (2021) [16] proposed a new approach called ElStream using majority voting. ElStream detects concept drift using ensembles and conventional machine learning techniques, in order to provide consistent performance. Many works have applied ensemble techniques, as according to Lu *et al.* (2018) [17] ensemble learner approaches played an increasingly important role in the context of CD. However, a large part of the literature focuses on classification problems, whereas for regression, where the target variable is continuous, it is still fairly unexplored.

Thus, due to the importance of the CD field, this work aims to carry out a systematic review of the CD literature regarding regression tasks. For that, we elaborated the following general research question:

A. "HOW TO DETECT CONCEPT DRIFT AND PERFORM ADAPTIVE LEARNING FOR REGRESSION USING COMPUTATIONAL INTELLIGENCE METHODS? IN PARTICULAR, WHICH ENSEMBLE REGRESSION APPROACHES GUARANTEE GOOD PERFORMANCE IN THIS SCENARIO?"

The selection of studies in this work follows an automatic search process carried out according to the following steps: (*i*) search in reference databases (ACM, IEEE, SCOPUS and Web of Science); (*ii*) evaluation of titles, abstracts and keywords according to the inclusion criteria defined in Section III, in order to decide which studies are approved for the next phase; (*iii*) evaluation of the introductions and conclusions of the selected studies according to inclusion criteria; (*iv*) full reading of remaining papers and quality analysis, as defined in Section III; (*v*) extracting data related to research questions. Upon completion of this process, 41 studies were selected.

The main contributions of this work are delivered by studying different detector methods and machine learning models to deal with CD for regression. In addition to being able to identify real and synthetic data datasets, evaluation metrics and challenges in the CD field. Therefore, our work covers the following topics:

- CD detectors for regression;
- adaptable models for dynamic regression environments;
- and insights on how to deal with CD for regression.

The rest of this paper is structured as follows. Section II provides and discusses CD. Section III presents our literature search methodology. Section IV presents the results to the research questions, making a comprehensive analysis of the main discoveries. Section V discusses some limitations of the study. And finally, Section VI presents an analysis of findings and future research.

II. BACKGROUND

This Section presents the historical context of CD, types of drifts, CD detector, and methodologies to train models in the presence of CD.

A. HISTORICAL CONTEXT

In 1986, authors Schlimmer and Granger (1986) [18] proposed the first method to deal with CD problems, called STANGER, which uses Bayesian statistics [19]. Then, in 1995, Kubat and Widmer presented the FRANN algorithm (Floating Rough Approximation in Neural Networks), which is based on Radial Basis Functions (RBF) and used a sliding window to determine the window size, in order to select data to be excluded by its heuristic [20]. One year later, in 1996, the same authors, Widmer and Kubat, proposed a new framework, called FLORA (FLOating Rough Approximation). This framework used algorithms that were flexible to CD. The idea for FLORA is to have a window of examples and hypotheses to store concept descriptions and use them to adapt the algorithm when an already known context comes up [11].

The year 2000 saw a new method based on Support Vector Machines (SVM), which identifies and handles CD problems through a window on a training dataset. The goal of this work is to automatically adapt the window size based on the training dataset, decreasing the inference error [21]. In 2001, Street and Kim presented the Streaming Ensemble Algorithm (SEA), an ensemble of decision trees combined with a CD heuristic [22].

In 2005, Scholz and Klinkenberg proposed the use of boosting techniques to train a classification ensemble algorithm through a pipeline that adapts to CD [23]. In 2007, Kolter and Maloof proposed the Dynamic Weighted Majority (DWM) method to work with CD problems. This algorithm dynamically creates and removes learners depending on their performance. The idea is to train the learners in an online ensemble, then assign weights to the learners based on their performance and add new learners to the ensemble when needed [19]. In the same year, Bifet and Gavalda (2007) proposed the ADWIN2 (ADaptive Algoritmo WINdowing), an improved version of the ADWIN algorithm. ADWIN2 proposes a dynamic window that can increase or decrease depending on CD detection. The same work also presented the combination of ADWIN2 with Naive Bayes (NB) applied to CD tasks [24].

Finally, in 2009, Minku el al. (2009) [25] proposed a new categorization for CD. In addition, the authors studied ensemble techniques for CD tasks and verified that, before the drift happens, an ensemble that contains less diversity has less inference errors. However, keeping diversity in the ensemble is a good strategy to obtain less inference errors right after the drift,regardless of the type of drift. After the drift, high diversity was not important.

B. CONCEPT DRIFT

CD is a problem that occurs when the statistical properties of the data change randomly through time [4]. Therefore, CD happens in the learning context of the data, changing the independent and/or dependent variables. Formally, Concept Drift can be defined in any scenario in which the probabilities of the variables vary through time, such that $P_{t+1}(X, y) \neq P_t(X, y)$, i.e. the distribution at time t is different than at time $t + 1$ [26]–[28]. Given this definition, we now address the types of CD and methods to detect it.

1) TYPES OF CONCEPT DRIFT

According to Gama *et al.* (2014) [27], CD can be classified into four different types regarding the speed of drift, as shown in Fig. 1. The types of drifts are: **sudden**, which happens when there is a sudden change from one concept to another; **gradual**, when one concept changes into another little by little, e.g. given an initial concept X and a new one Y, gradually more occurrences of Y are observed and fewer occurrences of X; **incremental:**, where there are several intermediate concepts, gradually moving from one concept to another; and **recurring**, which occurs when a previously active concept reappears after some time. The recurring type should not be confused with seasonality, as it is not periodic and it is not clear when a previous concept can reappear.

According to Liu *et al.* (2017) [29], Concept Drift does not happen at an exact moment. Actually, it can happen for a long period of time, resulting in intermediate concepts until the data finally move from one concept to another. In such cases, intermediate concepts can often represent a gradual drift. Gata *et al.* (2014) discusses the importance of differentiating between CD and outliers or noise, which refer to random deviations, not to a large drift in the data concept.

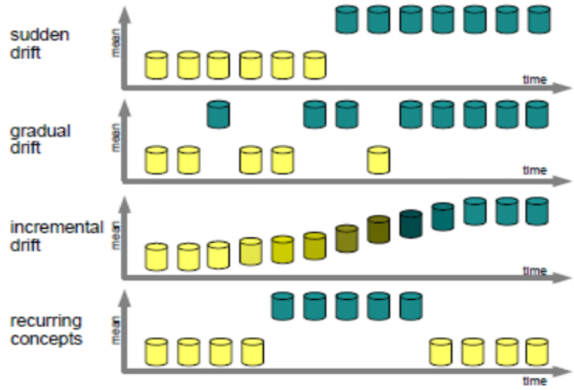


FIGURE 1. Types of drifts by speed of occurrence.

2) CONCEPT DRIFT DETECTION

CD detection concerns techniques to detect drift in data points or intervals as data are presented to a model during deployment. Several drift detection techniques have been proposed in the literature. Most detectors generate a prediction from the model to later compare its answer with the correct data label, in order to find out if the prediction was wrong or right. Some drift detection algorithms include:

- **Drift Detection Method (DDM):** considered one of the most referenced CD detection methods in the literature [17], [30]. DDM uses a reference time window. When a new data instance arrives, the method calculates the error and if it detects an increase in the error rate that exceeds a calculated threshold, either a drift is detected or the algorithm warns of a future drift, which is called the [30] alert zone.
- **Early Drift Detection Method (EDDM):** works similarly to DDM. The difference is that EDDM uses the distance between the models' errors, that is, it works by monitoring the average distance between two errors instead of just the error rate. Thus, it is a more suitable method for detecting gradual drifts [31];
- **ADaptive WINdowing (ADWIN):** uses an adaptive sliding window to detect changes. The goal is to keep the statistics of a variable-size window while detecting the CD. The window size is defined by dividing the statistics window at different points, so that the average of some of the windows can be checked. If the absolute value of the difference is greater than a previously defined limit, it indicates that a deviation occurred [24];
- **EWMA for Concept Drift Detection (ECDD):** an extension of the Exponentially Weighted Moving Average (EWMA) proposed to detect an increase in the average of a sequence of variables. In this scenario, ECDD monitors the model error rate [32];
- **Paired Learners (PL):** based on time windows, uses two models: one stable and one reactive. The stable

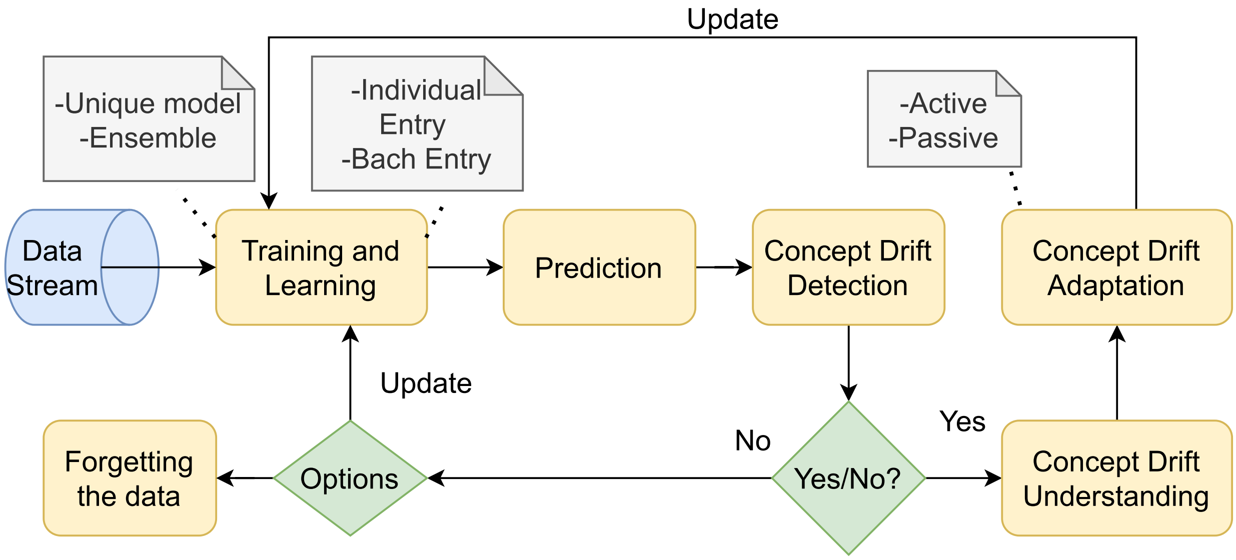


FIGURE 2. Generic approach to learning with CD. Adapted [17].

model is based on long data history, whereas the reactive one works in a small recent data window. The reactive model is used to identify CD, while the stable one is used to make predictions [33];

- **Kolmogorov-Smirnov Windowing (KSWin):** based on the Kolmogorov-Smirnov (KS) statistical test. It uses a sliding window of fixed size, with the last samples considered used to represent the last concept of the data (R). When a new windows of data arrives, KSWin uniformly draws a sample to represent the new candidate concept W. Then, KSWin runs a KS test on the two samples R, W) to check whether there is drift or not [34];
- **Just-In-Time adaptive classifiers (JIT):** one of the first methods that defines several CD detection hypotheses. The basis of this method is to extend Computational Intelligence-based CUSUM test (CI-CUSUM) to detect the mean change in data characteristics [35];
- **Information-Theoretic Approach (ITA):** a density-based method whose objective is to partition the (multidimensional) data into a set of boxes, using the Kullback-Leibler divergence to quantify the difference between the [36] boxes.

Other methods similar to DDM include Hoeffding's inequality based Drift Detection Method (HDDM) [37], Fuzzy Windowing Drift Detection Method (FW-DDM) [38] and Local Drift Detection (LLDD) [39]. In most of the drift detection methods presented here, as well as in most of the methods found in the literature, first the CD happens, which leads to model error, which is then followed by a reaction to the deviation.

C. LEARNING UNDER CONCEPT DRIFT

Traditional machine learning has two fundamental stages: training and testing/inference. However, when the data changes dynamically, characterizing CD problems, the process is divided into three phases: CD detection; drift understanding (where and how does it occur in the data?) and CD adaptation (how to react to drift?) [17]. Therefore, Fig. 2 presents a generic structure for working in dynamic environments with CD.

When analyzing Fig. 2, we see that training and learning can be applied in different ways for CD tasks depending on the application. For this work we used the categorization for training and learning and CD proposed by Kuncheva (2008) [40] and Elwell and Polikar (2011) [26]. Here we might learn from one sample at a time (individual input), which can be more sensitive to noise, or from batches of samples, which can be more stable, but performance depends on the chosen batch size. Additionally, we can use a single model or an ensemble of models in the training and learning step. The resulting models are then used for prediction.

In **Concept Drift Detection**, it can be used the detectors already presented in this section, followed by the approaches.

For **Concept Drift Understanding:** how to identify with precision the moment in which happens a drift it is very important for adaptation drift learning, because a delay or a false alarm will lead to failure to identify new deviations in the data, consequently, different concepts.

In **Concept Drift Adaptation:** where the training for new models for drift considering the whole data; building new specialist models for each kind of drift types; updating the currently models; and, retraining new models. **active:**, when there is a Drift detection method, being able to update the model only when the drift is detected; or **passive:** that considers existence to drift and periodically updates the model. If a change actually happens, it is learned, otherwise the knowledge is reinforced.

