manuscript No.
(will be inserted by the editor)

The impact of data difficulty factors on classification of imbalanced and concept drifting data streams

Dariusz Brzezinski · Leandro Minku · Tomasz Pewinski · Jerzy Stefanowski · Artur Szumaczuk

Received: date / Accepted: date

Abstract Class imbalance introduces additional challenges when learning classifiers from concept drifting data streams. Most existing work focuses on designing new algorithms for dealing with the global imbalance ratio and does not consider other data complexities. Independent research on static imbalanced data has highlighted the influential role of local data difficulty factors such as minority class decomposition and presence of unsafe types of examples. Despite often being present in real-world data, the interactions between concept drifts and local data difficulty factors have not been investigated in concept drifting data streams yet. We thoroughly study the impact of such interactions on drifting imbalanced streams. For this purpose, we put forward a new categorization of concept drifts for class imbalanced problems. Through comprehensive experiments with synthetic and real data streams, we study the influence of concept drifts, global class imbalance, local data difficulty factors, and their combinations, on predictions of representative online classifiers. Experimental results reveal the high influence of new considered factors and their local drifts, as well as differences in existing classifiers' reactions to such factors. Combinations of multiple factors are the most challenging for classifiers. Although existing classifiers are partially capable of coping with global class imbalance, new approaches are needed to address challenges posed by imbalanced data streams.

Keywords class imbalance · concept drift · data difficulty factors · drift categorization · stream classification

1 Introduction

Classification methods have impressively developed in the last decades. Although most of the research in the field is concerned with batch learning from static data

D. Brzezinski, T. Pewinski, J. Stefanowski, A. Szumaczuk
Institute of Computing Science & Center for Artificial Intelligence and Machine Learning, Poznan University of Technology, ul. Piotrowo 2, 60-965 Poznan, Poland;
L. Minku
School of Computer Science, University of Birmingham, Edgbaston, Birmingham B15 2TT, United Kingdom

repositories, recent years have seen more and more studies directed at the analysis of large data volumes dynamically generated in the form of *data streams*.

Compared to classifying static data, the task of learning from data streams introduces limits on computational resources [23] and forces classifiers to act in dynamic environments, where the data and target concepts change over time in a phenomenon called *concept drift* [79]. Examples of real-life data streams include spam categorization, weather predictions, and financial fraud detection [86]. Furthermore, many practical applications make learning classifiers from streams even more challenging by introducing additional data complexities. One such additional challenge is *class imbalance*, a situation where at least one of the target classes, called the minority class, is highly underrepresented in the data [34].

Separately, both concept drift and class imbalance have already received substantial research attention. Concept drift has been thoroughly analyzed in the last two decades, in particular in the context of non-stationary data streams [26, 18, 40, 33], resulting in drift taxonomies [79], detectors [26], evaluation techniques [84, 85], and adaptive streaming classifiers [8]. Research on class imbalance has also led to many novel methods, such as class resampling [21], specialized classification methods [34] or dedicated classifier performance measures [11, 12].

Furthermore, several recent works on static imbalanced data highlight the impact of *local data difficulty factors* on classification deterioration. These studies have shown the influence of such factors as the decomposition of the minority class concept into many sub-concepts [36, 37], class overlapping [29, 62], the presence of isolated rare groups of few minority class examples inside the majority class region located far from the class boundaries [57, 55]. Importantly, it has been shown that such data characteristics are common in real-world classification problems [20, 46, 56, 58, 69, 81], and can be more influential to the overall classifier performance than the global class imbalance ratio itself [37, 56].

Although class imbalance co-occurs with many practical data stream classification tasks, the number of specialized proposals for imbalanced streams is still relatively small, see e.g. [78]. Moreover, existing works on imbalanced stream classification mostly focus on re-balancing classes and reacting to changes affecting the global imbalance ratio. These works do not consider the aforementioned local difficulty factors manifested by changes in *local class distributions* and other *local drifts*. However, imbalanced streams can also be affected by these difficulty factors. For example, some minority classes can be fragmented into sub-concepts, which appear and disappear with time, or change their positions in the attribute space [38]. Moreover, drifting overlapping borderline regions between the minority and majority classes have also been observed in tweet streams [42], introducing additional difficulties for learning classifiers.

The conjunction of these data difficulty factors and concept drift may potentially be more challenging for classifiers than the impact of each factor separately, given that the classifier would need to adapt to local drifts based on very few minority class examples. However, the role of streaming local difficulty factors and local drifts has not been studied yet and is still an open research challenge. We claim that better understanding of the interactions between concept drifts and various difficulty factors in class imbalanced streams, will help diagnose and remedy more complex difficulties in learning from real-world data streams. These observations have led us to the following research questions:

RQ1 What is the impact of different types of single local data difficulty factors and single isolated drifts (i.e. drifts without the presence of other drifts in the data stream) on the predictive performance of online classifiers? Is it possible to identify which classifiers work better and when?
RQ2 What is the interaction between different types of local data difficulty factors and global class imbalance? Which local factors or drifts are the most demanding for classifiers at different imbalance ratios?
RQ3 Which complex scenarios integrating several data factors are the most harmful for classification performance?

In this paper, we answer the presented research questions by carrying out a comprehensive experimental study with five representative on-line classifiers applied to various synthetic and real-world data streams modelling the possible interactions between the global class imbalance ratio, local data difficulty factors, and concept drift. To the best of our knowledge, no such analysis has been previously done. To support our analysis and future work in the area, we also propose a categorization of concept drifts for imbalanced data streams and provide a data stream generator that takes into account all the presented factors. We pay special attention to controlled experiments on a wide collection of synthetic datasets, as these dissect the interactions between local data difficulties and classifier reactions to drifts in particular moments in time. Our analysis highlights the main not-yet-addressed challenges of imbalanced stream classification and discusses future directions of research in the field.

2 Problem formulation

A data stream can be defined as a sequence of labeled examples $\mathcal{S} = \{\mathbf{x}_t, \mathbf{y}_t\}_{t=1}^{T}$, where $\mathbf{x}_t \in \mathcal{X}$, $\mathbf{y}_t \in \mathcal{Y}$, $\mathcal{X}$ is the input space, $\mathcal{Y}$ is the output space, $(\mathbf{x}_t, \mathbf{y}_t) \sim_{i.i.d.} p_t(\mathbf{x}, \mathbf{y})$, where $p_t(\mathbf{x}, \mathbf{y})$ is the joint probability distribution underlying the data stream at time stamp t, and T may be infinite. In this paper, we will consider data streams where $\mathcal{Y} = \{-, +\}$. Therefore, we will refer to $\mathbf{y}$ as a binary label y.

We will investigate data streams produced in non-stationary environments, i.e., data streams that may suffer concept drift. Formally, it is said that a concept drift occurs if [23, 26, 40], for any two time stamps t and $t + \Delta$,

$$\exists x : p_t(\mathbf{x}, y) \neq p_{t+\Delta}(\mathbf{x}, y).$$

In particular, we will look into data streams which suffer from class imbalance at least during some period of time. A data stream is said to be imbalanced at a given time stamp t if $p_t(-) \ll p_t(+)$ or $p_t(+) \ll p_t(-)$. The class that contains less examples is called the *minority class*, whereas the remaining class is referred to as the *majority class*. In non-stationary environments, $p_t(-)$ and $p_t(+)$ may drift over time, therefore, the imbalance status of the problem may also change. To quantify the amount of class imbalance in a stream (or part of a stream), we will use the percentage of the cardinality of the minority class with respect to all examples in the stream $|\mathcal{S}_{min}|/|\mathcal{S}| \cdot 100\%$.

We consider classification problems where unlabelled examples $\mathbf{x}$ are received, and require their labels to be predicted. After some time, the true labels y_t of such examples are received and can be used to compose a training example $(\mathbf{x}_t, y_t)$.

Many classification problems operate in such scenarios [23], with examples including electricity price prediction, networks intrusion detection [49], software defect prediction [13], and credit card assessment [65].

In order to solve such problems, we will adopt *online* supervised *learning algorithms*. These algorithms work as follows [51]: assume that, at time stamp t, a model $f_{t-1} : \mathcal{X} \rightarrow \mathcal{Y}$ created based on past examples from $\mathcal{S}$ is available, and a new labelled example $(\mathbf{x}_t, y_t)$ becomes available. Online learning aims at building an updated model $f_t : \mathcal{X} \rightarrow \mathcal{Y}$ able to generalize to unseen examples of $p_t(\mathbf{x}, y)$, based on f_{t-1} and $(\mathbf{x}_t, y_t)$.

This online procedure differs from the so called *chunk learning algorithms* [40] in that it learns each training example as soon as it is received, rather than waiting for a whole chunk (block) of examples to arrive before updating the predictive model or its evaluation. This avoids (1) potential delays in reacting to concept drift, as there is no need to wait for a whole chunk of data before starting to react to drifts, and (2) the need for choosing a chunk size, which is particularly challenging in non-stationary environments [49]. It is also more adequate for applications with high time and memory constraints than typical chunk-based approaches, as each training example is processed once and only once.

3 Related work

3.1 Concept drifts

There are several different ways to categorize and characterize concept drifts in classification problems. A common way is to refer to the components of the joint probability distribution that are affected by the drift. The joint probability distribution can be written as $p(\mathbf{x}, y) = p(\mathbf{x}|y)p(y)$. Therefore, a concept drift affects one or both of the following components [26, 78]:

- prior probabilities of the classes $p(y)$ and
- class conditional probabilities $p(\mathbf{x}|y)$.

A joint probability distribution can also be written as $p(\mathbf{x}, y) = p(y|\mathbf{x})p(\mathbf{x})$. Therefore, a concept drift also affects one or both of the following components: [26, 18]:

- unconditional probability distribution $p(\mathbf{x})$ and
- posterior probabilities of the classes $p(y|\mathbf{x})$.

A concept drift affecting $p(\mathbf{x}|y)$ or $p(y|\mathbf{x})$ thus affects the relationship between the input and output of the problem. A concept drift affecting only $p(y)$ or $p(\mathbf{x})$ does not affect this relationship. In particular, it does not affect the true underlying decision boundaries of the problem.

In terms of the joint probability distribution, drifts can be further characterized by their severity or magnitude. Severity and magnitude are equivalent criteria that have been defined as the amount of changes in the joint probability distribution caused by a drift [50], and as the degree of difference between two points of time [79], respectively. Several different distance measures can be used to characterize drifts according to their severity or magnitude [32, 64, 50, 80], such as Kullback-Leibler Divergence, Hellinger Distance and Total Variation Distance. The Hellinger

Distance and the Total Variation Distance have the advantage of being symmetric. As the Total Variation Distance is more efficient to compute, it has been favoured in the literature [80]. It is defined as follows, where Z is a vector of discrete random variables and $dom(Z)$ is the domain of Z [80, 44]:

$$\sigma_{t,u}(Z) = \frac{1}{2} \sum_{z \in dom(Z)} |P_t(z) - P_u(z)|.$$

However, calculations of Hellinger Distance or the Total Variation Distance for continuous random variables make strong assumptions about the probability distributions, and so discretization of continuous random variables is recommended when using these metrics [80].