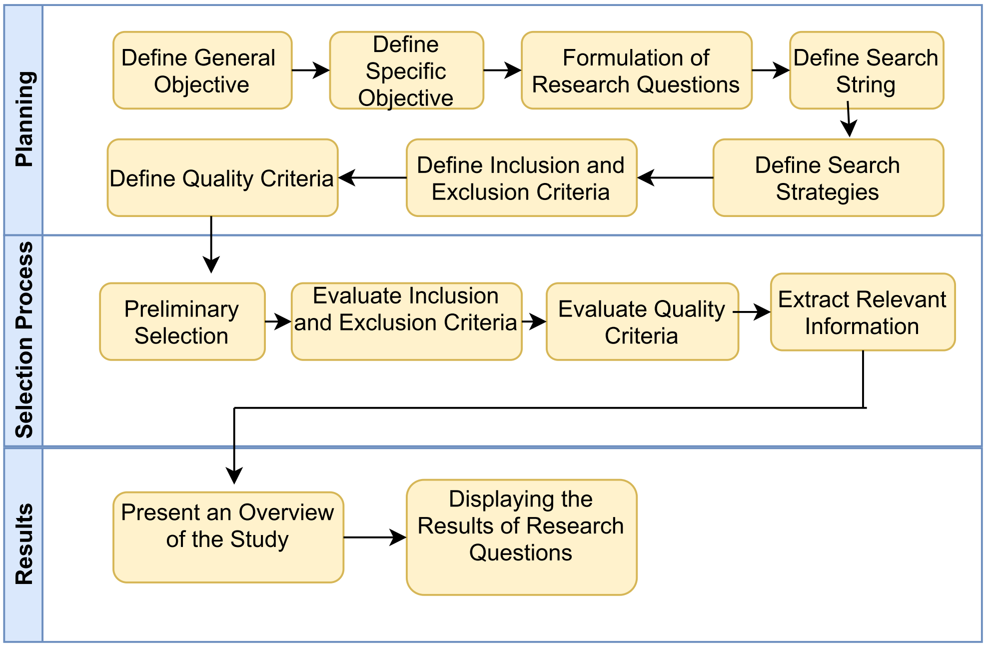


FIGURE 3. SLR process.

Finally, in **Forgetting the data**, the dataset not used in **Concept Drift Adaptation** can be excluded or storage for future use given a specific period of time.

III. RESEARCH METHODOLOGY

We built an empirical process in order to conduct this systematic literature review (SLR). The methodology used here is based on the study by Keele *et al.* (2007) [41], presented in Fig. 3. Our proposed SLR is composed of three basic steps: planning, selection process and results (SLR report).

In the **planing** section, we define objectives, research questions, inclusion and exclusion criteria and quality assessment. At this stage, the SLR protocol is drawn up, which will serve as a guide when conducting the research. In the **selection process**, we analyze selected papers, applying inclusion and exclusion criteria, quality assessment and extraction of relevant information for the SLR. Finally, the **Result** step provides an overview of the results of the studies and an analysis of the research questions. These results are presented in Section IV

The following Sections present the definitions of the processes as shown in Fig. 3.

A. OBJECTIVES

For this SLR work, the general objective is to identify detectors and machine learning techniques that deal with CD in the context of regression. Thus, we elaborated the following specific objectives:

- identifying CD detectors in the context of regression;
- identifying machine learning methods that deal with Concept Drift for regression;
- identifying datasets used for model evaluation under CD in the context of regression;
- and identifying current challenges in the field of CD for regression.

B. RESEARCH QUESTIONS

In this SLR, we analyzed adaptive learning approaches for CD, drift detectors and challenges in this area of research. Therefore, focusing on answering the question defined in Section I, we elaborated specific research questions, presented in Table 1, which we aim to answer in this work.

TABLE 1. Research questions.

ID	Question
RQ1	What methods are used to detect Concept Drift in non-stationary environments?
RQ2	Which machine learning method is used to adapt Concept Drift in non-stationary environments?
RQ3	What problems/challenges are encountered in detecting and adapatation Concept Drift in non-stationary regression environments
RQ4	Which datasets are used?
RQ5	What problems/challenges are present in real/simulated datasets, when using regression models for solving Concept Drift?
RQ6	Which assessment metrics are used?

C. SEARCH STRING

We specified the search query considering the main terms for CD for regression with ensemble approaches. We performed a sensitivity analysis of string terms (pilot study) to refine the search string and we excluded keywords whose inclusion made it so that no papers were returned. After the initial analysis, we defined the following search string used for the automatic search considering keywords, title and abstract:

1) ("CONCEPT DRIFT" OR "CONCEPT SHIFT" OR "COVARIATE SHIFT" OR "DATASET SHIFT" OR "NON-STATIONARY" OR "PRIOR PROBABILITY SHIFT") AND ("ENSEMBLE" OR "COMBIN*" OR "COMITTEE" OR "FUSION" OR "MULTIPL*") AND ("REGRESSION" OR "FORECAST" OR "PREDICT" OR "PREDICTING" OR "PREDICTION")

Some of the synonyms for "ensemble" were taken from Idri *et al.* (2016) [42], while synonyms for "concept drift" were taken from Lu *et al.*(2018) [17] and Almeida *et al.* (2019) [43]. Finally, we adapted the string to each Digital Library, in order to consider its restrictions and syntax.

D. SEARCH STRATEGIES

We used an automatic search, following the PICOC (Population, Intervention, Comparison, Outcome, Context) strategy [41], [44].

- **Population:** peer-reviewed publications reporting approaches related to the theme of concept drift in regression;
- **Intervention:** collecting evidence regarding approaches proposed for concept drift in non-stationary regression environments;
- **Comparison:** does not apply here;
- **Outcomes:** approaches for detection and adaptive learning in concept drift in non-stationary environments using regression and ensemble techniques;
- **Context:** primary research dealing with concept drift in regression problems.

The automatic search was performed in the following Digital Libraries, selecting papers in the period between 2010 and 2020:

- ACM Digital Library (www.dl.acm.org);
- IEEE Xplore (www.ieee.org);
- SCOPUS (www.scopus.com) and;
- Web of Science (www.webofscience.com).

E. INCLUSION AND EXCLUSION CRITERIA

The inclusion criteria are presented in Table 2, while exclusion criteria are shown in Table 3. We are only interested in primary studies which show some contribution on CD for regression and that were published between January 2010 and August 2020.

F. PAPER SELECTION PROCEDURE

The selection of papers for this SLR followed the methodology presented in Fig. 4, which is composed by five main steps. For the first step, the works were extracted using the automatic search with the search string, and the results are divided as follows. ACM returned 12 titles, IEEE Xplore returned 232, Scopus yielded 829 and Web of Science returned 502 search results. The resulting papers (1576) were downloaded and organized with the help of the Parsifal tool.[1] Out of the initial 1576 search results, 566 were excluded (step 2, Fig. 4) as they were duplicates. Then, after reading the title and abstract of the articles (Step 3), 655 studies were excluded, based on the inclusion criteria. Next, after reading the introduction and conclusion of the studies (Step 4), 285 papers were excluded. If there were not enough data to decide here, the study was kept for the next step. After the full reading of the articles and quality analysis, 29 studies were excluded, with 41 remaining studies. We excluded many studies from this SLR that primarily addressed CD for classification instead of regression.

TABLE 2. Inclusion criteria.

ID	Criterion
IC1	Primary studies that explicitly state Concept Drift detection/treatment
IC2	Studies that answer one or more of the questions defined in this systematic review
IC3	Studies in the area of Computational Intelligence that address ensemble regression models
IC4	The article is peer-reviewed (journal article, conference document)

TABLE 3. Exclusion criteria.

ID	Criterion
EC1	Studies developed in languages other than English
EC2	Articles not peer-reviewed
EC3	Studies that are not sufficiently available
EC4	Duplicated studies
EC5	Studies that are not related to the focus of this systematic review
EC6	Studies carried out before 2010
EC7	Editorial, poster, book, tutorial

G. QUALITY ASSESSMENT

The Qualitative Evaluation (QA) of the selected works was made by producing a score for each work. The works were evaluated according to a set of 10 quality criteria. The evaluation instrument used is shown in Table 4. Each quality assessment question is judged on three possible answers: "Yes" (score = 1), "Partially" (score = 0.5) or "No" (score = 0). Then, the quality index of a study is calculated by summing the scores from the answers to the QA questions. The Quality Scores of the selected studies are presented in Table 5.

H. DATA EXTRACTION AND SYNTHESIS

We prepared digital spreadsheets to record all the answers to the survey questions. We extracted the data described in Fig. 4 from each of the 41 primary studies included in this SLR. In addition to the selection process, conducting the SLR was aided by the Parsifal tool.

During the synthesis phase, we normalized terms that describe the same category by building a taxonomy using the terms that have the highest density of usage across all questions to get a breakdown of Concept Drift information for regression.

[1] https://parsif.al/

TABLE 4. Quality assessment criteria.

ID	Criterion
QA1	Are the research objectives clear?
QA2	Is there a clear description of the context in which the research was carried out?
QA3	Does the study clearly describe the methodological process used to detect or to adapt to Concept Drift?
QA4	Does the study make it clear what kind of Concept Drift is being detected/treated?
QA5	Does the study clearly describe the main difficulties for detecting or adapting to Concept Drift?
QA6	Does the study make it explicit which datasets were used?
QA7	Are the results described clearly?
QA8	Do the results have practical applications?
QA9	Does the study present which computational intelligence techniques used, along with any configuration parameters?
QA10	Does the study describe future research contributions and work?

IV. RESULTS AND ANALYSIS

This Section describes the results of our study; we discuss the answers of each research question separately.

A. OVERVIEW

The selected studies were published between 2010 and 2020. Fig. 5 presents the number of studies per year of publication, showing an increasing trend, with a maximum of 9 papers published in 2020. As one can see, there is a variation in the number of studies per year. In 2016 and 2020, those were the years that there are more articles dealing with CD in the regression context. There is no specific reason for this to happen, we believe that the advances in computational processing, making streaming data possible to be processed on a large scale and enabling the training of machine learning algorithms for CD taking the attention of the researchers, and then the number of publications increases.

Table 5 presents the 41 selected works with their respective quality scores. Some studies did not meet all quality criteria, but all studies that obtained a score of seven or higher were selected for the data extraction process related to the research questions.

TABLE 5. Quality scores for the 41 selected papers.

ID	Ref	year	QA1	QA2	QA3	QA4	QA5	QA6	QA7	QA8	QA9	QA10	Score
S1	[45]	2019	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.5	0.5	7.0
S2	[46]	2017	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.5	9.5
S3	[47]	2015	1.0	1.0	0.5	1.0	1.0	1.0	1.0	0.5	1.0	1.0	9.0
S4	[48]	2015	1.0	1.0	1.0	0.5	0.5	1.0	1.0	1.0	1.0	1.0	9.0
S5	[49]	2016	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.5	1.0	1.0	9.5
S6	[50]	2017	1.0	1.0	1.0	0.0	0.5	1.0	1.0	1.0	1.0	0.0	7.5
S7	[51]	2017	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	9.0
S8	[52]	2015	1.0	1.0	0.5	0.0	0.5	1.0	1.0	1.0	1.0	1.0	8.0
S9	[53]	2018	1.0	1.0	1.0	0.0	0.0	1.0	0.5	1.0	1.0	1.0	7.5
S10	[54]	2018	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.5	7.5
S11	[55]	2018	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.5	7.5
S12	[56]	2020	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.5	1.0	7.5
S13	[57]	2020	1.0	1.0	0.5	1.0	0.0	1.0	1.0	1.0	1.0	0.5	8.0
S14	[58]	2020	1.0	1.0	0.5	1.0	0.0	1.0	1.0	1.0	1.0	0.5	8.0
S15	[59]	2020	1.0	1.0	0.5	1.0	0.0	1.0	1.0	1.0	1.0	0.5	8.0
S16	[60]	2016	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	0.5	0.5	8.0
S17	[61]	2016	1.0	1.0	0.5	0.0	0.0	1.0	1.0	1.0	0.5	1.0	7.0
S18	[62]	2016	1.0	1.0	0.5	0.0	0.0	1.0	1.0	1.0	1.0	1.0	7.5
S19	[63]	2016	1.0	1.0	0.5	1.0	0.0	1.0	1.0	1.0	0.5	1.0	8.0
S20	[64]	2019	1.0	1.0	0.5	0.0	0.5	1.0	0.5	1.0	1.0	0.5	7.0
S21	[65]	2019	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	0.5	8.5
S22	[66]	2019	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.5	7.5
S23	[67]	2019	1.0	1.0	0.5	0.0	0.0	1.0	1.0	1.0	1.0	0.5	7.0
S24	[43]	2020	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.5	1.0	7.5
S25	[68]	2020	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	0.5	8.5
S26	[69]	2020	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.5	1.0	7.5
S27	[70]	2020	1.0	1.0	0.5	0.0	0.0	1.0	1.0	1.0	0.5	1.0	7.0
S28	[71]	2018	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	8.0
S29	[72]	2015	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	9.0
S30	[73]	2014	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	9.0
S31	[74]	2014	1.0	0.5	0.5	1.0	0.5	1.0	0.5	1.0	1.0	0.5	7.5
S32	[75]	2012	1.0	1.0	0.5	0.0	0.5	1.0	1.0	1.0	1.0	1.0	8.0
S33	[76]	2015	1.0	1.0	0.5	0.0	0.0	1.0	1.0	1.0	1.0	0.5	7.0
S34	[77]	2016	1.0	1.0	0.5	1.0	0.0	1.0	0.5	1.0	0.5	1.0	7.5
S35	[78]	2016	1.0	1.0	0.5	0.0	0.0	1.0	1.0	1.0	0.5	1.0	7.0
S36	[79]	2016	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	0.5	8.5
S37	[80]	2017	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	9.0
S38	[81]	2017	1.0	1.0	0.5	1.0	0.0	1.0	0.5	1.0	0.5	1.0	7.5
S39	[82]	2015	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	9.0
S40	[83]	2020	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	0.5	8.5
S41	[84]	2011	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	0.5	8.5