Drifts affecting $p(y|x)$ have been referred to as *severe* [50] or full-concept [79] drifts if they change the $p(y|x)$ of the whole input attribute space, whereas they have been referred to as *intersected* [50] or subconcept [79] drifts if they change the $p(y|x)$ of only part of the attribute space.

Another common way to categorize drifts is based on their rate of change [18], also known as speed [50]. Typically, drifts are categorized as *abrupt* (also called sudden) if they cause sudden changes, or *gradual* if their underlying joint probability distribution slowly evolves over time [18, 64]. Slow evolution can refer to a period of time where two distinct distributions co-exist in the problem, with the new distribution slowly taking over the old one [26, 50]. It can also refer to a distribution that continuously and incrementally changes by limited amounts over a period of time [64].

Other criteria such as recurrence, frequency, predictability and cyclical behaviour have also been used to categorize and characterize sequences of concept drifts [50, 18, 26], rather than single concept drifts. It is worth noting that several criteria can be used together to provide a richer description of concept drifts. For example, a drift may be categorized as affecting the posterior probabilities of the classes, being severe and gradual at the same time. However, none of the existing criteria address the problem of changes concerning local and global data difficulty factors of imbalanced streams.

3.2 Class imbalance and data difficulty factors

Research on class imbalance has resulted in numerous data preprocessing or algorithmic methods for static data; see reviews in [6, 20, 34]. Moreover, recent studies identified key data properties, which make imbalanced learning difficult. One such property is the *global imbalance ratio*, defined as the percentage of examples in the dataset that belong to the minority class (see Section 2). However, it has also been observed that strong class imbalance does not always deteriorate classification performance. It has led researchers to identify other data characteristics, called *data difficulty factors* or data complexities, as the sources of classifiers deterioration. Following the literature on static imbalanced data [20, 29, 46, 37, 56, 69], these difficulty factors include:

1. the decomposition of the minority class concept into several sub-concepts,
2. the presence of small, isolated groups of minority examples located deep inside the majority class region;

3. the effect of strong overlapping between the classes,

The second factor above corresponds to the so called rare cases, which are defined as isolated, very small, groups of minority examples (e.g., containing 1–3 examples) located more deeply inside regions of the majority class [55]. This is different from the first factor, which refers to the decomposition of the minority class into several larger sub-clusters containing more examples than rare cases and corresponding to sub-concepts of the minority class [36]. Both of these may or may not have overlapping between the classes, i.e., the third factor is not mutually exclusive with the first two factors. In particular, rare examples may be very near the examples of the majority class or mixed with them, presenting an overlap, or there might be no overlap. Similarly, minority class sub-concepts may or may not present overlaps near their decision boundaries [20].

Note that such difficultly factors deteriorate classification performance also in standard, balanced classification tasks. However, when these data complexity factors occur *together* with class imbalance, the deterioration of classifier performance is amplified and it affects mostly (or even only) the minority class [57].

Experimental studies on the role of the aforementioned factors have shown that data complexities occur in most static imbalanced datasets, and may play a key role in explaining the difference between the performance of various classifiers [56, 68]. In particular, proper handling of overlapping between classes and unsafe types of minority examples inspired recent proposals of new informed pre-processing [4], rule induction [54] and bagging ensemble generalizations [3], which outperformed many well known methods.

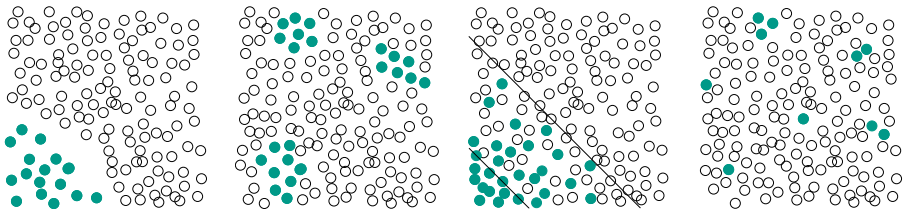
However, research on these local data difficulty factors has concerned mostly static data and there are few attempts to study them in data streams. In [76] data streams with different types of examples have been considered, but only in the scenarios with stationary underlying distributions. Other research concentrated either on classifier modifications [78, 39] or evaluation measures [10]. To the best of our knowledge, no existing work performed a comprehensive study of how concept drifts affecting local data distributions impact the predictive performance of various online class imbalance learning algorithms. There is also a lack of categorization and characterization of drifts that take such local data distributions into account, making systematic studies of the impact of such drifts difficult.

Below we further discuss the aforementioned difficulty factors and briefly describe their characteristics.

3.2.1 Decomposition of the minority class into sub-concepts

Experimental studies with various real-world imbalanced datasets show that the minority class usually does not form a homogeneous region (single concept) in the attribute space, but is often scattered into smaller sub-clusters (sub-concepts) spread over the space (Fig 1b) [20, 34, 56, 81]. Japkowicz *et al.* named this phenomenon *within-class imbalance* in contradiction to the global imbalance between classes. These authors initiated studies on the impact of *class decomposition* on performance of popular classifiers over several synthetic datasets [36, 37]. In their experiments the cluster forming the minority class distribution, initially surrounded by the majority class examples, was successively split into several sub-clusters, which were separated by the majority class regions. Their results

demonstrated that increasing the split of the minority class was more influential than changing the global imbalance ratio, in particular for a smaller number of examples. This was also independently studied for other more complex class distributions [62, 68]. According to Japkowicz this may be associated with, so called, *small disjuncts*, which originally referred to classification rules covering too few examples, contributing to classification errors more than rules covering more examples. The impact of small disjuncts arises particularly for the minority class. Identification of the minority class decomposition in real world static data influenced several studies on their discovery and improved resampling techniques or modifications of algorithms such as, e.g., cluster-based over-sampling techniques [36, 69].



(a) Safe minority class (b) Sub-clusters (c) Borderline region (d) Rare cases and outliers

Fig. 1: Minority class distributions with different difficulty factors. Minority examples depicted as filled (teal) circles, majority examples as hollow (white) circles.

3.2.2 Presence of rare minority examples

Occurrences of isolated minority examples located deeper inside regions of the majority class was identified as another difficulty factor. These are small "islets" of few examples (usually pairs, triplets) located quite far from the class boundaries (Fig 1d). Following [55], they are called *rare cases* and due to their rarity they are different from larger sub-concepts discussed in Section 3.2.1. As the minority class is under-represented they cannot be treated as noise [41]. Moreover, studies such as [56] have proved that if handled appropriately, e.g. by informed pre-processing, rare examples can be used to improve the recognition of the minority class.

Note that also single minority examples can be located inside the minority class or empty space region, playing the role of outliers [55]. However, their correct recognition is extremely difficult and most existing specialized methods treat outliers as noise. Therefore, in further experiments with imbalanced streams we focus our interest on rare cases only.

3.2.3 Overlapping between minority and majority classes

Other researchers considered difficulties linked to high *overlapping* between regions of minority and majority class examples in the attribute space (e.g., Fig. 1c). This difficulty factor is present in many real-life data and has been already identified

in balanced classification problems. However, experimental studies have shown that its role is even more influential for imbalanced data and the minority class in particular [29, 68]. For instance authors of [29] have shown that an increasing of overlapping region degraded performance of six popular classifiers much more than changing the imbalance ratio. Furthermore they observed the higher impact of the local imbalanced ratio in the overlapping region. Other researches also have shown that usefulness of various resampling methods depends on the amount of overlapping and that have to be dealt with appropriately while constructing new methods [46, 56].

3.2.4 Types of minority examples with respect to their local characteristics

A related view on data difficulty factors leads to distinguishing different types of minority examples based on the number of minority and majority class examples near them [41, 43, 55]. One of the approaches to distinguishing types minority examples [55, 56] analyzes class labels of examples in their local neighborhood defined by either by k-nearest neighbours or kernels. In this approach, an example of a given class is labeled as *safe* if most of its neighbours also belong to this class. If the numbers of labels from both classes is similar, then the example is labeled as a *borderline* one. If all of its neighbours are from opposite class the example is treated as an *outlier*, otherwise it is a *rare* case. The advantage of adopting this definition of safe, borderline, outlier and rare examples is that it is possible to determine the proportion of minority class examples of each of these types in real world data sets [55, 56].

3.3 Cluster transitions

Existing work on data stream clustering has attempted to model and monitor the evolution of clusters (cluster transitions) in data streams [66, 67]. These transitions can be external or internal transitions from a given time step t_i to another time step t_j, where $j > i$. An external transition of a given cluster can be categorised as a split into multiple clusters, cluster disappearance, emergence or absorption by another cluster. An internal transition of a given cluster can be a size transition where the cluster shrinks or expands, a compactness transition where the cluster becomes more compact or diffuse, or a local transition where the center or the distribution of the cluster changes. Even though the work presented in [66, 67] is not about concept drift in classification problems, it can potentially inspire new work on class imbalanced data streams, in view of the data difficulty factors discussed in Section 3.2. In particular, concept drifts such as decomposition of the minority class concept into several sub-concepts could potentially be monitored and modelled from a cluster perspective, where a given sub-concept could potentially be modelled by a cluster of minority class examples.

3.4 Classifiers for imbalanced data streams

Compared to the number of machine learning approaches designed to tackle balanced data streams [26, 18, 40, 33], there are relatively few approaches aimed

to tackle class imbalanced data streams [78]. In both settings, different types of concept drift require different strategies to be effectively and efficiently handled. *Active* approaches, which rely on a concept drift detector to trigger adaptation, tend to work well for abrupt drifts, but may struggle with gradual drifts. *Passive* approaches, which continuously update the learning system with incoming data in an attempt to tackle concept drift, are typically well suited for gradual drifts. In this section, we discuss classifiers for imbalanced data streams from these two perspectives. We give emphasis to approaches that perform online learning, which are the focus of this work, and only briefly discuss chunk learning algorithms.

Recursive Least Square Adaptive Cost Perceptron (RLSACP) [30] is a perceptron-based online passive approach. It uses a loss function that is both cost-sensitive and time-decayed, to tackle both class imbalance and concept drifts affecting $p(y|x)$, respectively. The cost associated with each class can vary over time, enabling these approaches to also cope with concept drifts affecting $p(y)$. The Online Neural Network (ONN) [31] is a similar approach, however, it assumes a fixed imbalance ratio and cannot cope with concept drifts affecting $p(y)$. Finally, Ensemble of Subset Online Sequential Extreme Learning Machine (ESOS-ELM) [52] is a perceptron-based passive-active learning approach that can operate in chunk or online mode, so long as an initialization chunk is provided. It trains an ensemble of ELMs based on a resampling technique to deal with class imbalance. Each ELM is associated with a weight that is updated over time. ESOS-ELM uses G-mean, which is an appropriate performance metric for class imbalanced problems, to update the weights over time. The ensemble is reset upon a threshold-based concept drift detection. ESOS-ELM also maintains a memory of batch models. Old models can be recovered from the memory to deal with recurrent concepts.