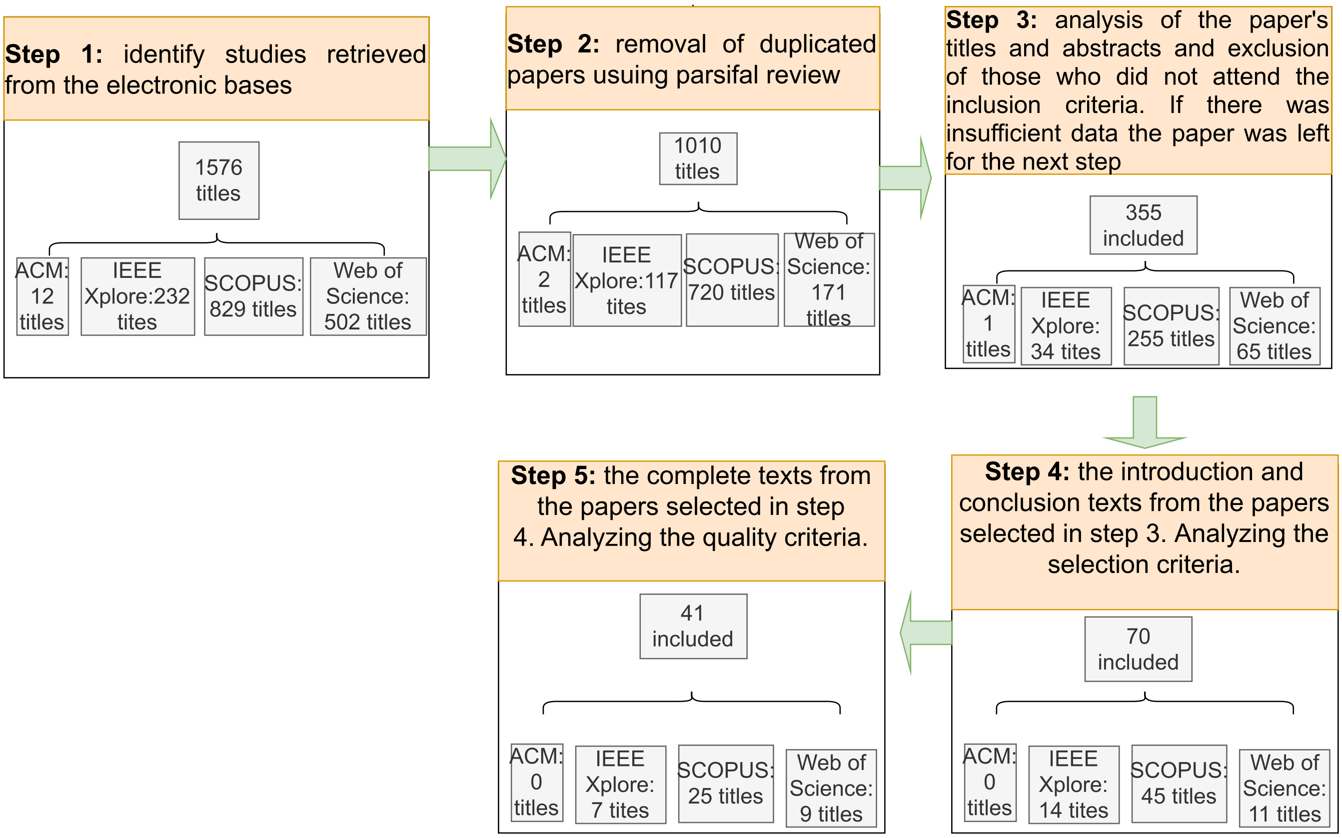


FIGURE 4. Number of articles selected in each step.

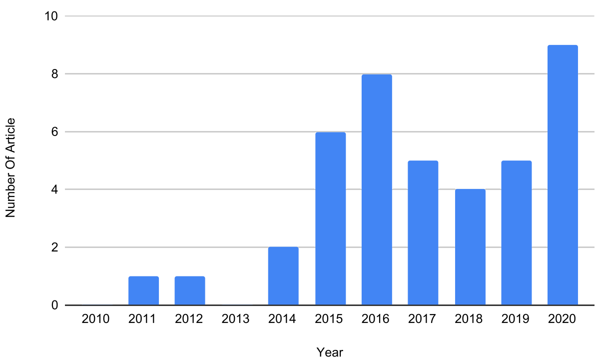


FIGURE 5. Number of articles per year (2010-2020).

Some of the selected works did not answer all the research questions. However, at least one of the research questions must be answered in order for a paper to be kept in the survey. Thus, subsequent sections present the answers to the research questions, as well as a qualitative analysis.

B. RQ1- WHAT METHODS ARE USED TO DETECT CONCEPT DRIFT IN NON-STATIONARY REGRESSION ENVIRONMENTS?

The idea behind this question is to identify and analyze approaches for CD detection in the context of regression. Only 12 (26.83%) works implemented any drift detectors for non-stationary environments.

As shown in Fig. 6, among all detection methods discussed in this work, DDM was the most discussed in the selected papers, appearing as the sole CD detection approach in three works ($S9$, $S22$, $S40$) and together with ECDD $S8$.

Seven studies ($S2$, $S4$, $S5$, $S14$, $S15$, $S19$ and $S26$) propose new CD detection methods, with $S14$ and $S15$ using the Just-In-Time-Learning (JITL) approach with the aim to assist in the detection process. This way, the models are only updated when the detector actually decides that a drift occurred. This can provide better accuracy for the trained model.

$S2$ proposes the use of lazy algorithms to detect if there was a change in the behavior of the data distribution throughout a neighborhood. This detection is made based on an estimated value through a data query/row. This methodology's main limitation is that it is necessary to select a good query/row to assess whether or not there was a change in the neighborhood. $S4$ introduces the use of an adaptive mechanism that analyzes the data in batches. $S5$ proposed a generic method that estimates the forecast error and uses Bayes's rule to identify whether the new value $t + 1$ represents CD. $S14$ introduced a detector that checks the difference between the prediction accuracy of Just-In-Time Learning (JITL) and JITLW models via EWMA and CUSUM charts. In $S15$, the authors proposed

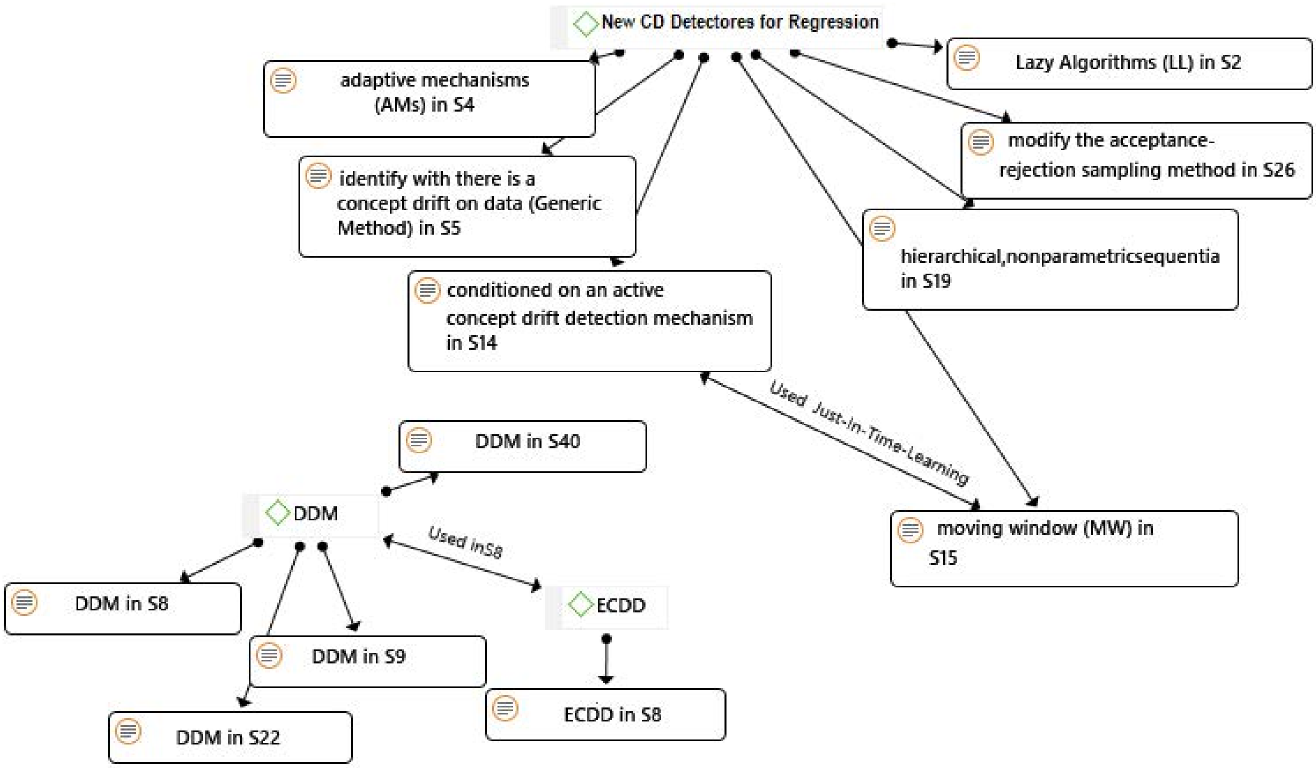


FIGURE 6. Drift detection methods network used in the studies.

a moving window (MW) and JITL-based approach, using transacting inference (MWtr). $S19$ used the hierarchical, non-parametric sequential change-detection proposed by [85]. In, $S26$ the authors modified the acceptance-rejection sampling method to detect CD, calling their method IncLKDE.

C. RQ2- WHICH MACHINE LEARNING ALGORITHM IS USED TO ADAPT TO CONCEPT DRIFT IN NON-STATIONARY REGRESSION ENVIRONMENTS?

The objective of this question is to identify machine learning approaches/models, in particular, ensemble models, which seek to improve generalization performance when dealing with non-stationary data environments in regression tasks. All selected studies answered this research question. In order to identify the techniques that are most frequently used in this scenario, we divided the papers into nine categories, namely: Bayes, clustering, ensemble, fuzzy, framework, Neural Networks (NN), Tree, Support Vector Regression (SVR) and other algorithms. The category of other algorithms was used to contain papers where there is no clear indication of what algorithm was used. Similarly, in the case of the NN category, there is an "Other" subcategory, for when the actual NN architecture is unclear. The categories are not mutually exclusive, that is, a study can be in more than one. Table 6 shows the distribution of studies into categories and their respective density (number of papers per category).

We observed that for neural networks the most used technique is Online Sequential Extreme Learning Machine (OS-ELM), which employs incremental learning, i.e. it can be incrementally trained when data from new concepts arrive. Few studies use SVR, Tree and Bayes techniques. Fig. 7 shows the distribution of studies by their categorization and the intersections between them. Dotted circles mean that these studies do not use different techniques, while solid circles mean that at least one study in the category also uses at least one method from another category.

TABLE 6. Distribution of the different types of algorithms included in this systematic review.

Algorithm category	Papers	Density
Bayes	S5	1
Clustering	S2, S35	2
Ensemble	S1, S5, S4, S7, S10, S11, S12, S14, S15, S16, S17, S18, S21, S24, S25, S26, S29, S30, S33, S35, S37, S39, S40	23
Fuzzy	S2, S3, S11, S38	4
Framework	S20, S21, S23, S25, S26	5
NN	ELM - S18 OS-ELM - S1, S6, S7, S8, S9, S22, S34, S36, S39 MLP - S2 Other- S11, S24, S27, S38, S40, S41	17
Tree	S28, S32, S33	3
SVR	S19	1
Other Algoritms	S13, S31	2

We note that studies using ensembles of neural networks are considerably common in CD for regression. In $S1$ and $S7$, the authors proposed an approach based on Dynamic and

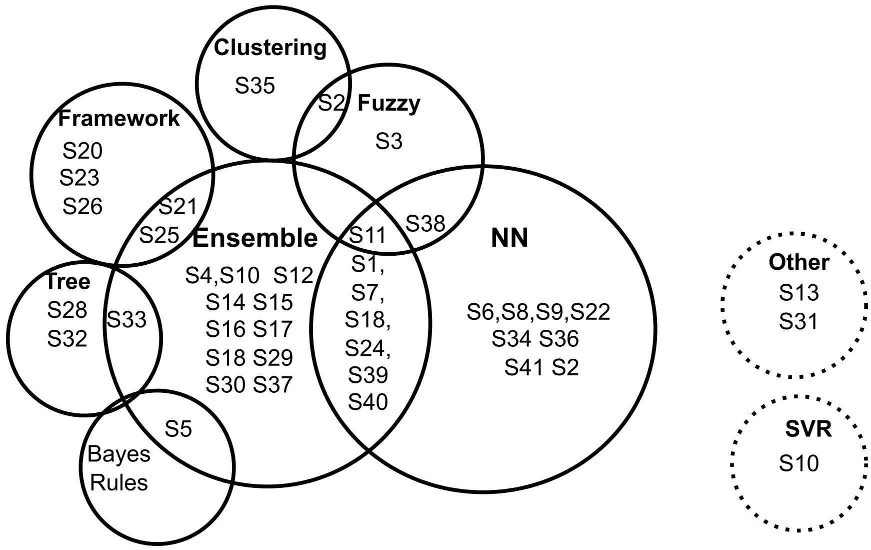


FIGURE 7. Distribution of studies by category.