Oversampling (OOB) and Undersampling (UOB) Online Bagging [76] are two online passive resampling-based ensemble approaches that can be trained with any type of online base classifier. They are extensions of the Online Bagging [60] algorithm, which uses each incoming training example to update each classifier k times. The value k is drawn from a $Poisson(1)$ distribution for each classifier independently. The use of the $Poisson(1)$ distribution enables Online Bagging to be a good approximation of Breiman's Bagging [7] algorithm. However, Bagging is not prepared to handle class imbalance. Instead of using $Poisson(1)$, OOB and UOB use $Poisson(\lambda)$, where λ is set as $\hat{p}_t^{maj}/\hat{p}_t^{min}$ for OOB and $\hat{p}_t^{min}/\hat{p}_t^{maj}$ for UOB, where $\hat{p}_t^{maj}$ and $\hat{p}_t^{min}$ are the estimated prior probabilities of the class considered to be a majority and a minority at time t, respectively. This will result in oversampling being applied to tackle class imbalance in OOB, and undersampling in UOB. The probabilities are estimated based on a time-decayed function. This enables OOB and UOB not only to deal with class imbalance, but also to deal with concept drifts affecting $p(y)$. These approaches can be generalized to multi-class settings [77] and combined with drift detectors to become active approaches that tackle concept drifts that affect $p(y|x)$ [78].

A few different concept drift detection methods have also been proposed for class imbalanced data streams. They monitor performance metrics suitable for class imbalanced problems and trigger when the metric significantly deteriorates. Drift Detection Method for Online Class Imbalance learning (DDM-OCI) monitors the time-decayed Recall on the minority class [75], Linear Four Rates (LFR) monitors a time-decayed confusion matrix, whereas Prequential Area Under the Curve Page Hinkley (PAUC-PH) monitors the area under the ROC curve.

Several chunk-based approaches have been proposed based on the storage of old minority class examples to help overcoming class imbalance [1, 27, 28, 82, 14, 15, 35, 45]. Some of these approaches train components by combining all minority class examples seen so far with majority examples from the most recent chunk [27, 28, 82, 45]. Other approaches filter out past minority class examples based on their similarity to the minority class examples of the current data chunk [14, 15], or based on their age [35]. Some other chunk-based approaches tackle class imbalance by using standard offline re-sampling techniques to create a new offline learning classifier for each new data chunk [17, 47]. Yet another recent proposal promotes a reinforcement mechanism to increase the weights of the base classifiers that perform better on the minority class and decrease the weights of the classifiers that perform worse [83].

Contrary to all the aforementioned algorithms which focus on global class imbalance, few data stream mining studies attempted (explicitly or implicitly) to tackle local difficulty factors. Some works have proposed approaches that attempt to deal with local data distributions, however, none of them took into account drifting difficulty factors. For example, Lyon *et al.* [48] discussed the problem of the minority class decompositions into smaller sub-clusters, and claimed that Hellinger Distance Trees may better classify such data than standard VFDT. The role of the class decomposition was also considered in a chunk-based ensemble SCUT-DS [59]. Moreover, Ren *et al.* [63] proposed a clustering-based oversampling technique to reduce the risk of increasing class overlapping. Finally, an attempt to handle several types of difficulty factors was recently presented in [39], where the difficulty of incoming examples is exploited in the online learning of a cost sensitive tree. As more difficult examples are presented more times during training, the trees shift towards concentrating on more difficult examples.

4 Proposed categorization of concept drifts for class imbalanced data

As discussed in Section 3.2, local data difficulty factors strongly influence the ability of classifiers to cope with static class imbalanced data. In particular, depending on the underlying distribution of the minority examples, standard classifiers with no strategies to deal with class imbalance can perform either quite well or very poorly. Therefore, it is likely that the extent to which the class imbalance and concept drift issues are exacerbated when combined together in non-stationary environments also depends on the underlying distribution of the minority class, and how it changes over time. The conjunction of local data difficulty factors and concept drift may be potentially more challenging for classifiers than the impact of each of these factors separately when dealing with non-stationary data streams.

However, existing literature on categorization and characterization of concept drifts in streaming classification problems does not take these local data difficulties into account. Recall that most research about imbalanced streams is concerned with the role of the global imbalance ratio only, either for static or changing class proportions. Given the major role that local data difficulty factors may play in class imbalanced data streams, existing concept drift categorizations may not be enough for evaluating class imbalance learning in non-stationary environments. They may not capture all aspects that make a concept drift particularly challenging or easy to solve. Therefore, we propose an *extended concept drift categorization* that takes

local data difficulty factors into account. This extended categorization should open up the path for further research on classifiers for class imbalanced data streams in non-stationary environments, enabling systematic studies under different relevant drifting conditions.

We will concentrate on extending the criteria for categorizing and characterizing single concept drifts. It is worth reiterating that any drift can be categorized and characterized using several different criteria at the same time. In particular, both existing criteria (e.g., those mentioned in Section 3.1) and our proposed criteria (Sections 4.1 and 4.2) can be used together to describe drifts. The proposed categorization is graphically depicted in Fig. 2.

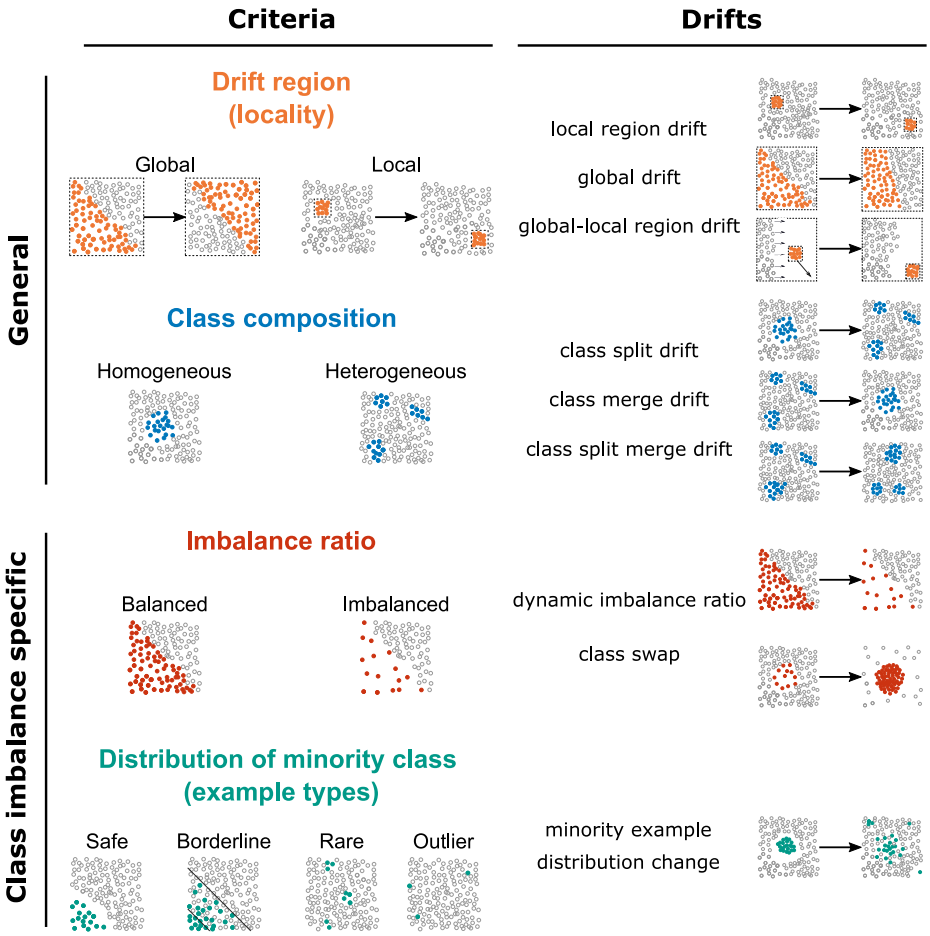


Fig. 2: Imbalanced stream drift categorization. Each criterion (left column) is accompanied by a list of possible drifts (right column). The minority class is depicted as filled (colored) circles, the majority class as hollow (white) circles.

4.1 Proposed drift criteria for general problems

4.1.1 Locality of drift region

We start the proposed categorization by distinguishing the locality of the data streams. For that, we need a definition of a "local region", which makes use of the definition of a "partitioning" shown below:

Definition 01 (Partitioning) *Consider a bounded d-dimensional attribute space $\mathcal{X}$ and output space $\mathcal{Y}$ and a given posterior probability distribution $p(y|x)$, where $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. A partitioning $\mathcal{P}$ of $\mathcal{X}$ is a set of bounded regions $\mathcal{R}_1, \cdots, \mathcal{R}_n$ such that (a) $\forall i \neq j : \mathcal{R}_i \cap \mathcal{R}_j = \emptyset$; (b) $\cup_{i=1}^{n} \mathcal{R}_i = \mathcal{X}$; (c) $\forall y \in \mathcal{Y},\ \forall x, x' \in \mathcal{R}_i : |p(y|x) - p(y|x')| < \tau$, where $\tau \in \mathbb{R}$ is a threshold; and (d) all examples that can possibly be sampled from a given region R_i are connected to each other through a d-dimensional grid, where the granularity of this grid is infinitely small for real attributes, and is discrete for discrete attributes.*

For example, in Fig. 2, the four plots under the heading "Drift region" are composed of two regions each, whereas the right side plot under the heading "Class composition" is composed of four regions. The criterion (d) ensures that a region is not formed by disconnected components. For instance, the three blue regions in the right side plot under the heading "Class composition" are three different regions, rather than a single region.

The definition above has been partly inspired by [66]'s definition of clustering. However, it differs in several aspects. First, our partitioning is a partitioning of the attribute space, whereas clustering is about clusters of examples sampled from a given unconditional probability distribution. Second, we include criterion (c), which is absent from the clustering definition. This criterion is important because we are dealing with classification problems, and so a given region must have the vast majority of its examples associated to a given class. And third, we consider a grid that binds together all examples that can be possibly sampled from the region. This grid means that a given region cannot be composed of several disconnected areas of the attribute space. Any neighbourhood operator can be used to connect points in a grid. For instance, for a 2-dimensional space, the classic Moore or Von Neumann neighbourhoods [73] could be adopted.

Definition 02 (Local Regions) *Given a partitioning $\mathcal{P}$, a region $\mathcal{R}_i \in \mathcal{P}$ is a local region if $V(\mathcal{R}_i) << V(\mathcal{X})$, where $V(\dot{})$ is the geometric volume.*

For example, in Fig. 2, the orange regions in the two plots under the heading "Drift region: Global" are not local regions, whereas the orange regions in the two plots under the heading "Drift region: Local" are local regions.

Even though existing literature [79, 38, 24, 50] has considered concept drifts that affect the probability distributions associated with the whole problem space, or to part of the problem space, it has not taken locality into account. Specifically, when the drifts affect only part of the problem space, existing studies have not considered whether these drifts affect a *local region* of the space or not. Such locality is a general criterion, but is likely to be particularly important for characterizing drifts in class imbalanced data streams. For instance, if we have drifts that affect small portions of the space associated with the minority class, this may be very difficult to track.

Based on the locality of the region affected by the drift, a concept drift can be categorized as *local region drift*, *global drift* or *global-local region drift* (Fig. 2). A local region drift is a concept drift that affects the joint probability distribution ($p(x, y)$) of one or more local regions. If the drift affects regions of the space that are not local regions, it is referred to as a global drift. An example of local region drift in a real world data stream can be found in Fig. 6 of Sun et al.'s work [71], whereas several synthetic data streams widely adopted by the data stream mining community (e.g., Hyperplane [70]) present global drifts. A global-local region drift affects both specific local regions and non-local regions of the input attribute space.

When a drift is a local region drift, other criteria for characterizing and categorizing drifts such as severity and rate of change can be used to either describe the changes suffered by each local region separately, or to describe the changes as a whole. For example, the severity of the change caused by a local drift to a given local region can be characterised by the volume of the intersection between the old and the new position of this local region. A smaller/larger intersection means higher/lower severity. And, according to the rate of change, a certain local region may be changing slowly, while others may be changing fast. Or, all regions may be changing fast. The combination of the locality criterion with the rate of change for a specific local region may be a particularly useful combination of criteria to describe drifts. For instance, a given local region may change incrementally in a way that this region "moves" through the attribute space. Local region drifts can also be further characterized by the number and size of the affected local regions.

4.1.2 Class composition

Class composition refers to how examples of each class are spread over the input attribute space. Specifically, it corresponds to checking whether the class examples belong to a homogeneous region or not. A *homogeneous* class composition means that examples of the class are concentrated in a single local region (see the earlier discussion on single minority concepts in Section 3.2.3). A *heterogeneous* class composition means that examples of the class are spread over multiple local regions. Note that here we refer to local regions that are sub-concepts larger than regions of rare minority examples (rare example types are considered in a subsequent criterion). Up until now, class composition has been mainly considered in the literature on static imbalanced data, where it referred to the split of the minority class into several rare sub-concepts [37, 69].

Such minority class decomposition was found to have a greater effect on classification performance than the global imbalance ratio [36, 37, 69, 81]. Given the impact that class (de-)composition can have on the difficulty of class imbalanced problems, it should also be considered in data streams. This was reflected in recent studies, which demonstrate that the detection of sub-concepts in streaming chunks can improve classifier performance [63]. Furthermore, class composition may also change over time in real world problems. For example, in trend topic data streams several distinct groups of opinions on the same topic may appear as time goes on [38]. Changes of example clusters have also been observed in unsupervised streams [66, 67]; see earlier discussion in Section 3.3.

We propose to use class composition as a general criterion to describe concept drifts. We refer to drifts that break down homogeneous classes into heterogeneous ones as *class split drifts*, drifts that merge several different local regions together

as *class merge drifts*, and drifts that both break down and merge local regions as *class split-merge drifts* (Fig. 2). Such drifts can be characterized by a list of local regions with their corresponding $p(y|x)$ before and after the drift. Evidence of class merge drifts in real world problems can be found in [71], suggesting that class split type of drifts may also occur in real world problems.

4.2 Proposed drift criteria for class imbalanced problems

4.2.1 Imbalance ratio

This criterion corresponds to the prior probabilities of classes, representing the proportion of examples expected to be received from each class. Previous studies [78, 76] have taken this criterion into account when designing experiments to evaluate learning approaches for class imbalanced data streams. However, this criterion has not been explicitly emphasized as part of a concept drift taxonomy before.

We consider two situations: *static imbalance ratio* if the class proportions do not change over time, and *dynamic imbalance ratio* if the imbalance ratio changes. For example, it is known that the imbalance ratio in problems such as software defect prediction [13] and tweet topic classification [71] can vary over time, requiring specific learning algorithms able to cope with such variations. It is even possible for the role of the minority and majority classes to swap [13, 71], i.e., for the majority to become a minority and vice-versa (Fig. 2).

The above-mentioned drifts can be characterized by a vector containing prior probabilities of each class before and after the drift. Such a description can be further generalized for the case of multi-class classification.

4.2.2 Distribution of minority class examples with respect to their types

As discussed in Section 3, the distribution of different *types of minority class examples* can influence classifiers learning from imbalanced data. Following the method from [55, 56] the type of the minority example can be identified based on the analysis of class labels of other examples in its local neighbourhood. Depending on the number of examples from the opposite class, minority examples are labeled as safe, borderline, rare or outlying [56], as explained in Section 3.2.4. In case of streams, this analysis could be done with neighbours inside a sliding windows.

Although types of examples have been used to improve classifiers for static data, they were rarely considered in data streams [39, 42]. However, it has been shown that the number of minority class examples of particular types may vary over time in real-world problems, such as the analysis of tweet streams [42]; more examples of challenging and drifting class distributions in real-world streams will be presented in Section 5.

Following the above discussion, streams can be categorized as ones with static or dynamically changing example type distributions.

Minority class example type distribution drifts can be characterized through a vector containing the proportion of the minority examples belonging to safe, borderline, rare and outlier examples before and after the concept drift. It is worth noting that these drifts cover changes of single example types, such as increasing

the proportion of borderline examples at the cost of safe examples, as well as changes of many types at once (Fig. 2).

5 Experimental evaluation of the influence of data factors and drifts on online classifiers

5.1 Experimental Aims and Setup

We experimentally study the impact of the discussed data difficulty factors and drifts on the performance of selected online stream classifiers. Compared to earlier studies on imbalanced streams, where only global factors like changes of the imbalance ratio were considered, here we pay more attention to local data characteristics and local drifts.

Our series of experiments is organized along the following research questions, which have been preliminary introduced in Section 1:

RQ1 What is the impact of different types of single data difficulty factors and isolated, single drifts on the predictive performance of selected online classifiers? Is it possible to identify which classifiers work better and when?

RQ2 What is the interaction between the different types of local data factors and global class imbalance? Which local factors or drifts are the most demanding for classifiers at different imbalance ratios?

RQ3 Which complex scenarios integrating several data factors and drifts are the most harmful for classification performance? Is it possible to determine which components (single factors or drifts) in these scenarios are the most influential?

In order to examine these issues, we carried out most of the experiments in a controlled framework based on synthetic generated data. Using a synthetic data stream generator, each data factor can be modeled and parametrized according to different planned scenarios. This is important because it allows us to obtain a detailed understanding of when and under what circumstances classifiers work well or fail. Moreover, we also supplement this study with experiments on real-world imbalanced streams, where most of the considered data difficulty factors of drifts were identified.

For the purposes of the synthetic data experiments, we have implemented an imbalanced data stream generator. A detailed description of the generator is given in Section A of the supplementary materials.[1] The source code of a MOA [2] compatible implementation of the generator is available at: https://github.com/dabrze/imbalanced-stream-generator.

The synthetic data stream characteristics are controlled by modifying parameters referring to the criteria proposed in this paper. The list of parameter values (stream elements) used to create streams for this study is listed in Table 1. The generated streams where named according to a naming convention of their elements (Split, Move, Merge, Borderline, Rare, Im, StaticIm), introduced in Section B of the supplement.

[1] Supplementary materials at: https://doi.org/10.6084/m9.figshare.12098127.v2

Table 1: Elements with their parameters used to generate streams

RQ	Stream parameter (element)	Used parameter values
Streams with single factors or drifts		
RQ1	Class composition (CD)	{Split[N], Move[N], Merge[N]: for N ∈ {3, 5, 7}}
	Example types (TD)	{Borderline[N], Rare[N]: for N ∈ {20, 40, 60, 80, 100}}
	Imbalance drift; minority ratio after drift (ID)	{Im[N]: for N ∈ {50%, 40%, 30%, 20%, 10%, 5%, 3%, 2%, 1%}}
	Static imbalance; minority ratio throughout the entire stream (SI)	{StaticIm[N]: for N ∈ {50%, 40%, 30%, 20%, 10%, 5%, 3%, 2%, 1%}}
Combined stream scenarios		
RQ2, RQ3	Class composition (CD')	{Split[N]: for N ∈ {1, 5}}
	Example types (TD')	{Borderline[N], Rare[N], Borderline[M]+Rare[M], for N ∈ {20, 40, 60, 80, 100} and M ∈ {20, 40}}
	Imbalance drift (ID')	{Im[N]: for N ∈ {50%, 10%, 1%}}
	Static imbalance (SI')	{StaticIm[N]: for N ∈ {50%, 10%, 1%}}

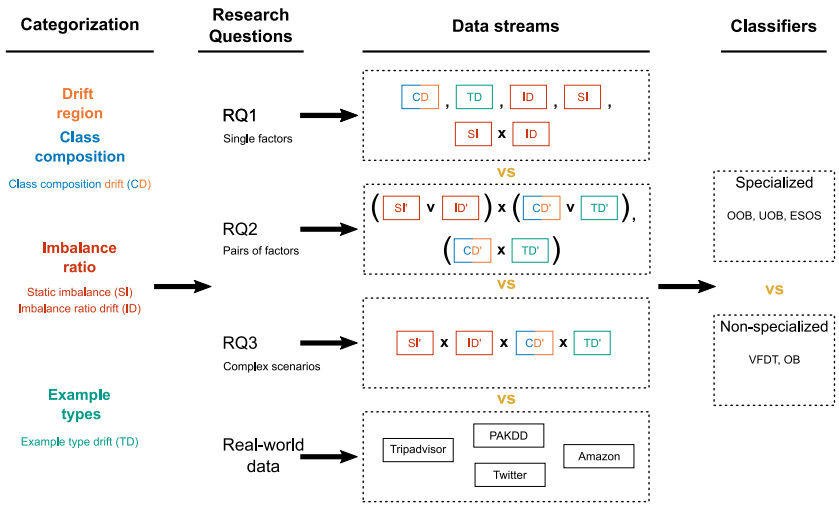


Fig. 3: Schematic of the experimental setup.