Online Ensemble Regression (DOER) using the OS-ELM algorithm as the base learner. OS-ELM was chosen because of its fast training, however it suffers from high variance due to initialization. $S39$ uses the DOER algorithm and compares it with ensembles based on OS-ELMs. In $S1$, the algorithm performed with errors smaller than 1%, demonstrating it is a promising approach for the study (modeling of plants) since the error is smaller than the requirement, i.e., 1% considered a good margin in the study. The $S7$ also performs the requirement in actual plant operation of the general MAPE prediction error: < 1% on simulated and real data. In addition to scalability for different configurations, and it is a simple approach. In $S39$, the proposed method was able to provide more accurate predictions when compared to the traditional SW approach using a commonly used single model. Therefore, this technique can be applied in industrial applications since it reduces traditional systems' maintenance time and costs.

In the work $S4$, it is proposed the Simple Adaptive Batch Local Ensemble (SABLE) technique. This ensemble operates with data batch where a base model can be created in each new batch having the possibility of the dimensionality reduction feature. If the previous batch performs better than the batch that is in use, the model will be changed considering the parameters of the current batch, otherwise, the model will not be changed. The study made different configurations for this ensemble. The one that presented the lowest errors is the one that uses Cross-validation. This was empirically found to be the most effective as the process was stable for the multiple data batch.

In $S5$, the author used an ensemble composed of estimators based on Bayes's rule, with each estimator using a different confidence threshold. The method performance is compared with the overall prediction cost of the base version of an algorithm. Overall, the results indicate when the accuracy is high, it significantly improves the overall cost of forecasting. When the accuracy is low, the overall cost of forecasting is the same as that achieved by forecasting over time. $S10$ proposed an ensemble, called TW-FE-Adaboost, which considers the time factor for each sample, adopting a strategy based on prediction accuracy in order to determine the weights of the ensemble's weak learners. The proposed model had desirable performance in most cases, since it reduces the value of MAPE by 1.94% in the forecast results compared to the SVM approach; also reduces the MAPE value by 1.88% comparing the Adaboost-SVM model, and comparing with the ARIMA Model it can reduce the MAPE reaching up to 17.8%. The results indicate that the algorithm can provide stable predictions for the wind power grid. $S11$ employed a fuzzy neural network, called pseudo-incremental ensemble rough set pseudo-outer product (PIE-RSPOP) and which is also used to learn and predict trends in complex time series, by adapting four mechanisms which are based on how human beings learn and memorize concepts. The accuracy of the proposed model is not satisfactory for non-linear data due to the inadequacy of the cluster used in the model. However, the model works well for time series forecasting, where the latest measurements are used to predict future measurements despite the data having complex trends and rapidly changing. In $S13$, the authors developed a novel adaptive learning algorithm based on Relevance Vector Machine (RVM). The algorithm combines active and passive learning, is able to adapt the size of the sliding window and uses JITL in order to find appropriate appropriate historical data for the new concepts and to update the weights of the models. The prediction accuracy and robustness of the proposed method are superior to both conventional and ensemble methods developed for adaptive learning.

In $S12$, they combined Machine Learning techniques aiming to reduce the risk of using a prediction model that is inadequate for the current data concept, assuming that, at any given time, a different model might be the best option. The proposed model provides higher prediction accuracy for both classification and regression cases. For $S14$, the author introduced Online Weighted Euclidean Distance (JITL-OWED) models, which are simple to implement, computationally efficient and often significantly outperform traditional JITL models. The predictions of the proposed model are superior to those of the conventional JITL method in various configurations, datasets, and tuning scenarios, indicating that industrial application may be viable. In $S15$, the same author combined moving windows and JITL with lasso regression, producing a robust and stable method, capable to adapt to various types of CD. When the RMSE metric is observed, we realize that the proposed models provide superior prediction accuracy for most cases compared to the traditional MW model.

For $S16$, the authors proposed an incremental heterogeneous ensemble model using different approaches to calculate the weights with Genetic Algorithms and Particle Swarm Optimization. The method combines multiple regression and time series models in order to capture data seasonality and is able to adapt to changes in the distribution of the target variable. The best results were achieved by weighting calculated by the PSO method in the real and synthetic datasets. However, for one case this approach is not a superior configuration. In $S17$, the same authors proposed an incremental heterogeneous ensemble technique, which is robust and parallelizable, which makes it adequate to process data streams in a big data environment. The ensemble model predictions were statistically better than the individual models' results, i.e., Wilcoxon's hypotheses test. The average MAPE was approximately 1.3%.

$S18$ proposed a novel prediction method, called Ensemble Real-time Sequential Extreme Learning Machine (ERS-ELM), which runs an initial fast training on historical data and then incrementally learns using a sliding window, obtaining high performance on short-term traffic flow prediction. For the all scenarios, the proposed model is superior considering the accuracy and time consumption compared to classical WAVE-NN, MLP-NN, and classical ELM Methods. The performance on an public online dataset shows the superior advantage and generalization of ERS-ELM under different conditions. This method was improved in $S21$ with an approach that allows it to discard weak or obsolete models from the ensemble. The proposed method is tested on real-world traffic volume datasets and is proven to be more accurate than conventional incremental and ensemble methods, especially when concept drift occurs.

$S24$ proposed a bagging ensemble method, which allows various model update strategies to deal with CD. The work evaluated three main update approaches: pruning, substitution and weighting. The proposed model showed an average improvement of 8% in prediction accuracy for all datasets used, and the computational time up to 47 times lower than the compared model. In $S27$, a densely connected Neural Network is proposed to make a food sales forecast for a lead multiplexes company in India. The results show that the models compared ARIMA or S-ARIMA due to lack of resources are not able to capture the variation of the data and recurrent networks. However, the proposed model outperforms these models by around 7.7%, saving, on average, about 170 units of food per day. $S29$ proposes an ensemble of regression models, which is capable to keep previous knowledge that can be useful for recurring scenarios and learns incrementally in the presence of various types of CD. The proposed model achieves better accuracy than other state-of-the-art methods compared. Most of the time, the ensemble model outperforms the single model, and the ensemble model number is essential to consider. Similarly, for $S25$, the authors proposed a framework, called BRIGHT, which is able to deal with different kinds of CD. BRIGHT also aims at reducing the risk of overfitting, by guaranteeing diversity among its base learners. For one BRIGHT's configurations, it model achieved reductions for generalization error of up to 19%, and variance error reduction up to 50%.

$S30$ used an ensemble-based approach applied to multidimensional problems. Their method dynamically recalculates the coefficients of the regression functions, taking into account not only the current data, but previous samples as well. The approach outperformed other regression algorithms for stream data, such as sliding window regression and incremental stream regression. The experiment results show that the proposed model performs better for the data stream than the SWR method and incremental regression method, and it is more efficient in terms of storage and processing time. In $S33$, the authors performed a comparison of various models based on decision trees for online regression. According to their evaluation, ensembles of Hoeffding trees are a good choice for streams of big data, due to their processing speed. The online option trees have faster learning and are more accurate, whereas ensemble models are slower, however if implemented by parallelization, the execution time would be reduced.

$S37$ introduced an ensemble method that adapts to sudden CD by using a penalization scheme, such that in the absence of drift, the ensemble's base learners perform similarly across data batches, while in the presence of drift, the method is able to select relevant models and reduce the effects of CD. The proposed models can adaptively track the CD. The penalized ensemble is preferable to the weighted average ensemble, and even better suited to sudden changes.In $S40$ is focused on evaluating overall measures is an extension of the work of Bueno *et al.* (2017) [86]. This method uses a dynamic adaptation procedure of learners using a weighted average for better adaptation of the learners to the deviation. The results show that the performance of the proposed model is statistically better than all the algorithms in comparison, except ARIMA. Their domain number and specificity were insufficient to differentiate them statistically.

Regarding approaches based on clustering, we highlight $S2$, where the authors used the well-known fuzzy kernel c-means (FKCM) along with a lazy learning (LL) algorithm. The idea is to update the training set using lazy learning, then FKCM is used to find the best training set for the current concept. The LL is simple to model, however it requires more effort to choose the neighbors of the learning set. The combined model shows significant improvement over LL in terms of accuracy and can overcome recurring deviations error. For $S35$, the pseudo outer-product fuzzy neural network (RSPOP) algorithm is improved with incremental learning by using incremental clustering. This makes the system robust against deficiencies in the training set. Issues with the incremental rough set attribute reduction are also addressed by the approach. Due to the two-phase adaptation, the proposed model works well for time series. However, when modeling non-linear models or not time series datasets. The results vary depending on the order of the data, but in terms of the size and training quality of the datasets the results are promising.

In $S6$, the authors propose the use of alternating OS-ELM models, with one model learning stable concepts from a large sliding window and another one learning faster-changing concepts from a small window. The method selects, updates and reset the two models dynamically, by monitoring their performances, maintaining accuracy along the data stream. The proposed algorithm outperforms other baseline and benchmark methods and meets the industry standard of 1% performance for energy prediction. OS-ELM is also employed in $S8$, where ECDD and DDM are applied to decide when the OS-ELM model needs to be trained, achieving similar performance to conventional OS-ELM approaches, which usually update the model with every data batch,but requiring much less processing time. The experiments showed that ELM with DDM, and ELM with ECDD presented lower computational costs but lower accuracy than OS-ELM, especially in the real world time series. Furthermore, OS-ELM with ECDD was able to reduce processing time when compared to with unique OS-ELM in all-time series approaches tested.This is a similar technique to $S9$, which uses DDM and a meta-cognitive approach to decide when to update the OS-ELM model. The results show that the proposed model can improve the prediction performance in synthetic and real-world datasets. In addition, the model reduces training time due to the threshold present in the model.

In the work $S22$, it is proposed the Meta-cognitive Recurrent Recursive Kernel Online Sequential Extreme Learning Machine with Drift Detector Mechanism (meta-RRKOS-ELM-DDM). For this approach, the detection method can identify the DC in time series prediction and improves prediction accuracy. In additional, this metacognitive learning strategy has good performance in reducing learning time and solves the parameter dependency. The results indicate good performance in terms of accuracy and computational time for the proposed model. For $S34$, the authors used a combination of OS-ELM and Constructive Enhancement OS-ELM (CEOS-ELM), aiming to handle real, virtual and hybrid drifts, which can be either recurrent or sudden. The proposed model has better adaptive capacity than the non-adaptive OS-ELM and CEOS-ELM, including the presence of CD. However, the accuracy of the proposed model is not be better than the non-adaptive sequential ELM. For real dataset, the non-adaptive ELM is better when the distribution of data is known. OS-ELM was also adapted in $S36$, where the authors proposed the forgetting parameters extreme learning machine (FP-ELM), which is able to forget previously learning weights, improving its generalization performance. In generally, the FP-ELM achieves similar performance to AddExp(ELM) and outperforms SEA(ELM). It also is faster training than the other tested approaches, and generalization performance is not sensitive to the regularization parameter.In $S38$, Recurrent Interval-Valued Metacognitive Scaffolding Fuzzy Neural Network (RIVMcSFNN) is combined two sound theories in cognitive psychology–metacognitive learning, the method is used because it presents totally flexible resources and computationally efficient learning principles. Therefore, it is able to automate the parameters with respect to the nonlinearity of the data distribution. The proposed method outperformed in four aspects: predictive accuracy, fuzzy rule, runtime, and training samples the other models.

In $S41$, it is presented an Artificial Neural Network Model based on linear algorithm proposed by Fontenela-Romero *et al.* (2010) [87]. This algorithm is able to forget the learning (weights) in the cost function through adapting dynamically new data. Considering the dataset with gradual drift, the proposed method starts with a small error that steadily increases as the data distribution changes in the three configurations. But a dataset with the best behavior is achieved by the proposed method using the exponential configuration. When considering the simulation of the vibration analysis of a bearing, the proposed method obtains a more accurate approximation, specifically in the last part of the data.

Regarding papers in the Frameworks category, $S20$ proposed a framework that combines multitask regression to increase the effective sample size of each task by augmenting it with pseudo-labeled instances generated from the training data of other related tasks with a series of optimal transport steps. The results show that the proposed framework outperforms all other methods tested for all datasets. $S23$ presented the P-ART (Predictive -Adaptive Real Time) framework with a novel concept drift compensation technique to make the predictions closer to reality by taking into account long term traffic variations. At the same time, near real time updates of the prediction models take care of sudden short term variations. The results show the proposal has small errors and executes fast so that the data can be used to take corrective actions. $S26$ proposed the $\mathrm{EnsP}^{\mathrm{KDE}}$ & $\mathrm{IncL}^{\mathrm{KDE}}$ frameworkd, which integrates Dynamic Ensemble Pruning (DEP), Incremental Learning (IL) and Kernel Density Estimate (KDE). The approach dynamically selects appropriate predictor sets based on the kernel density distribution of all base learners' prediction values. The proposed model has superior predictive performance in most datasets, fully confirming the model's effectiveness.