To answer research questions RQ1–RQ3, a comprehensive set of data streams was generated as follows:

1. Streams with single factors or drifts (RQ1, discussed in Section 5.2)
   – First we modeled the occurrence of a single element (data factor or drift) in isolation from other data stream aspects. All single values of elements

from lists CD, TD, ID or SI were used to generate data streams. If drifts were modeled (CD, TD, ID), then the stream starts as fully balanced, with only safe examples, and one cluster of 'minority' examples surrounded by examples from the 'majority' class.
   - Moreover, we considered imbalance ratio changes, where we started from a static imbalanced stream and generated streams corresponding to all combinations of ratio values from the product SI × ID.
2. Streams with pairs of elements (RQ2, Section 5.3)
   - Streams with global imbalance were generated as pairs of elements from (SI' or ID') x (TD' or CD')
   - Combinations of values from the product TD' × CD'
3. Complex scenarios consisting of triples or more elements (RQ3, Section 5.4)
   - The set of streams was created as a Cartesian product: SI' × ID' × CD' × TD', excluding streams with mutually exclusive characteristics such as StaticIm1 and Im1, and streams where the global share of minority class was increasing with time.

The diagram summarizing these generation scenarios, their relations to research factors and corresponding categories is presented in Fig. 3.

The above described scenarios resulted into generating 385 synthetic streams (224 static and 161 drifting). For their full list, see the online supplementary materials. All the generated data streams consisted of 200,000 examples. If a concept drift is present in a stream, it spans from example number 70,000 to 100,000. In case of combined drift, they occur all together in this time period. If the imbalance ratio is not directly stated, the stream is balanced. If no minority class composition or examples type details are given, the minority concept is defined as a singular cluster of safe-type examples surrounded by majority class examples uniformly distributed in the attribute space.

In addition to synthetic streams, four real-world datasets [5, 72, 74, 53] with various imbalance ratios and local data difficulty factors were used. The real streams range from 8,000 to 50,000 examples and were evaluated as difficult for learning classifiers in previous studies [42, 9]. Reproducible experimental scripts are available at: `https://github.com/dabrze/imbalanced-stream-generator`.

The experiments compare the predictive performance of five online classifiers quite often used in the related literature: Oversampling Online Bagging (OOB) [76], Undersampling Online Bagging (UOB) [76], ESOS–ELM (ESOS) [52], Online Bagging (OB) [61], and Hoeffding Tree (VFDT) [19]. OOB, UOB and ESOS were selected as representative specialized classifiers for imbalanced streams which use different strategies and are based on different methods. OB represents a non-specialized data stream ensemble exploiting online bagging (which was an inspiration for OOB and UOB). VFDT serves as a reference for single classifiers.

All ensembles used 15 Hoeffding Trees as component classifiers. For OOB, UOB, OB and VFDT we used MOA implementations with default parameter values suggested by the algorithms' authors. Note that we used the standard version of VFDT classifier[2] without any drift-related mechanisms. ESOS was implemented for this study and, as proposed by the algorithm's authors, utilized artificial neural networks with 70 hidden neurons. The size of the initialization batch for the OS-ELM classifiers in ESOS was set to 100 examples with an evaluation period

[2] Trees were induced with the *moa.classifiers.trees.HoeffdingTree* class from MOA

of 1000 examples, and the drift detection threshold coefficient set to 0.9, as recommended in [52]. With the exception of ESOS-ELM, which by design employs a basic performance-tracking drift detector, the other considered classifiers are not integrated with any drift detectors as the ones provided by MOA are not suitable for dealing with class imbalance and are thus outside the scope of our experimental study.

The classifiers were evaluated using two performance measures — *Recall* and *G-mean*. Recall (also called Sensitivity or True Positive Rate – TPR) is the correct recognition rate for the minority class. G-mean is the geometric mean of recall (TPR) and specificity (True Negative Rate – TNR), defined as $G_{mean} = \sqrt{TPR \cdot TNR}$. Recall and G-mean were selected from a larger list of measures [11, 34] mainly due to their complementary nature and easy interpretation. Recall focuses only on the minority class, allowing us to see when the recognition of the minority class drops. In contrast, G-mean captures the balance between recognition ratios of both classes. Therefore, by analyzing both measures it can be noticed whether one class was recognized more often at the cost of the other. For example, if Recall improves but G-mean deteriorates, this means that the recognition of the minority class has improved at the cost of the recognition rate of the majority class. Moreover, G-mean is skew-invariant, meaning that G-mean's interpretation remains the same for all possible class imbalance ratios [11, 12], being particularly relevant for studying drifting imbalance ratios.

Both Recall and G-mean were calculated prequentially [25] using a sliding window of 1000 examples. Each analyzed stream was visualized as a line plot with the number of processed examples on the x-axis, value of Recall/G-mean on the y-axis, and the drift region plotted over a gray background. To increase readability, the line plots were smoothed using a moving average of 20 data points.

Moreover, measure values averaged over entire streams (mean performance values) are presented in a tabular form, which constituted the base for carrying out a ranked Friedman test [16] to compare classifier performance. Due to the large number of experiments, here we only present the most representative plots; the reader is referred to the online supplementary material and code repository for additional results.

5.2 Experiments with single drifts or data difficulty factors

5.2.1 Static imbalance between classes

The fully balanced stationary stream without any difficulty factors was quite easy to learn, with most classifiers achieving, on average, 0.99 G-mean and Recall (for detailed values see Supplementary Tables S3 and S4). Average classifier performance on stationary imbalanced streams with minority class ratios 10%, 20%, 30% and 40% were nearly the same (Fig. 4). For all the aforementioned minority class ratios, classifier performance plots looked very similar — performance values rise fairly quickly up to a certain level and remain stable until the end of the stream.

The situation changes for higher class imbalance (i.e., minority class ratio $\leq$ 5%). With 3% or 5% of minority class examples, plots of OOB, UOB, and ESOS achieved similar G-mean values to those obtained for higher minority class ratios

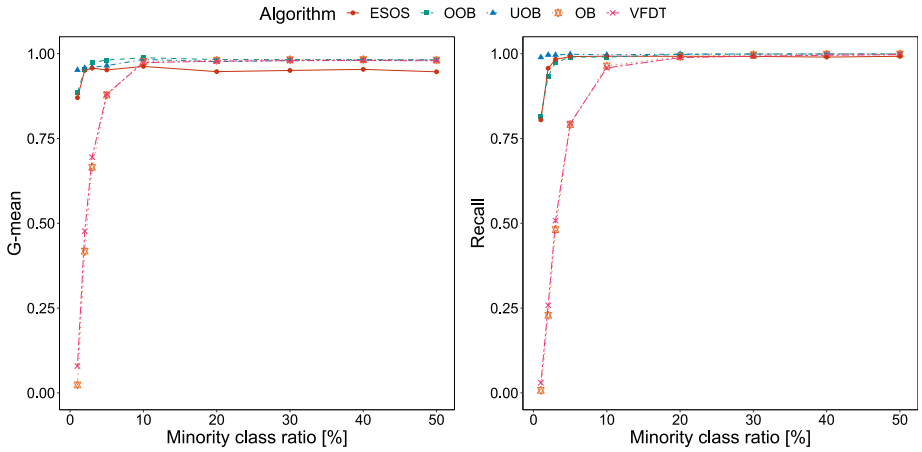


Fig. 4: Comparison of the effect of static imbalance (the same minority class ratio over the entire stream) on average classifier performance; G-mean and Recall averaged over entire streams for a given classifier on a given stream.

(i.e. the aforementioned less imbalanced streams). However, OB's and VFDT's G-mean was worse (at most 0.830–0.850). An even bigger classifier distinction can be seen for the two highest tested class imbalance ratios (1% and 2% minority examples in the stream). Here, the non-specialized OB and VFDT classifiers performed dramatically worse — plots of their G-mean dropped down to ∼0.100, whereas minority class Recall was as low as 0.020 for OB. The remaining (specialized) classifiers still achieved fairly high G-mean values when the imbalance ratio was 2% (0.947, 0.956, 0.876 for OOB, UOB, and ESOS respectively). However, for the imbalance ratio of 1%, the recognition of the minority class (Recall) was worse even for the specialized classifiers (e.g., OOB's average Recall was 0.815). Figure 4 shows how average G-mean and Recall averaged over entire streams changed between data streams with different static imbalance ratios.

The statistical comparison of classifier performance on all static imbalanced streams, using G-mean and Recall averaged over entire streams, was carried out with the Friedman test followed by the Nemenyi post-hoc test (Table 2). The test resulted in the following rankings: OOB $\succ$ UOB $\succ$ OB $\succ$ VFDT $\succ$ ESOS (G-mean) and UOB $\succ$ OOB $\succ$ OB $\succ$ ESOS $\succ$ VFDT (Recall). Importantly, OOB is significantly better than all the remaining classifiers except UOB on G-mean, whereas UOB is significantly better than all but OOB.

5.2.2 Drifts of imbalance ratios

Next, we analyzed streams that were initially balanced and then the single global imbalance ratio drift was modeled. If the minority class ratio after the drift remained $\geq$ 10%, then all the analyzed classifiers achieved similar results to those obtained for stationary imbalanced streams discussed in the previous section. For lower minority class ratios, OB and VFDT performance dropped to slightly lower values (in particular for 1% and 2% minority ratios).

Table 2: Mean Friedman test ranks for comparing classifier performance using G-mean and Recall averaged over entire streams (see them in Supplementary Tables S3 and S4) on different sets of data streams involving single data difficulty factors. All tests were significant with p-values < 0.0001. Best value (lowest rank) and values which were found to be not statistically different from the best value according to the Nemenyi post-hoc test are highlighted in bold. Column *CD* shows the critical distance of the Nemenyi post-hoc test at significance level $\alpha = 0.05$.

Data stream set	Metric	OOB	UOB	OB	VFDT	ESOS	*CD*
static imbalance	G-mean	**1.18**	**2.35**	3.00	3.94	4.53	1.51
class ratio changes		**1.17**	2.25	3.59	3.95	4.03	0.77
sub-cluster merge		**1,33**	**3,17**	**1,67**	**3,83**	5,00	2,68
sub-cluster move		**1,00**	**2,83**	**2,17**	4,00	5,00	2,68
sub-cluster split		**1,33**	**3,00**	**1,67**	**4,00**	5,00	2,68
borderline examples		**1,70**	**2,80**	**1,50**	4,00	5,00	2,01
rare examples		**1,80**	3,90	**2,40**	**2,30**	4,60	2,01
static imbalance	Recall	**2.29**	**1.12**	3.59	4.35	3.65	1.51
class ratio changes		2.30	**1.06**	4.13	4.23	3.28	0.77
sub-cluster merge		**2,17**	**2,17**	**2,17**	**4,17**	**4,33**	2,68
sub-cluster move		**1,17**	**3,00**	**2,00**	4,17	4,67	2,68
sub-cluster split		**1,33**	**3,00**	**1,67**	4,17	4,83	2,68
borderline examples		**2,30**	**3,20**	**2,40**	4,90	**2,20**	2,01
rare examples		3,80	3,90	3,70	**2,00**	**1,60**	2,01

On the other hand, scenarios in which the stream initially had $\leq 10\%$ minority examples which then went down to 1–2%, were more demanding. In such scenarios, UOB, OOB and even ESOS still perform quite well, whereas VFDT and OB performed poorly after the drift. Figure 5 shows such a case with the minority class ratio changing from 10% to 1%. Moreover, Fig. 6 presents the performance of classifiers (x-axis) on different minority ratio drifts (y-axis).

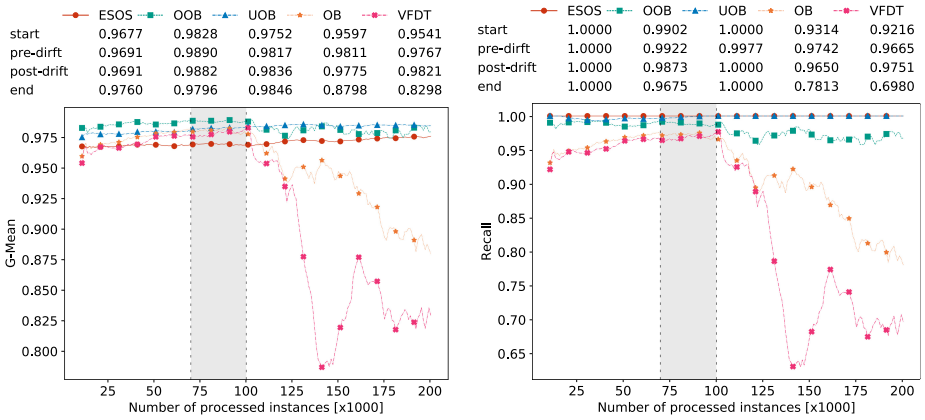


Fig. 5: G-mean (left) and Recall (right) of the analyzed classifiers on stream StatIm10+Im1, where there is a class ratio drift changing the minority class ratio from 10% to 1%.

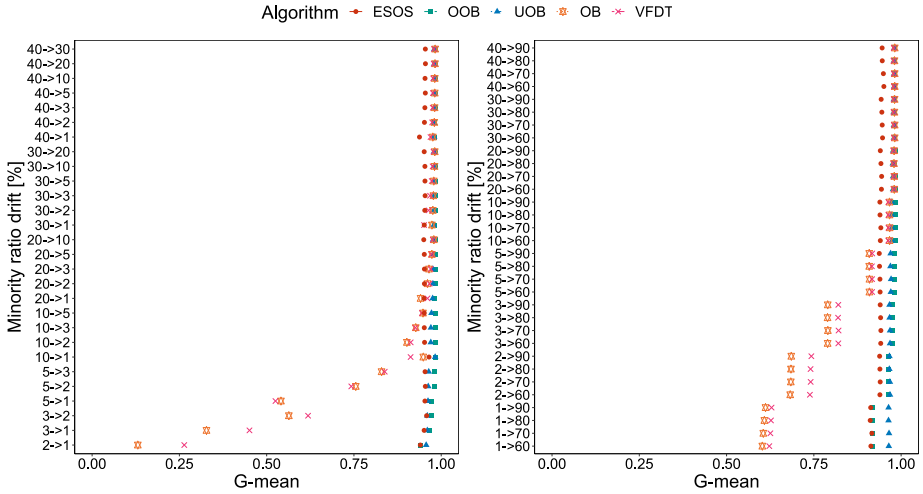


Fig. 6: Comparison of the effect of decreasing minority class ratios (left) and class swaps (right) on average classifier performance; G-mean averaged over entire streams for a given classifier on a given stream.

Interestingly, scenarios which involved class swapping (e.g. 1% or 2% $\rightarrow$ 60%) were less demanding. Specialized classifiers consistently and fairly quickly improved their performance from the beginning of the stream, whereas OB and VFDT started with close-to-zero G-mean and Recall values only to improve up to similar values as OOB or UOB after the drift.

Looking at Fig. 6, one can notice that streams with the minority class ratio over 5% are sufficiently well recognized by all classifiers. It is also clear that non-specialized classifiers (OB and VFDT) were much more susceptible to global class imbalance than specialized learners (OOB, UOB, ESOS). Finally, it seems that the difficulty of imbalance ratio drifts depends on the number of minority examples available before the drift. If a classifier had enough examples to learn the minority concept prior to the drift, the effect of the drift was smaller.

The observed relations were confirmed by the statistical analysis of the averaged values of both measures done with the Friedman and Nemenyi tests at significance level $\alpha = 0.05$. The classifier rankings (Table 2) were the same as for the stationary imbalanced streams, but the difference between average ranks of the best performing OOB/UOB and the remaining classifiers was much higher.

5.2.3 Changing class composition

In this group of experiments, we considered three balanced scenarios with changes in class composition: 1) stationary streams with a class decomposed into 3, 5 or 7 sub-clusters, 2) a drift of sub-clusters gradually moving in the attribute space to new random positions[3], 3) a drift splitting splitting one concept into multiple

[3] The new random positions were generated in such a way to ensure that they would not result in the sub-clusters overlapping with each other – see details in the description of the generator in the supplement.

sub-clusters, and 4) several sub-clusters merging into one concept. Note that, even though such class compositions have been discussed in the context of class imbalanced data streams in Section 4, they are also applicable to a given balanced class as done in the current section. Such balanced scenarios constitute baseline cases – a corresponding classifier that is successful in dealing with class imbalance would present predictive performance as good as the ones obtained in these scenarios when faced with corresponding imbalanced scenarios. In Section 5.3, we will discuss these data difficulty factors in imbalanced data streams.

Stationary streams with a class decomposed into sub-clusters were difficult for the analyzed classifiers. All classifiers incrementally improved G-mean (or Recall) to rather high values ∼0.982, except for ESOS which reached a slightly smaller G-mean of 0.943. However, to achieve this level of predictive performance, the classifiers required more examples than in the scenarios with the global imbalance ratio from Section 5.2.2. Moreover, the more sub-clusters the worse the observed classifier performance.

Scenarios involving a single drift of sub-clusters gradually moving in the attribute space were more demanding and the decrease of the performance measures was clearly visible. However, although G-mean initially dropped from 0.975 down to 0.940, the classifiers recovered quite well after the drift. The effects of moving sub-clusters were only slightly stronger when the number sub-clusters was larger.

The impact of splitting one cluster into 3, 5, or 7 sub-clusters was similar to cluster movement. For example, splitting a class into 5 sub-clusters resulted in a decrease of G-mean of OOB/UOB from 0.989 before the drift to 0.919/0.901 after the drift, and from 0.963 down to only 0.750 respectively for ESOS (Fig. 7). It is worth noting that all the analyzed classifiers only partly recovered from this type of drift.

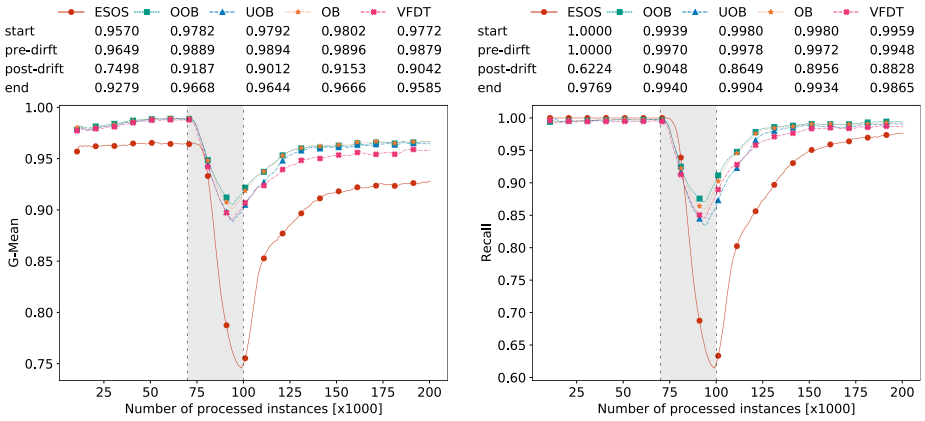


Fig. 7: G-mean (left) and Recall (right) of the analyzed classifiers on stream Split5, in which the minority class is gradually split into 5 sub-clusters.

Finally, a drift merging many sub-clusters did not pose a significant challenge for the analyzed classifiers. The plots showed that classifiers started off with slightly lower performance values, followed by a small decrease in classifier per-

formance, but then recovered quite quickly and achieved values of G-mean and Recall comparable to those achieved on stationary streams.

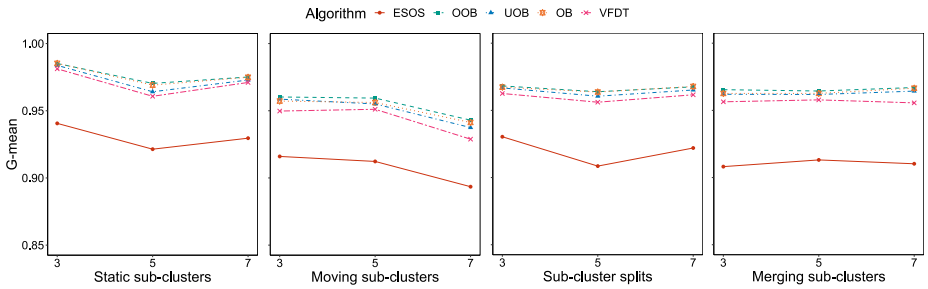


Fig. 8: Comparison of the effect of class composition drifts on average classifier performance in balanced data streams; G-mean averaged over entire streams for a given classifier on a given stream.

Figure 8 compares the effect of different class composition drifts on G-mean averaged over entire streams (for detailed values see the Section C of the supplement). It can be noticed that all the class composition drifts have similar, rather low, impacts on G-mean since the performance drops were only temporary.

The rankings of classifiers according to the Friedman and Nemenyi tests performed over averaged values were slightly different compared to previous scenarios (Table 2). In particular, UOB was better than OOB on Recall, and also partially on G-mean, and ESOS performed generally better than OB and VFDT.

5.2.4 Distribution of example types

Similar to Section 5.2.3, in this group of experiments, we analyzed balanced stationary streams with a given percentage of either borderline or rare types of examples for a given class. Outlier examples were discarded from our analysis as they demonstrated to be extremely difficult for static data and rare cases were already very difficult for the considered stream classifiers.

Classifier performance plots showed that the presence of both types of examples always decreases G-mean and Recall. However, the effect depends on the type and proportion of difficult examples (Table 3). Increasing the proportion of borderline examples reduces G-mean values in comparison to a stream with only safe minority examples, but the deterioration is relatively small (e.g. for OOB G-mean drops from 0.982 (0% borderline) to 0.963 (100% borderline)). On the other hand, the proportion of rare examples showed to be clearly more influential, and even proportions of rare examples as small as 20% lead to G-mean decreasing to $\sim$0.88 for all classifiers, whereas streams with 100% of rare examples exacerbate the effect leading to G-means $\sim$0.55. Similar trends were observed for Recall.