In the Fuzzy logic category, we highlight $S3$ which uses fuzzy models to handle uncertainty in the outputs of the predictive models. In $S3$, the generalized fuzzy models of Takagi-Sugeno (TS) are combined and compared with the model itself without using fuzzy. It was verified that the fuzzy join requires lower computational cost. However, the generalization error is similar in the models, thus there was no significant advance for his proposal comparing the canonical one.

Finally, among Tree and SVR papers, $S19$ proposes an approach for online training with Support Vector Regression (SVR), by combining Feature Vector Selection (FVS) and Incremental and Decremental Learning. The proposed approach modifies the model only when certain drift patterns are detected according to proposed criteria. The proposed approach shows that it is able to reduce the number of data in the model and learn the input patterns fast. Regarding MSE and MARE errors, the approach proved effective, with accuracy comparable to that of the best reference methods. For data with noise, the proposal also presented is robust in some cases. In $S28$, the authors proposed an algorithm that integrates two incremental decision tree models with a drift detection engine, which can keep regression models up to date at any time. In terms of performance, the model manages to reduce the error for the study datasets. In terms of computational time, it is a similarity in performance. $S31$ introduced an adaptation of two existing methods for incremental learning to improve prediction accuracy. The explanation methodology is combined with a state-of-the-art concept drift detector and a visualization technique to enhance explainability in dynamic streaming settings. Results are promising as they improve accuracy for the ten of the eleven datasets used. And finally, for $S32$, regression trees weree used with online and offline training, as they achieve good performance for software effort estimation (SEE). The proposed approach significantly improves performance compared to the valuable cross-company (CC) model. However, the results show that the online changing nature of software forecasting tasks should be explored.

D. RQ3 -WHAT PROBLEMS/CHALLENGES ARE ENCOUNTERED IN DETECTING AND ADAPATATION CONCEPT DRIFT IN NON-STATIONARY REGRESSION ENVIRONMENTS

The idea behind this question is to verify any challenges for detecting Concept Drift and adapting machine learning models to Concept Drift. The problem presented in the papers is related to selecting and defining the ideal batch value to be used in data streams. The work $S2$ addresses batch size, as well as the dynamic management of models in ensembles, as old concepts can reappear. The authors claim it is necessary to define the batch value and window in the algorithm initialization and as long as the data stream comes, the old data is forgotten and those new data are inserted into the algorithm. However, the forgetting process presents negative points once the coming data cannot be enough to update the algorithm properly since the current data can represent the problem better, and the concept drift for new data can take more time to update the algorithm in order to represent the problem. In the work $S6$, the authors evaluated the impact of batch sizes when retraining the model in the presence of CD. If the batch is too big, knowledge about previously observed data distributions is lost, whereas, with small batches, it is harder to learn new concepts. The authors for $S6$ are still looking for new approaches to get outcomes with high quality, for that they have used two methods: a "long-memory learner (L) that is trained on a long time window (LW) with samples relevant to the current concept, and a short-memory (S) learner that is trained on a short window (SW) that only contains most recent samples". In this way, the defined batch and the window size can be tested with the idea: if there is no deviation, the LW data will have the same distribution and will be more adequate, if there is deviation, the SW data will be more adequate, since the model S will not be trained with LW data, having the ability to react faster to deviation.

E. RQ4- WHICH DATASETS WERE USED BY THE AUTHORS?

Since we selected 41 works for this systematic literature review, and the dataset used for its proposals validation can be real or simulated data. Many works used both types of data, we divided the studies into only synthetic (6 works-14.64%), only real/benchmarks(17 works-41.46%) and both (18 works-43.90%).

1) SIMULATED DATASETS

$S5$ The simulated dataset used in this work as described as follow: Drifts of controlled magnitude; Drifts of controlled type, frequency, and area of effect; Local expending abrupt drift; Global reoccurring abrupt drift; Global and slow gradual drift; Comparison between stable and drifting concept.$S8$ The authors made three datasets based on time series problems using the auto-regressive process.

For the work $S22$, the authors created three artificial time series without concept drift and three time series with concept drift, an auto-regressive process was applied. $S25$ uses a Tennessee Eastman Process (TEP) [88] industrial process simulator, with added deviations. Whereas, for the $S34$, the authors defined three datasets with 50000 training/5000 testing using the sinc, sinus, Gaussian functions. In the $S41$, it was used two sets of synthetic data with 500 different training samples and test data: the first being formed with four explanatory variables (generated following a normal distribution with zero mean and standard deviation equal to 0.1) output is obtained by mean sof a linear mixture; the second with four explanatory variables with normal distribution with zero mean and equal standard deviation a 0.1, and output is obtained by a linear mixture of the four input variables.

2) REAL DATASETS

For the work $s1$, it was used three real data sets were collected from three power plants as described: Two of them are 2×1combined cycle power plants (2gas turbines and 1 steam turbine). The other is a 3×1 combined cycle power plant, with 3 gas turbines and 1 steam turbine. In summary, the result of this process is a dataset with 3 ambient-related input variables - ambient temperature, ambient pressure, ambient relative humidity,and 3 operation mode-related variables for each gas turbine. $S2$The authors used the Ailerons Dataset in which addresses a control problem, namely flying a F16 aircraft. The attributes describe the status of the aeroplane, while the goal is to predict the control action on the ailerons of the aircraft. For more details, this is the link of the dataset published [89]. $S4$ this work, the authors used two different types of datasets from real-world problems: A Novel, Integrated Adsorber [90]/Catalyst [91] Oxidizer for TCCS. $S6$ real dataset is made by time series sensors readings from a power plant. A collection of 9 signals related to turbine operating status are simultaneously recorded: Compressor inlet temperature;Compressor inlet humidity; Ambient pressure;Inlet pressure drop Exhaust pressure drop;Inlet guide vane angle;Fuel temperature;Compressor flow; Controller calculated firing temperature. And, it contains 500 datasets and 2000 samples with different kinds of inputs and outputs. $S10$ In this study, wind speed time series from four sites in Hexi Corridor of China is used to test the proposed combined approach. $S11$ It was used a series of benchmark experiments are performed, viz., chaotic Mackey–Glass time series (Mackey Glass, 1977) [92], Nakanishi datasets (Nakanishi, Turksen, Sugeno, 1993) [93], and stock price prediction (Das *et al.*, 2016) [78].

For $S12$ This work relies on public transport data made available by the city of Warsaw. The data is provided through the Open Data portal of the city availableat [94] and is publicly available. It takes the form of GPS-based location records produced in near-real time for individual trams. The system includes 26 tram lines with an overall length of over 360km. Further details on the Warsaw tram system can be found in [95]. $S15$ Debutanizer Column (DC) [96], Sulfur Recovery Unit (SRU) [97], Sequential-Reactor-Multi-Grade (SRMG) [98]. In the work $S17$, the authors used a sample of data from smart meters installed in Slovakia that take measurements every 15 minutes. In the $S18$, it is used two datasets: Caltrans Performance Measurement Systems (PeMS version 14.0) database, and traffic flow data from the PeMS, these data is collected every thirty seconds on the California highway system. For the work $S21$, the data colleted are from Caltrans Performance Measurement Systems (PeMS) [99] in the California highway system in the time period 2016-11-6 to 2016-12-3. While, for the $S27$, it was used a dataset of a collection of transactions belonging to multiple food items, in $S28$, the authors used a dataset for commercial flights within the United States from October 1987 to April 2008 [100]. For the $S31$, it uses the electricity consumption dataset for the state of New York, USA [101]. $S32$ five datasets are used: ISBSG2000, ISBSG2001, ISBSG [102], CocNasaCoc81 and CocNasaCoc81Nasa93 [103]. And finally, for the $S33$ work, it is used the Infobiotics PSP datasets protein structure data, City traffic dataset generated for competition data part of IEEE ICDM2010.5, Airline dataset [104], and in $S35$, three datasets: highway traffic flowmodelling and prediction [105], real life stock data price forecasting, and real life stock volatility forecasting.

3) REAL AND SIMULATED DATASETS

In $S3$ the synthetic dataset $y(t) = \frac{\sum_{i=1}^{m} y(t-i)}{1+\sum_{i=1}^{m} y(t-1)^2} + u(t-1)$, where $u(t) = sin(2\pi k/20)$, $y(t) = 0$ for all $j = 1 \dots m$ and the automatic prediction of resistance value in rolling mill processes, respectively. $S7$ Multiple synthetic and real (industry applications) - GE Power simulation tool, known as GTP (Gas Turbine Performance). $S9$ his work, it was used three time series data sets without concept driftproblem–containing 20.035 values each set–are generated by autoregressive processas follows: $Xt = 1.5Xt - 1 - 0.4Xt - 2 - 0.3Xt - 3 + 0.2Xt - 4 + wt(TS1)$, $Xt = -0.1Xt - 1 + 1.2Xt - 2 + 0.4Xt - 3 - 0.5Xt - 4 + wt(TS2)$;$Xt = 0.9Xt - 1 + 0.8Xt - 2 - 0.6Xt - 3 + 0.2Xt - 4 - 0.5Xt - 5 - 0.2Xt - 6 + 0.4Xt - 7 + wt(TS3)$. hen these sets are combined to new time series with concept drift problem–the concept drift appears in the 10001-st value in the time series. Where, $TS4$ is a combination of $TS1$ and $TS2$ while $TS5$ is a combination of $TS3$ and $TS1$.$TS6$ is a combination of $TS3$ and $TS2$. Furthermore, the authors employ two real-world data in the experiments, i.e., Shanghai Stock Exchange Composite Index (SSE) and ozone concentration of Toronto(Ozone).

For $S13$ the real dataset is obtained from a real industrial sequential distillation process, containing 2394 samples and seven input variables [96]; The polymer manufacturing process plant dataset consists of 331 observations of 29 process variables collected over two years. [106]; The wastewater treatment plant dataset consists of 360 observations with 11 explanatory variables and one dependent variable (fluorine concentration in the effluent stream). [107]; The synthetic dataset is based on an isothermal CSTR model, in which a first-order exothermic reaction $A- > B$ takes place, to which data are added virtual deviations [108].

$S14$ authors use Friedman synthetic dataset [109], real datasets: Gas Multi-Sensor Device (GMSD) [110], Debutanizer Column (DC) [96], Waste Water Treatment Plant (WWTP) [96] used to remove polluting agents from acid gas, Sulfur Recovery Unit (SRU) [107]. For $S16$ The real datasets used Slovakia's intelligence metering system was only available for the quarter-hourly power load measurements for 20 regions. To create two synthetic datasets, they used the sinus function to simulate the shape of the daily energy charge. $S19$ four datasets are created, the authors use the Friedman function to create three datasets following the process of [76], and the fourth uses the Hyperplane function generated according to [72].

$S20$ use two synthetic datasets, the first a response drift due to translation, and the second captures a response drift due to rotation. The authors also used actual data Lake Ecology Data [111] which contains 13 explanatory and four dependent variables. $S23$ synthetic data using the queuing-theoretic model and real data using CloudLab [112]. $S24$ use four synthetic datasets: 3-D Mex. Hat (Mex), Friedman #1 (Fried1), Friedman # 3 (Fried3), Multi (Multi), following the process of [113] and four real datasets California housing (Housing) [114], Wine quality (Quality), Condition-based maintenance (Maintenance), Appliances energy prediction (Energy) [104]. For $S26$ a synthetic dataset called: Mackey-Glass dataset (MKG) and five real datasets were used: Mackey-Glass dataset (MKG), International airline passengers (IAP), Volume of money, ABS definition m1 (VOM), IBM common stock closing prices (ICS), monthly closings of the Dow-Jones industrial index (MCD) and quarterly increase in total non-farm stocks (QIS). $S29$ the Friedman function is used to create three synthetic datasets based on [115], also used the hyperplane and the real dataset: FCCU.

$S30$ the experiments were evaluated on three synthetic datasets with three different types of drift based on [115] and a real dataset Tropical Atmosphere Ocean (TAO) [116]. $S36$ use synthetic data: Hyperplane [117] and real datasets: Abalone, AutoMPG [104], Space-ga [118]. $S37$ the authors also use abruptly shifted synthetic data using the $\sum_{i=1}^{10} \alpha_i + X_i + \epsilon$, where X is the independent variable, and ϵ the associated error, and actual datasets: o Combined Cycle Power Plant (CCPP) Data [104] and Airplane data [119].

$S38$ uses real dataset of valuation of real estate prices in Poland from 11 year-period from 1998 until 2008 [120]. $S39$ uses synthetic dataset: hyperplane with gradual and recurrent drifts and Friedman's [115], and real datasets: polymerization reactor (catalyst activity) [121], cement kiln process free lime (CaO) is Provided by "A Control - Automação e Controle Industrial, Lda.", Coimbra, Portugal, debutanizer column (butane concentration) [122], powder detergent production (NOx concentration) [123], thermal oxidizer sulfur recovery unit (SRU) - H2O concentration (output 1) and S2O concentration (output 2) [122]. $S40$ synthetic data: hyperplane and Friedman's function. Actual datasets: Sulfur Recovery Unit (SRU), Debutanizer Column [72] and, PM datasets de 01-jan-2012 a 01-jan-2015in the cities of Jundiaí, São Caetano do Sul, and Campinas (Brazil).

F. RQ5- WHAT PROBLEMS/CHALLENGES ARE PRESENTED IN REAL/SIMULATED DATA SETS USING REGRESSION MODELS FOR SOLVING CONCEPT DRIFT?

According to the authors, the most challenging problem faced in the Concept Drift is to figure out how to update the algorithm to still working on new arrives without deprecating the results for predicted data, i.e., $\bar{y}$. This is related in [49]. in which the authors describes that to deal with different kinds of data is a big problem. A more detailed example of this is presented in [46]. In this work, at first, the authors tried to use a window size for the new out-coming data, and the oldest 200 data instances are abandoned from the learning set. To figure out the solution, the authors proposed an algorithm based on Fuzzy Kernel c-means Clustering (FKCM), that they called it as KLL. According to the authors, it was well performed for high-dimensional distances complex problems presented in FKCM's kernel function.

G. RQ6- WHICH ASSESSMENT METRICS ARE USED?

For this question we seek to identify the evaluation techniques both in the deviation detection process and in machine learning methods. However, most studies use the prediction error to evaluate their proposals. Thus, the 7 Table presents the evaluation metrics used in each study, namely: Mean Absolute Error (MAE), Root Mean Absolute Error (RMAE), Mean Squared Error(MSE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), Symmetric Mean Absolute Percentage Error (SMAPE), Mean Absolute Relative Error (MARE), Determination coefficient (R^2)

TABLE 7. Metrics used in the studies.

Metric	Study	Density
MAE	S2 S3 S4 S10 S12 S13 S25 S26 S27 S28 S32	11
RMAE	S23	1
MSE	S6 S7 S10 S19 S22 S24 S29 S33 S38 S39 S40 S41	12
RMSE	S2 S3 S4 S11 S13 S14 S15 S18 S20 S21 S23 S26 S30 S35 S36 S37 S38	17
MAPE	S1 S6 S7 S8 S10 S16 S18 S21 S22 S27	10
SMAPE	S9 S17 S22 S25	4
MARE	S19	1
R^2	S11 S13 S14 S15 S23 S35	6
Other	Kutosis - S2 maxEa -S13 Time - S40 Redution in the overall cost - S5 Q Statistc - S31 The testing accuracy and Cohen's Kappa - S34	5

H. ANALYSIS AND DISCUSSION

We carried out a qualitative analysis of data extraction from the research questions. We consider the creation of a cloud of more frequent words, as well as the elaboration of a taxonomy based on the highest density of responses.

1) BAG OF WORDS

We created a Word Cloud of answers to questions $RQ1$ and $RQ2$, shown in Fig. 8. Thus, as expected, the words ensemble, DDM, regression, fuzzy were highlighted. Other highlighted words were sequential and incremental important words in the context of CD, as many works use the learning of models sequentially. Therefore, the input data are used continuously to extend the knowledge of the existing model, always training the model, fundamental in environments that have Concept Drift.

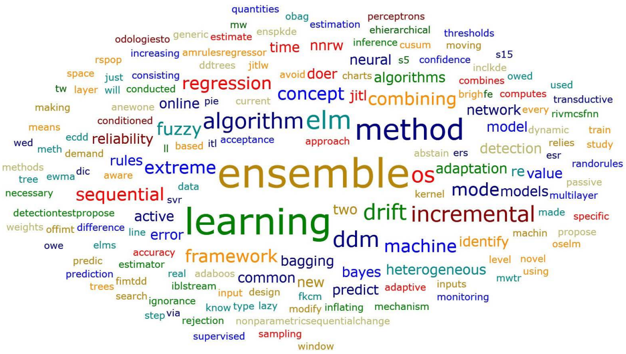


FIGURE 8. Word Cloud most used in study responses.

2) TAXONOMY

We developed a taxonomy with the answers to the research questions that have the highest density. Thus, we can consider it in future research. Fig. 9 displays the taxonomy. One can observe that Concept Drift is caused by an arbitrary change in the data distribution and that it can be of different types, the detection method can be associated with the ensemble NN, and OSELM, the ensemble, in turn, can be formed by NN, where the OSELM are a type of NN. These methods can be evaluated by different metrics such as Mean Absolute Percentage Error (MAPE), Mean Absolut Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). In additional, we be used in different datasets, where the batch size needs to be defined, the computational cost measured, which is the cause of the batch size, as well as the rule of evolution of models can be changed by different types of drift.

Therefore, we can, as initial tests, test approaches that use some detection method, where it is necessary to define the form whether passive or active. Also, define whether to use ensemble models composed by OSELM, or the individual model and the best way to define the batch size as input to the model.

3) DISCUSSIONS

Data acquisition was performed using reference bases commonly used in computing, which contributes to providing reliable results. For didactic reasons, we classified the studies according to the predictive methods used to identify the most common techniques on CD for regression. In general, OS-ELM ensembles and NN approaches are more used. Most of the selected studies use OS-ELM. This can be explained by the characteristic of the method of achieving a considerable speed in the training process. In addition, it is essential to use a model that can be updated incrementally. This model is proper when amounts of data arrive along the way time. As there is data available to be trained, the model can be updated constantly.

In general, we can still observe that some problems reported by studies to deal with CD are identifying the batch size when the model must be updated again, and the ability to react to deviations as there are different ways they can suggest in the data. In addition, there is more frequent is to Root Mean Squared Error (RMSE) evaluation on real and synthetic data. We note that Friedman's hyperplane functions are often used in the creation of synthetic datasets in CD cases, as well as some benchmark datasets: Airline and Abalone datasets [104]. Each dataset has its characteristics. However,

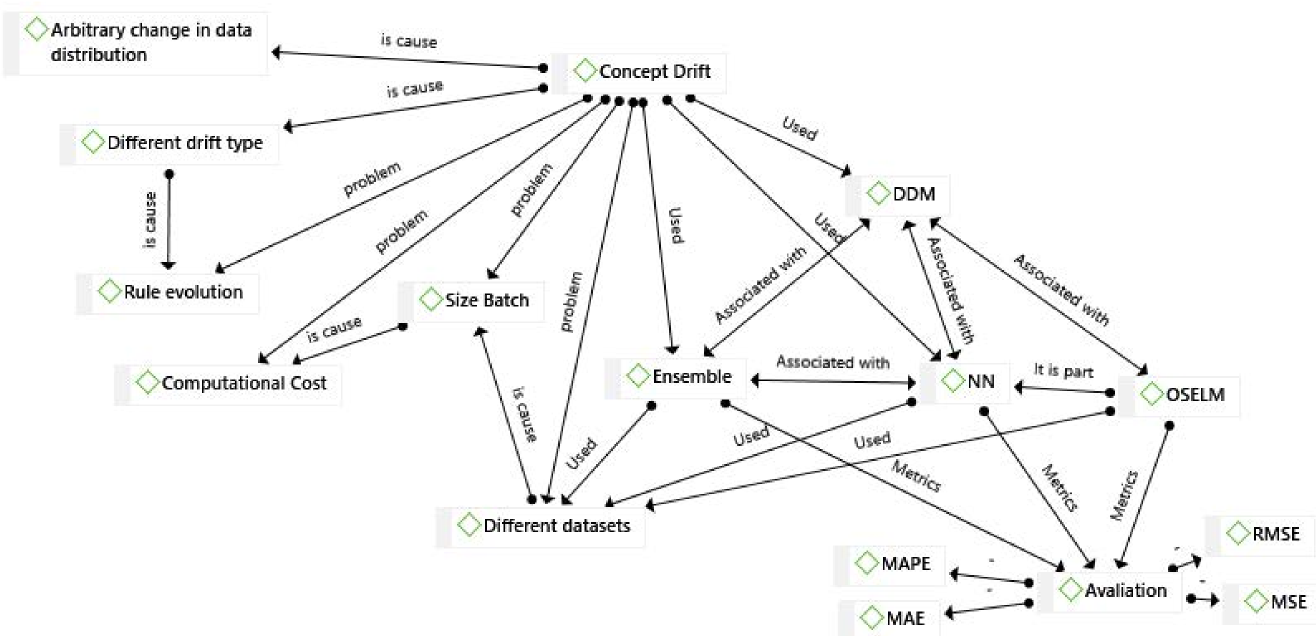


FIGURE 9. Taxonomy based on density of answers to research questions.

in the case of the speed of the CD, in most of the works there is no justify the reason for CD use. In addition, due to the characteristics of OS-ELM, it is the most frequently used.

V. THREATS TO VALIDITY

We categorized the threats validity proposed by Wohlin *et al.* (2012) [124]. Thus, we categorized into four types of validity threats in our work:

- Construct validity: it refers to the generalization of the result to the concepts behind the study [124], that is, it deals with whether the researcher measures the results [125]. For example, in addition to the keyword ''Concept Drift'' it uses five more synonyms. We built a taxonomy with methods that had more frequency in the answers to the research questions.
- Internal validity: it is related to possible wrong conclusions in the causal relationships of the results [124], that is, it checks whether the result makes a difference or not, and if what is raised there is evidence [125]. According to Vilela *et al.* (2017) [126] with in the context of SLR study is always to minimize the internal validity of the research. Therefore, in our work, we seek to minimize the researcher's bias, that is, the conduct of the research was carried out by more than one survey, with a doctoral student who conducts research in the field of machine learning, another master and with experience in the market in the field of learning and two other PhDs with research in the area.
- External validity: it is related to the generalization of the SLR results, that is, whether the discovery of the results is relevant [125]. In our work, to mitigate these threats, we carried out a sensitivity study of the string keywords if we search in consensus with at least two authors. In addition to defining our inclusion, exclusion and validity criteria to exclude articles from the gray literature.
- Conclusion validity: this is related to the degree to which the conclusions are presented within the collected data [125]. In our work, we seek to eliminate the researcher's bias, for this we defined the research protocol carefully validated by the authors, as well as the analysis of the results.

VI. CONCLUSION

In this work, we conducted an SLR on Concept Drift for regression with the main objective of synthesizing the existing knowledge on the subject. Different aspects were addressed, such as the challenges of the CD, the datasets used, the evaluation metrics, in addition to a density taxonomy. Our SLR is based on 41 studies from 1546, through a step-sequence process. An important feature is that it is not restricted to a machine learning technique or application context. This broader scope gives us broader insights into the state of the art of content. The most relevant findings and their implications for future research are: (1) DDM drift detectors play an important part in CD detection research for regression; (2) Regarding machine learning techniques, ensemble techniques, followed by NN are the most used, having, therefore, an increasingly important role; (3) We implicitly identify that most existing drift detection and adaptation algorithms assume that the dependent variable (true label) is available; (4) Fuzzy logic techniques and Framework proposal have been addressed on CD for regression; (5) There is no data flows analysis of the CD aspect for regression, such as drift occurrence point, drift severity, drift shape; (6) The number of studies grows per year recognizes the importance of working with CD for regression.

Therefore, based on these findings, this SLR generated some research observations for analysis: (1) What is the way to compose an ensemble and its merge method to work in CD environments for regression? (2) evaluate the composition of drift detection methods with ensemble and NN in the context of CD for regression?. (3) in which types of deviation approaches with drift and ensemble detection methods can be more efficient in the CD context for regression?. (4) to what extent can the combination of drift detection methods and machine learning techniques be efficient in dynamic environments with CD for regression?. (5) What is the best way to measure the cost-effectiveness of using machine learning techniques to deal with CD for regression?.

REFERENCES

[1] A. S. Iwashita and J. P. Papa, "An overview on concept drift learning," *IEEE Access*, vol. 7, pp. 1532–1547, 2019.

[2] Q. Zhang, K. Yu, Z. Guo, S. Garg, J. Rodrigues, M. M. Hassan, and M. Guizani, "Graph neural networks-driven traffic forecasting for connected internet of vehicles," *IEEE Trans. Netw. Sci. Eng.*, early access, Nov. 17, 2021.

[3] Z. Guo and H. Wang, "A deep graph neural network-based mechanism for social recommendations," *IEEE Trans. Ind. Informat.*, vol. 17, no. 4, pp. 2776–2783, Apr. 2021.

[4] N. Lu, G. Zhang, and J. Lu, "Concept drift detection via competence models," *Artif. Intell.*, vol. 209, pp. 11–28, Apr. 2014.

[5] N. L. A. Ghani, I. A. Aziz, and M. Mehat, "Concept drift detection on unlabeled data streams: A systematic literature review," in *Proc. IEEE Conf. Big Data Anal. (ICBDA)*, Nov. 2020, pp. 61–65.

[6] A. A. Khine and H. W. Khin, "Credit card fraud detection using online boosting with extremely fast decision tree," in *Proc. IEEE Conf. Comput. Appl. (ICCA)*, Feb. 2020, pp. 1–4.

[7] D. Chasman, N. Iyer, A. F. Siahpirani, M. E. Silva, E. Lippmann, B. McIntosh, M. D. Probasco, P. Jiang, R. Stewart, J. A. Thomson, R. S. Ashton, and S. Roy, "Inferring regulatory programs governing region specificity of neuroepithelial stem cells during early hindbrain and spinal cord development," *Cell Syst.*, vol. 9, no. 2, pp. 167–186, 2019.