Similar trends were observed when the difficult examples were not present from the start of the stream, but appeared as part of a concept drift. To show that the classifiers recovered from such drifts, in Table 4 we present the values of G-mean directly after the drift (post) and at the end of the stream (end). For the drifts of

Table 3: The impact of borderline and rare type minority examples in balanced stationary streams on G-mean values of classifiers; G-mean averaged over entire streams for a given classifier on a given stream.

Configuration	N	OOB	UOB	OB	VFDT	ESOS
Safe stream	0%	0.982	0.981	0.981	0.979	0.947
Borderline[N] static	20%	0.973	0.973	0.973	0.971	0.953
	40%	0.969	0.968	0.969	0.966	0.949
	60%	0.967	0.966	0.967	0.965	0.944
	80%	0.964	0.964	0.965	0.962	0.940
	100%	0.963	0.963	0.963	0.961	0.938
Rare[N] static	20%	0.885	0.883	0.884	0.883	0.852
	40%	0.766	0.765	0.766	0.766	0.737
	60%	0.633	0.632	0.633	0.643	0.620
	80%	0.535	0.519	0.530	0.546	0.529
	100%	0.566	0.554	0.563	0.530	0.517

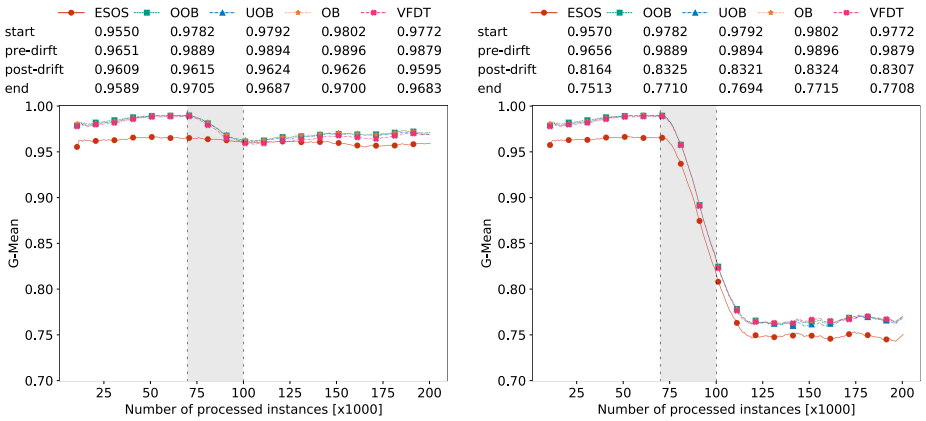
Table 4: The impact of type of examples on G-mean of classifiers. The → symbol shows the values of G-mean after the drift (post) and at the end of the stream ([post]→[end]).

Type	N	OOB	UOB	OB	VFDT	ESOS
Before drift	0%	0.989	0.989	0.990	0.988	0.965
Bordeline[N] drift	20%	0.972→0.975	0.972→0.974	0.972→0.975	0.970→0.972	0.962→0.961
	40%	0.962→0.971	0.962→0.969	0.963→0.970	0.960→0.968	0.961→0.959
	60%	0.955→0.969	0.955→0.968	0.955→0.968	0.948→0.967	0.959→0.956
	80%	0.954→0.967	0.955→0.966	0.954→0.966	0.945→0.964	0.955→0.951
	100%	0.959→0.967	0.956→0.966	0.956→0.967	0.950→0.962	0.955→0.948
Rare[N] drift	20%	0.919→0.886	0.919→0.886	0.919→0.886	0.918→0.884	0.901→0.861
	40%	0.833→0.771	0.832→0.769	0.832→0.772	0.832→0.772	0.816→0.751
	60%	0.707→0.638	0.707→0.634	0.707→0.635	0.709→0.651	0.697→0.628
	80%	0.530→0.521	0.527→0.491	0.528→0.509	0.546→0.539	0.531→0.530
	100%	0.496→0.558	0.459→0.547	0.466→0.559	0.504→0.529	0.516→0.527

borderline examples, one can notice that G-mean decreases slightly more than in analogous static streams and that all classifiers slightly recovered after the drift. In contrast, drifts which introduced rare examples were definitely more influential and the classifiers did not recover after the drift.

Figure 9 compares the G-mean performance of classifiers on a stream with an increasing number of borderline (left) and rare (right) examples from 0% to 40%. Introducing borderline examples had visible yet limited effect on all the analyzed algorithms' performance, with classifiers such as UOB and OOB reacting slightly better to these drifts compared to other methods. However, the proportion of rare examples showed to be clearly more influential than the number of borderline examples. In this case we can notice much stronger deterioration of performance, with nearly no recovery. Similar observations were made for Recall.

We have also analysed the values of G-mean and Recall averaged over entire streams and grouped in sub-categories corresponding to different types of examples (for detailed values see Supplementary Tables S3 and S4). The results of Friedman and Nemenyi tests (presented in Table 2) confirm our observations, ranking OOB/UOB highest in terms of G-mean. Interestingly, UOB performed slightly better on borderline scenarios, whereas OOB was marginally better on data streams



(a) Increasing number of border examples (b) Increasing number of rare examples

Fig. 9: Classifier G-mean on streams (a) Borderline40 and (b) Rare40, which respectively have ratios of borderline and rare examples within the minority class increasing from 0% to 40%.

with rare examples. In both cases, however, the differences between OOB and UOB were not statistically significant. In terms of Recall, ESOS achieved highest Friedman ranks on both rare and borderline data streams, followed by UOB. It is also worth noting that, on data streams with rare examples, all classifiers had fairly similar ranks and were mostly not significantly different from each other according to the Nemenyi test, which might suggest they all had similar problems with classifying these streams.

Finally, Fig 10 shows the comparison of the impact of different example type distributions with respect to G-mean averaged over entire streams. This comparison confirms that the proportion of rare examples has the biggest impact on classifier performance. It can also be noticed that static difficulty factors are slightly more demanding than drifting ones, since their effect is not temporary.

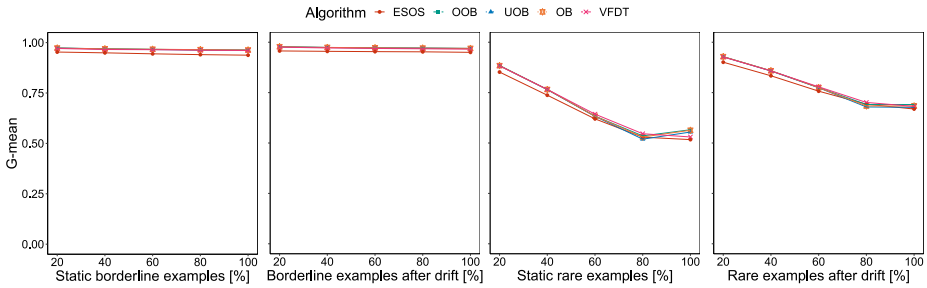


Fig. 10: Comparison of the effect of minority example type distribution on average classifier performance; G-mean averaged over entire streams for a given classifier on a given stream.

RQ1: There are considerable differences in the impact of single difficulty factors and drifts. The presence of rare examples demonstrated the highest deterioration of classifier performance, both in static and drifting streams. The effect of static and dynamic global imbalance was also influential, but depends on the imbalance ratio: 1–2% minority class ratio strongly affected the classifiers, whereas 5–40% had almost no effect. The decomposition of the a given class, either by splitting/merging sub-clusters or moving them, temporarily decreases the performance of all classifiers. The impact of borderline examples in a given class is slightly less important than local drifts in the composition of a given class. The reactions of the studied classifiers to single drifts vary. Specialized classifiers (OOB, UOB, ESOS) coped well with static class imbalance, especially OOB and UOB, but were unable to learn rare examples or fully recover from changes in class composition. Non-specialized classifiers (OB, VDFT) did not cope well with high class imbalance and local drifts.

5.3 Impact of pairs of elements in the stream

To study the impact of different types of data difficulties and class imbalance (RQ2), we have analyzed pairs of elements in the generated streams. We have studied various difficulty factors paired with moderate and high class imbalance ratios, and considered combinations of factors referring to the minority class split and the distributions of minority example types.

5.3.1 Interactions between local data factors and the global class imbalance ratio

We have examined how different global imbalance ratios interact with the most influential drifts identified in the previous subsection. As it was explained in subsection 5.1, two representative imbalance ratios were chosen 1% (as the most influential one) and 10% (having a moderate impact). Firstly, these static global imbalance ratios were combined with a drift on either the class composition or distributions of the minority example types.

The general observation from analyzing the plots of classifier performance is that class imbalance amplifies the effect of other difficulty factors. This amplification is especially visible on streams involving high static imbalance.

For instance, the combination of a static 1% minority class ratio with a minority class split into 5 sub-clusters (StatIm1+Split5 stream, Fig. 11) resulted in three widely different post-drift G-mean values for the specialized classifiers — UOB: 0.519, OOB: 0.271, ESOS: 0.741. In comparison, in the scenario where the split was performed on a balanced stream these classifiers achieved G-mean of ∼0.900 (Fig. 7). Note that there is worse recovery after the stronger drifts and even no recovery for OB and VFDT. Similar trends were observed for pairs involving high static imbalance ratios combined with moving sub-cluster as well as rare and borderline types of minority examples.

In contrast, pairs involving combinations of drifting class imbalance ratios, i.e. streams that started off as balanced and class imbalanced appeared in conjunction with a drift, were slightly less demanding. For instance, let us consider UOB in the stream Im1+Split5. Its G-mean values changed as follows: 0.988 (the moment

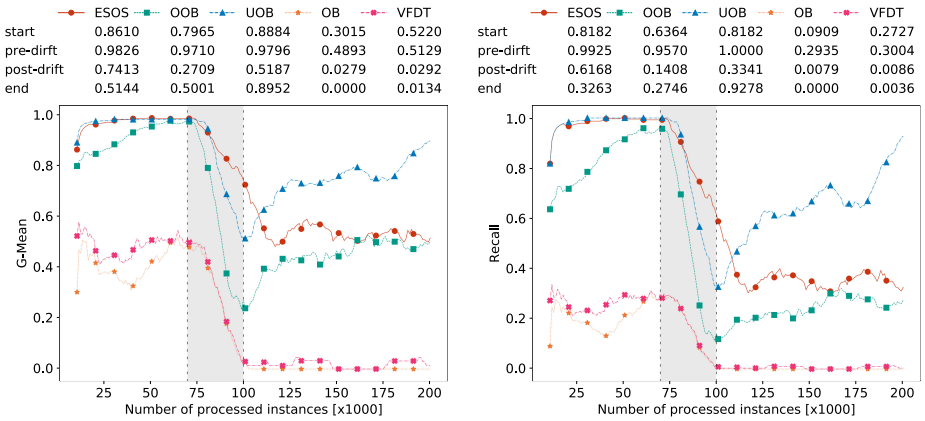


Fig. 11: G-mean (left) and Recall (right) of the analyzed classifiers on stream StatIm1+Split5 stream, which has a 1% minority class ratio and a minority class split into 5 sub-clusters.

before the drift) → 0.892 (after the drift) → 0.919 (at the end of the stream). For comparison, G-mean values for UOB in StatIm1+Split5 were: 0.978 → 0.519 → 0.895. Similar differences between the effect of static and dynamic imbalance were observed for OOB. More results showing G-mean values in characteristic moments in the streams are presented in Section D of the supplement.