[8] F. Shen, Y. Liu, R. Wang, and W. Zhou, "A dynamic financial distress forecast model with multiple forecast results under unbalanced data environment," *Knowl.-Based Syst.*, vol. 192, Mar. 2020, Art. no. 105365.

[9] V. M. dos Santos, R. F. de Mello, T. Nogueira, and R. A. Rios, "Quantifying temporal novelty in social networks using time-varying graphs and concept drift detection," in *Proc. Brazilian Conf. Intell. Syst.* Springer, 2020, pp. 650–664.

[10] R. K. Jagait, M. N. Fekri, K. Grolinger, and S. Mir, "Load forecasting under concept drift: Online ensemble learning with recurrent neural network and ARIMA," *IEEE Access*, vol. 9, pp. 98992–99008, 2021.

[11] G. Widmer and M. Kubat, "Learning in the presence of concept drift and hidden contexts," *Mach. Learn.*, vol. 23, no. 1, pp. 69–101, 1996.

[12] L. L. Minku and X. Yao, "DDD: A new ensemble approach for dealing with concept drift," *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 4, pp. 619–633, Apr. 2012.

[13] D. Brzezinski and J. Stefanowski, "Reacting to different types of concept drift: The accuracy updated ensemble algorithm," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 1, pp. 81–94, Jan. 2014.

[14] P. M. Gonçalves, Jr., S. G. T. D. C. Santos, R. S. M. Barros, and D. C. L. Vieira, "A comparative study on concept drift detectors," *Expert Syst. Appl.*, vol. 41, no. 18, pp. 8144–8156, Dec. 2014.

[15] H. Guo, S. Zhang, and W. Wang, "Selective ensemble-based online adaptive deep neural networks for streaming data with concept drift," *Neural Netw.*, vol. 142, pp. 437–456, Oct. 2021.

[16] A. Abbasi, A. R. Javed, C. Chakraborty, J. Nebhen, W. Zehra, and Z. Jalil, "ElStream: An ensemble learning approach for concept drift detection in dynamic social big data stream learning," *IEEE Access*, vol. 9, pp. 66408–66419, 2021.

[17] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 12, pp. 2346–2363, Dec. 2019.

[18] J. C. Schlimmer and R. H. Granger, "Beyond incremental processing: Tracking concept drift," in *Proc. AAAI*, 1986, pp. 502–507.

[19] J. Z. Kolter and M. A. Maloof, "Dynamic weighted majority: An ensemble method for drifting concepts," *J. Mach. Learn. Res.*, vol. 8, pp. 2755–2790, Dec. 2007.

[20] M. Kubat and G. Widmer, "Adapting to drift in continuous domains," in *Proc. Eur. Conf. Mach. Learn.* Springer, 1995, pp. 307–310.

[21] R. Klinkenberg and T. Joachims, "Detecting concept drift with support vector machines," in *Proc. ICML*, 2000, pp. 487–494.

[22] W. N. Street and Y. Kim, "A streaming ensemble algorithm (SEA) for large-scale classification," in *Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2001, pp. 377–382.

[23] M. Scholz and R. Klinkenberg, "An ensemble classifier for drifting concepts," in *Proc. 2nd Int. Workshop Knowl. Discovery Data Streams*, vol. 6, no. 11, Porto, Portugal, 2005, pp. 53–64.

[24] A. Bifet and R. Gavaldà, "Learning from time-changing data with adaptive windowing," in *Proc. SIAM Int. Conf. Data Mining*. Philadelphia, PA, USA: SIAM, Apr. 2007, pp. 443–448.

[25] L. L. Minku, A. P. White, and X. Yao, "The impact of diversity on online ensemble learning in the presence of concept drift," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 5, pp. 730–742, May 2010.

[26] R. Elwell and R. Polikar, "Incremental learning of concept drift in nonstationary environments," *IEEE Trans. Neural Netw.*, vol. 22, no. 10, pp. 1517–1531, Oct. 2011.

[27] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Comput. Surv.*, vol. 46, no. 4, pp. 1–37, 2014.

[28] I. Žliobaitė and J. Hollmén, "Optimizing regression models for data streams with missing values," *Mach. Learn.*, vol. 99, no. 1, pp. 47–73, Apr. 2015.

[29] A. Liu, Y. Song, G. Zhang, and J. Lu, "Regional concept drift detection and density synchronized drift adaptation," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, Aug. 2017, pp. 1–7.

[30] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with drift detection," in *Proc. Brazilian Symp. Artif. Intell.* Springer, 2004, pp. 286–295.

[31] M. Baena-Garcıa, J. del Campo-Ávila, R. Fidalgo, A. Bifet, R. Gavalda, and R. Morales-Bueno, "Early drift detection method," in *Proc. 4th Int. Workshop Knowl. Discovery Data Streams*, vol. 6, 2006, pp. 77–86.

[32] G. Ross, N. M. Adams, D. K. Tasoulis, and D. J. Hand, "Exponentially weighted moving average charts for detecting concept drift," *Pattern Recognit. Lett.*, vol. 33, no. 2, pp. 191–198, Jan. 2012.

[33] S. H. Bach and M. A. Maloof, "Paired learners for concept drift," in *Proc. 8th IEEE Int. Conf. Data Mining*, Dec. 2008, pp. 23–32.

[34] C. Raab, M. Heusinger, and F.-M. Schleif, "Reactive soft prototype computing for concept drift streams," *Neurocomputing*, vol. 416, pp. 340–351, Nov. 2020.

[35] C. Alippi and M. Roveri, "Just-in-time adaptive classifiers—Part I: Detecting nonstationary changes," *IEEE Trans. Neural Netw.*, vol. 19, no. 7, pp. 1145–1153, Jul. 2008.

[36] T. Dasu, S. Krishnan, S. Venkatasubramanian, and K. Yi, "An information-theoretic approach to detecting changes in multidimensional data streams," in *Proc. Symp. Interface Statist., Comput. Sci., Appl.* Princeton, NJ, USA: Citeseer, 2006.

[37] I. Frías-Blanco, J. D. Campo-Ávila, G. Ramos-Jiménez, R. Morales-Bueno, A. Ortiz-Díaz, and Y. Caballero-Mota, "Online and non-parametric drift detection methods based on Hoeffding's bounds," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 3, pp. 810–823, Mar. 2015.

[38] A. Liu, G. Zhang, and J. Lu, "Fuzzy time windowing for gradual concept drift adaptation," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, Jul. 2017, pp. 1–6.

[39] J. Gama and G. Castillo, "Learning with local drift detection," in *Proc. Int. Conf. Adv. Data Mining Appl.* Springer, 2006, pp. 42–55.

[40] L. I. Kuncheva, "Classifier ensembles for detecting concept change in streaming data: Overview and perspectives," in *Proc. 2nd Workshop SUEMA*, 2008, pp. 5–10.

[41] S. Keele *et al.*, "Guidelines for performing systematic literature reviews in software engineering," Citeseer, Princeton, NJ, USA, Tech. Rep. EBSE, 2007.

[42] A. Idri, M. Hosni, and A. Abran, "Systematic literature review of ensemble effort estimation," *J. Syst. Softw.*, vol. 118, pp. 151–175, Aug. 2016.

[43] R. de Almeida, Y. M. Goh, R. Monfared, M. T. A. Steiner, and A. West, "An ensemble based on neural networks with random weights for online data stream regression," *Soft Comput.*, vol. 24, no. 13, pp. 9835–9855, Jul. 2020.

[44] M. Petticrew and H. Roberts, *Systematic Reviews in the Social Sciences: A Practical Guide*. Hoboken, NJ, USA: Wiley, 2008.

[45] R. Xu and W. Yan, "Continuous modeling of power plant performance with regularized extreme learning machine," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–8.

[46] Y. Song, G. Zhang, J. Lu, and H. Lu, "A fuzzy kernel c-means clustering model for handling concept drift in regression," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, Jul. 2017, pp. 1–6.

[47] E. Lughofer, "Efficient sample selection in data stream regression employing evolving generalized fuzzy models," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, Aug. 2015, pp. 1–9.

[48] R. Bakirov, B. Gabrys, and D. Fay, "On sequences of different adaptive mechanisms in non-stationary regression problems," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2015, pp. 1–8.

[49] P.-X. Loeffel, V. Lemaire, C. Marsala, and M. Detyniecki, "Improving the prediction cost of drift handling algorithms by abstaining," in *Proc. IEEE 16th Int. Conf. Data Mining Workshops (ICDMW)*, Dec. 2016, pp. 1213–1222.

[50] Y. Xu, R. Xu, W. Yan, and P. Ardis, "Concept drift learning with alternating learners," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 2104–2111.

[51] Y. Xu, R. Xu, and W. Yan, "Power plant performance modeling with concept drift," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 2096–2103.

[52] R. C. Cavalcante and A. L. I. Oliveira, "An approach to handle concept drift in financial time series based on extreme learning machines and explicit drift detection," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2015, pp. 1–8.

[53] Z. Liu, C. K. Loo, and K. Pasupa, "Handling concept drift in time-series data: Meta-cognitive recurrent recursive-kernel OS-ELM," in *Proc. Int. Conf. Neural Inf. Process.* Springer, 2018, pp. 3–13.

[54] L. Xiao, Y. Dong, and Y. Dong, "An improved combination approach based on AdaBoost algorithm for wind speed time series forecasting," *Energy Convers. Manage.*, vol. 160, pp. 273–288, Mar. 2018.

[55] A. R. Iyer, D. K. Prasad, and C. H. Quek, "PIE-RSPOP: A brain-inspired pseudo-incremental ensemble rough set pseudo-outer product fuzzy neural network," *Expert Syst. Appl.*, vol. 95, pp. 172–189, Apr. 2018.

[56] M. Grzenda, K. Kwasiborska, and T. Zaremba, "Hybrid short term prediction to address limited timeliness of public transport data streams," *Neurocomputing*, vol. 391, pp. 305–317, May 2020.

[57] A. Urhan and B. Alakent, "Integrating adaptive moving window and just-in-time learning paradigms for soft-sensor design," *Neurocomputing*, vol. 392, pp. 23–37, Jun. 2020.

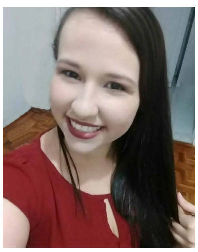
[58] B. Alakent, "Online tuning of predictor weights for relevant data selection in just-in-time-learning," *Chemometric Intell. Lab. Syst.*, vol. 203, Aug. 2020, Art. no. 104043.

[59] B. Alakent, "Soft sensor design using transductive moving window learner," *Comput. Chem. Eng.*, vol. 140, Sep. 2020, Art. no. 106941.

[60] G. Grmanová, V. Rozinajová, A. B. Ezzedine, M. Lucká, P. Lacko, M. Lóderer, P. Vrablecová, and P. Laurinec, "Application of biologically inspired methods to improve adaptive ensemble learning," in *Advances in Nature and Biologically Inspired Computing*. 2016, pp. 235–246.

[61] G. Grmanová, P. Laurinec, V. Rozinajová, A. B. Ezzeddine, M. Lucká, P. Lacko, P. Vrablecová, and P. Návrat, "Incremental ensemble learning for electricity load forecasting," *Acta Polytech. Hungarica*, vol. 13, no. 2, pp. 97–117, 2016.
[62] D. Wang, J. Xiong, Z. Xiao, and X. Li, "Short-term traffic flow prediction based on ensemble real-time sequential extreme learning machine under non-stationary condition," in *Proc. IEEE 83rd Veh. Technol. Conf. (VTC Spring)*, May 2016, pp. 1–5.
[63] J. Liu and E. Zio, "An adaptive online learning approach for support vector regression: Online-SVR-FID," *Mech. Syst. Signal Process.*, vols. 76–77, pp. 796–809, Aug. 2016.
[64] B. Liu, P.-N. Tan, and J. Zhou, "Augmented multi-task learning by optimal transport," in *Proc. SIAM Int. Conf. Data Mining*. Philadelphia, PA, USA: SIAM, 2019, pp. 19–27.
[65] J. Xiao, Z. Xiao, D. Wang, J. Bai, V. Havyarimana, and F. Zeng, "Short-term traffic volume prediction by ensemble learning in concept drifting environments," *Knowl.-Based Syst.*, vol. 164, pp. 213–225, Jan. 2019.
[66] Z. Liu, C. K. Loo, and M. Seera, "Meta-cognitive recurrent recursive kernel OS-ELM for concept drift handling," *Appl. Soft Comput.*, vol. 75, pp. 494–507, Feb. 2019.
[67] L. Gupta, R. Jain, D. Bhamare, and A. Erbad, "The P-ART framework for placement of virtual network services in a multi-cloud environment," *Comput. Commun.*, vol. 139, pp. 103–122, May 2019.
[68] A. Saadallah, L. Moreira-Matias, R. Sousa, J. Khiari, E. Jenelius, and J. Gama, "BRIGHT—Drift-aware demand predictions for taxi networks," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 2, pp. 234–245, Feb. 2020.
[69] G. Zhu and Q. Dai, "EnsP^{KDE}&IncL^{KDE}: A hybrid time series prediction algorithm integrating dynamic ensemble pruning, incremental learning, and kernel density estimation," *Appl. Intell.*, vol. 51, no. 2, pp. 617–645, 2021.
[70] V. A. Ganesan, S. Divi, N. B. Moudhgalya, U. Sriharsha, and V. Vijayaraghavan, "Forecasting food sales in a multiplex using dynamic artificial neural networks," in *Proc. Sci. Inf. Conf.* Springer, 2019, pp. 69–80.
[71] Z. Liao and Y. Wang, "Rival learner algorithm with drift adaptation for online data stream regression," in *Proc. Int. Conf. Algorithms, Comput. Artif. Intell.*, Dec. 2018, pp. 1–5.
[72] S. G. Soares and R. Araújo, "An on-line weighted ensemble of regressor models to handle concept drifts," *Eng. Appl. Artif. Intell.*, vol. 37, pp. 392–406, Jan. 2015.
[73] C. H. Nadungodage, Y. Xia, P. S. Vaidya, Y. Chen, and J. J. Lee, "Online multi-dimensional regression analysis on concept-drifting data streams," *Int. J. Data Mining, Model. Manage.*, vol. 6, no. 3, pp. 217–238, 2014.
[74] Z. Bosnić, J. Demšar, G. Kešpret, P. P. Rodrigues, J. Gama, and I. Kononenko, "Enhancing data stream predictions with reliability estimators and explanation," *Eng. Appl. Artif. Intell.*, vol. 34, pp. 178–192, Sep. 2014.
[75] L. L. Minku and X. Yao, "Can cross-company data improve performance in software effort estimation?" in *Proc. 8th Int. Conf. Predictive Models Softw. Eng.*, 2012, pp. 69–78.
[76] E. Ikonomovska, J. Gama, and S. Džeroski, "Online tree-based ensembles and option trees for regression on evolving data streams," *Neurocomputing*, vol. 150, pp. 458–470, Feb. 2015.
[77] A. Budiman, M. I. Fanany, and C. Basaruddin, "Adaptive online sequential ELM for concept drift tackling," *Comput. Intell. Neurosci.*, vol. 2016, pp. 1–17, Aug. 2016.
[78] R. T. Das, K. K. Ang, and C. Quek, "ieRSPOP: A novel incremental rough set-based pseudo outer-product with ensemble learning," *Appl. Soft Comput.*, vol. 46, pp. 170–186, Sep. 2016.
[79] D. Liu, Y. Wu, and H. Jiang, "FP-ELM: An online sequential learning algorithm for dealing with concept drift," *Neurocomputing*, vol. 207, pp. 322–334, Sep. 2016.
[80] L.-Y. Wang, C. Park, K. Yeon, and H. Choi, "Tracking concept drift using a constrained penalized regression combiner," *Comput. Statist. Data Anal.*, vol. 108, pp. 52–69, Apr. 2017.
[81] M. Pratama, E. Lughofer, M. J. Er, S. Anavatti, and C.-P. Lim, "Data driven modelling based on recurrent interval-valued metacognitive scaffolding fuzzy neural network," *Neurocomputing*, vol. 262, pp. 4–27, Nov. 2017.
[82] S. G. Soares and R. Araújo, "A dynamic and on-line ensemble regression for changing environments," *Expert Syst. Appl.*, vol. 42, no. 6, pp. 2935–2948, Apr. 2015.
[83] A. Bueno, G. P. Coelho, and J. R. B. Junior, "Dynamic ensemble mechanisms to improve particulate matter forecasting," *Appl. Soft Comput.*, vol. 91, Jun. 2020, Art. no. 106123.
[84] D. Martínez-Rego, B. Pérez-Sánchez, O. Fontenla-Romero, and A. Alonso-Betanzos, "A robust incremental learning method for non-stationary environments," *Neurocomputing*, vol. 74, no. 11, pp. 1800–1808, May 2011.
[85] C. Alippi, G. Boracchi, and M. Roveri, "A hierarchical, nonparametric, sequential change-detection test," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 2011, pp. 2889–2896.
[86] A. Bueno, G. P. Coelho, and J. R. Bertini, "Online sequential learning based on extreme learning machines for particulate matter forecasting," in *Proc. Brazilian Conf. Intell. Syst. (BRACIS)*, Oct. 2017, pp. 169–174.
[87] O. Fontenla-Romero, B. Guijarro-Berdiñas, B. Pérez-Sánchez, and A. Alonso-Betanzos, "A new convex objective function for the supervised learning of single-layer neural networks," *Pattern Recognit.*, vol. 43, no. 5, pp. 1984–1992, 2010.
[88] N. L. Ricker, "Decentralized control of the Tennessee Eastman challenge process," *J. Process Control*, vol. 6, no. 4, pp. 205–221, Aug. 1996.
[89] *Dataset Ailerons*. Accessed: Oct. 2020. [Online]. Available: https://www.openml.org/search?type=data&q=Ailerons
[90] *Dataset Novel Intergrated Catalyst*. Accessed: Oct. 2020. [Online]. Available: https://data.nasa.gov/dataset/A-Novel-Intergrated-Adsorber-Catalyst-Oxidizer-for/sk84-faqn
[91] (Oct. 2020). *Dataset catalyst*. [Online]. Available: https://github.com/Open-Catalyst-Project/ocp/blob/master/DATASET.md
[92] M. C. Mackey and L. Glass, "Oscillation and chaos in physiological control systems," *Science*, vol. 197, no. 4300, pp. 287–289, 1977.
[93] H. Nakanishi, I. B. Turksen, and M. Sugeno, "A review and comparison of six reasoning methods," *Fuzzy Sets Syst.*, vol. 57, no. 3, pp. 257–294, Aug. 1993.
[94] *Api Warszawa*. Accessed: Oct. 2020. [Online]. Available: http://api.um.warszawa.pl
[95] A. Zychowski, K. Junosza-Szaniawski, and A. Kosicki, "Travel time prediction for trams in Warsaw," in *Proc. Int. Conf. Comput. Recognit. Syst.* Springer, 2017, pp. 53–62.
[96] L. Fortuna, S. Grazizni, A. Rizzo, and G. M. Xibilia, *Soft Sensors for Monitoring and Control of Industrial Processes*, vol. 22. Springer, 2007.
[97] J. Opsomer, Y. Wang, and Y. Yang, "Nonparametric regression with correlated errors," *Stat. Sci.*, vol. 16, no. 2, pp. 134–153, 2001.
[98] H. Lou, H. Su, L. Xie, Y. Gu, and G. Rong, "Inferential model for industrial polypropylene melt index prediction with embedded priori knowledge and delay estimation," *Ind. Eng. Chem. Res.*, vol. 51, no. 25, pp. 8510–8525, Jun. 2012.
[99] *Pems*. Accessed: Oct. 2020. [Online]. Available: https://pems.dot.ca.gov/
[100] [Online]. Available: http://stat-computing.org/dataexpo
[101] [Online]. Available: http://www.nyiso.com/public/markets_operations/market_data/load_data/index.jsp
[102] [Online]. Available: http://www.isbsg.org
[103] [Online]. Available: http://promise.site.uottawa.ca/SERepository
[104] *Dataset UCI*. Accessed: Oct. 2020. [Online]. Available: https://archive.ics.uci.edu/ml/datasets.html
[105] A. Singh, C. Quek, and S.-Y. Cho, "DCT-Yager FNN: A novel Yager-based fuzzy neural network with the discrete clustering technique," *IEEE Trans. Neural Netw.*, vol. 19, no. 4, pp. 625–644, Apr. 2008.
[106] Y. Liu, Z. Gao, and J. Chen, "Development of soft-sensors for online quality prediction of sequential-reactor-multi-grade industrial processes," *Chem. Eng. Sci.*, vol. 102, no. 11, pp. 602–612, Oct. 2013.
[107] F. A. A. Souza, R. Araújo, T. Matias, and J. Mendes, "A multilayer-perceptron based method for variable selection in soft sensor design," *J. Process Control*, vol. 23, pp. 1371–1378, Nov. 2013.
[108] S. Yoon and J. F. MacGregor, "Fault diagnosis with multivariate statistical models. Part I: Using steady state fault signatures," *J. Process Control*, vol. 11, no. 4, pp. 387–400, Aug. 2001.
[109] J. H. Friedman, "Multivariate adaptive regression splines," *Ann. Statist.*, vol. 19, no. 1, pp. 1–67, Mar. 1991.
[110] S. De Vito, E. Massera, M. Piga, L. Martinotto, and G. D. Francia, "On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario," *Sens. Actuators B, Chem.*, vol. 129, no. 2, pp. 750–757, 2008.
[111] P. A. Soranno *et al.*, "LAGOS-NE: A multi-scaled geospatial and temporal database of lake ecological context and water quality for thousands of US lakes," *GigaScience*, vol. 6, no. 12, 2017, Art. no. gix101.

[112] R. Ricci and E. Eide, "Introducing cloudlab: Scientific infrastructure for advancing cloud architectures and applications," *Mag. USENIX SAGE*, vol. 39, no. 6, pp. 36–38, 2014.
[113] J. Ding, H. Wang, C. Li, T. Chai, and J. Wang, "An online learning neural network ensembles with random weights for regression of sequential data stream," *Soft Comput.*, vol. 21, no. 20, pp. 5919–5937, Oct. 2017.
[114] *Cmu*. Accessed: Oct. 2020. [Online]. Available: https://lib.stat.cmu.edu
[115] E. Ikonomovska, "Algorithms for learning regression trees and ensembles on evolving data streams," Jožef Stefan Int. Postgraduate School, Ljubljana, Slovenia, Tech. Rep. 1, 2012, vol. 1.
[116] *Tao*. Accessed: Oct. 2020. [Online]. Available: http://www.pmel.noaa.gov/tao/
[117] G. Hulten, L. Spencer, and P. Domingos, "Mining time-changing data streams," in *Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2001, pp. 97–106.
[118] *Datasets CMU*. Accessed: Oct. 2020. [Online]. Available: http://lib.stat.cmu.edu/datasets/
[119] E. Ikonomovska, J. Gama, R. Sebastião, and D. Gjorgjevik, "Regression trees from data streams with drift detection," in *Proc. Int. Conf. Discovery Sci.* Springer, 2009, pp. 121–135.
[120] E. Lughofer, B. Trawiński, K. Trawiński, O. Kempa, and T. Lasota, "On employing fuzzy modeling algorithms for the valuation of residential premises," *Inf. Sci.*, vol. 181, no. 23, pp. 5123–5142, Dec. 2011.
[121] P. Kadlec, R. Grbić, and B. Gabrys, "Review of adaptation mechanisms for data-driven soft sensors," *Comput. Chem. Eng.*, vol. 35, no. 1, pp. 1–24, 2011.
[122] [Online]. Available: http://www.springer.com/engineering/control/book/978-1-84628-479-3
[123] R. Grbić, D. Slišković, and P. Kadlec, "Adaptive soft sensor for online prediction and process monitoring based on a mixture of Gaussian process models," *Comput. Chem. Eng.*, vol. 58, no. 22, pp. 84–97, 2013.
[124] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in Software Engineering*. Science Business Media, Springer, 2012.
[125] A. Ampatzoglou, S. Bibi, P. Avgeriou, M. Verbeek, and A. Chatzigeorgiou, "Identifying, categorizing and mitigating threats to validity in software engineering secondary studies," *Inf. Softw. Technol.*, vol. 106, pp. 201–230, Feb. 2018.
[126] J. Vilela, J. Castro, L. E. G. Martins, and T. Gorschek, "Integration between requirements engineering and safety analysis: A systematic literature review," *J. Syst. Softw.*, vol. 125, pp. 68–92, Mar. 2017.



MARÍLIA LIMA received the degree in information systems from the University of Pernambuco Campus Caruaru, in 2017, and the master's degree in computer engineering from the University of Pernambuco, in 2019, where she is currently pursuing the Ph.D. degree in computer engineering with the Polytechnic School of Pernambuco, in the field of computer science concentration. She has experience in the field of computer science, with an emphasis on computational intelligence.

MANOEL NETO received the bachelor's degree in information systems from Faculdade Joaquim Nabuco, in 2015, and the master's degree in computer engineering from Escola Politécnica de Pernambuco (UPE), in 2018. He has been working in the technology field, since 2010, and from 2015 to 2016, he studied English at the Vancouver International College. Currently, he is working as a Senior Data Scientist with C.E.S.A.R. His main research interests include regression, optimization algorithms, natural language problems, machine learning algorithms, deep learning, statistics, and innovation.

TELMO SILVA FILHO received the graduate degree in computer science, the master's degree in artificial intelligence, and the Ph.D. degree in computer science having as a research topic machine learning from the Federal University of Pernambuco, in 2010, 2013, and 2017, respectively, and the Sandwich Ph.D. degree from the University of Bristol, U.K. He is currently an Adjunct Professor with the Federal University of Paraíba. He has experience in computer science, with an emphasis on machine learning.

ROBERTA A. DE A. FAGUNDES (Member, IEEE) received the graduate degree in telematics technology from the Federal Center for Technological Education of Paraíba (CEFET-PB), in 2002, the master's and Ph.D. degrees in computer science, and the Ph.D. degree in statistics from the Federal University of Pernambuco (UFPE), Brazil, in 2006, 2013, and 2015, respectively. She is currently an Adjunct Professor of information systems and computer engineering with the University of Pernambuco (UPE), Brazil. She is also a Vice-Coordinator and a Professor with the Graduate Program in Computer Engineering (PPGEC), where there are masters and doctorate courses. Her research interest includes computer science, with an emphasis on computational intelligence.