The classifier performance plots for pairs with the imbalance drifts were more similar to those obtained for scenarios containing only single difficulty factors. We hypothesize that, compared to static imbalance, in scenarios involving class ratio drifts the classifiers were capable of learning much more before the drift (on the balanced portion of the stream). The main observation is that having a constantly low proportion of minority class examples is much more challenging than having a varying imbalance ratio with periods of relative class balance.

5.3.2 Interactions of class split with various types of examples

In this subset of experiments, drifts of a selected class split (5 sub-clusters) with drifts of example type proportions were considered in balanced data streams. The use of balanced streams here offers a baseline for further analyses on more complex scenarios including multiple drifts and data difficulty factors presented in Section 5.4.

The general observation is that the most difficult scenarios combine the minority class split with rare examples. These scenarios always led to stronger deterioration than pairs with the same percentage of borderline examples. For instance, let us consider OOB and its G-mean values after the drift and at the end of the recovery. In Split5+Rare20 the values in these moments are 0.853 and 0.861 while in Split4+Borderline20 they are 0.897 and 0.927, respectively. Moreover, the larger the proportion of rare examples after the drift, the larger the performance drop. The classifier plot for Split5+Rare40 (Fig. 12) shows that all algorithms behave in a similar way and are unable to recover after the drift. Moreover, when compared against plots depicting scenarios containing only the split (Fig. 7) or only rare

examples (Fig. 9b), it can be noticed that it is probably the proportion of rare examples that has a bigger influence on the final performance of the classifiers. Similarly, it was observed that combinations of minority class split and borderline examples were more difficult than scenarios containing the respective single difficulty factors.

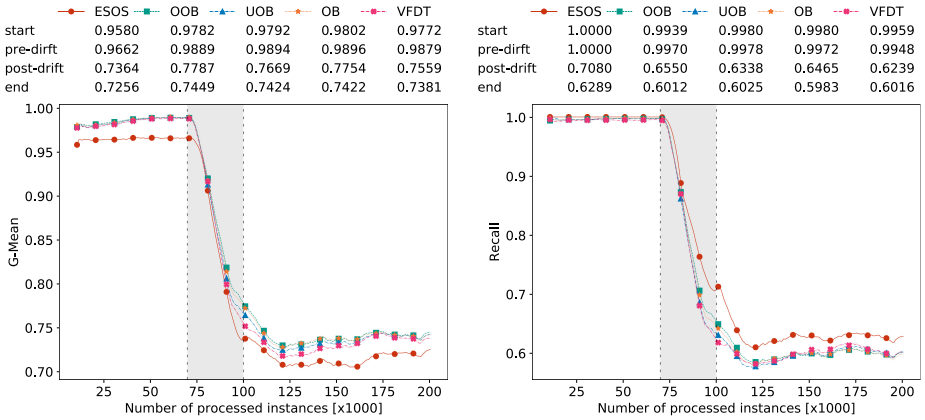


Fig. 12: G-mean (left) and Recall (right) of the analyzed classifiers on balanced stream Split5+Rare40, which has a drift introducing 40% rare type examples and a minority class split into 5 sub-clusters.

5.3.3 The global comparison of pairs of difficulty factors

Similarly to the analysis made in Section 5.2, we performed a global comparison of the impact of pairs of difficulty factors on classifier G-mean and Recall calculated over entire streams. The most influential pairs are presented in Fig. 13.

Definitely combinations with rare examples lead to the largest decrease of both measures for all classifiers (although OB and VFDT are the most affected). One explanation is that high rarity corresponds to a very scattered distribution of small islands of few minority examples. Together with the high imbalance it makes the data less safe and difficult to learn. On the other hand, combinations of rare examples with minority class splits were modeled with balanced streams, therefore the impact of the same percentage of rare examples was visibly smaller. As noticed in the earlier subsections, combinations of class imbalance drift with borderline examples are less influential. For some of these borderline pairs even OB and VFDT performed on par with OOB, UOB or ESOS.

Finally the Friedman test over stream-averaged values (Table 5) was performed to compare the classifiers. The test indicates that UOB should be the preferred classifier for streams involving combinations of high imbalance and an additional difficulty factor. However, it should be noted that the difference between UOB and OOB in terms of G-mean is not statistically significant. Interestingly, this statistical analysis also shows that, contrary to most previous analyses, in case of

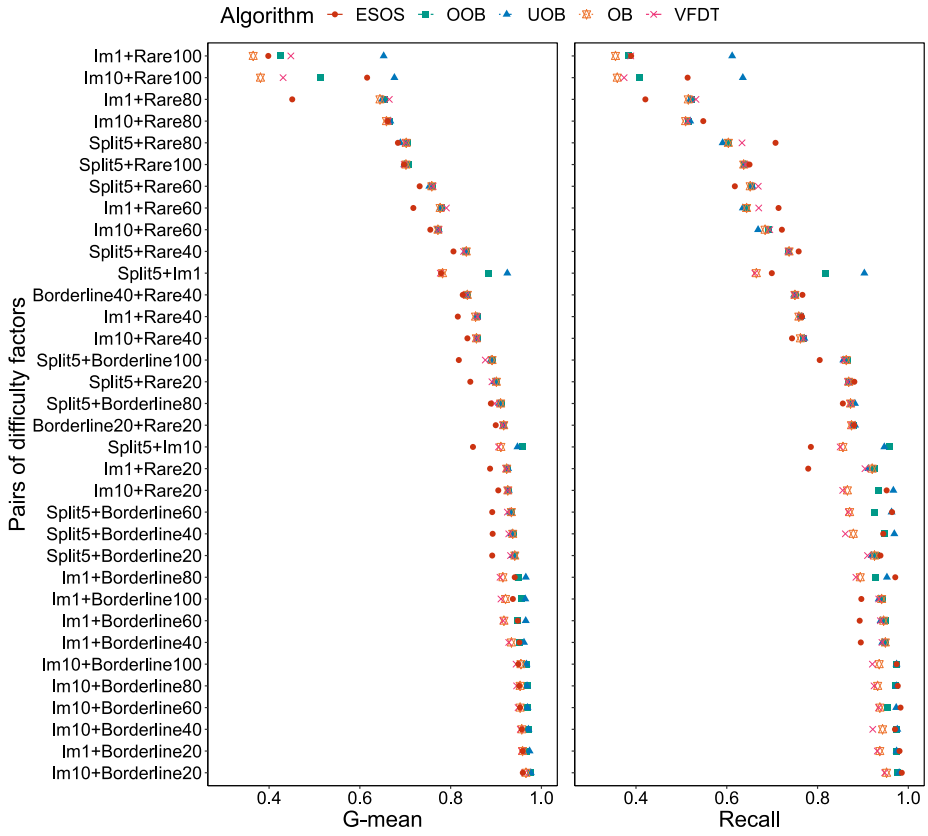


Fig. 13: Comparison of the effect of pairs of factors classifier G-mean and Recall. Scenarios ranked according to the median performance of all classifiers on a given scenario. Borderline`[N]`/Rare`[N]`: denotes a drift to `N`% borderline/rare examples, StaticIm`[N]`: `N`% minority examples before the drift, Im`[N]`: drift to `N`% minority ratio, Split`[N]` splitting into `N` sub-clusters, +: combination of factors.

combining split and rare examples UOB substitutes OOB at the first position in the ranking for G-mean, whereas ESOS takes the better rank for Recall.

RQ2: Pairing imbalance ratios (both static and drifts) with other factors amplifies the deterioration of classifier performance. Similar effects were observed for combinations involving rare/borderline minority examples, as well as changes in class composition. The presence of rare types of examples in pairs is the most influential element. The strongest interactions occur for high static imbalance (1%), however, moderate imbalance (10%) was also more detrimental in pairs than as an isolated factor. High static imbalance is also more influential than the analog imbalance ratio drifts.

Table 5: Mean Friedman test ranks for comparing classifier performance using the averaged G-mean on different data stream sets involving pairs of changes and multiple data difficulty factors. All test were significant with p-values < 0.0001. Best value (lowest rank) and values which were found to be not statistically different from the best value according to the Nemenyi post-hoc test are highlighted in bold. Column *CD* shows the critical distance of the Nemenyi post-hoc test at significance level $\alpha = 0.05$.

Data stream set	Metric	OOB	UOB	OB	VFDT	ESOS	*CD*
pairs: imbalance + move	G-mean	**2.00**	**1.50**	4.67	4.33	**2.50**	1.82
pairs: imbalance + join		**2.00**	**1.50**	4.33	4.17	**3.00**	1.82
pairs: imbalance + split		**2.17**	**1.39**	4.61	4.22	**2.61**	1.47
pairs: imbalance + borderline		**2.18**	**1.25**	4.25	4.60	2.73	0.97
pairs: imbalance + rare		**2.35**	**2.03**	4.48	3.53	**2.63**	0.97
pairs: split + borderline		**1.82**	**1.88**	4.28	4.02	3.00	0.87
pairs: split + rare		**2.28**	**1.92**	4.32	3.64	2.84	0.87
multiple factors		**2.15**	**1.76**	4.27	3.91	2.91	0.43
pairs: imbalance + move	Recall	**2.92**	**1.42**	4.67	4.33	**1.67**	1.82
pairs: imbalance + join		**2.92**	**1.25**	4.58	4.42	**1.83**	1.82
pairs: imbalance + split		**2.83**	**1.39**	4.72	4.17	**1.89**	1.47
pairs: imbalance + borderline		2.90	**1.53**	4.43	4.58	**1.58**	0.97
pairs: imbalance + rare		3.03	**1.53**	4.73	3.83	**1.90**	0.97
pairs: split + borderline		**2.46**	**1.74**	4.40	4.14	**2.26**	0.87
pairs: split + rare		2.90	**2.04**	4.60	3.54	**1.92**	0.87
multiple factors		2.82	**1.76**	4.51	3.95	**1.96**	0.43

5.4 Experiments with complex scenarios including multiple drifts

In case of complex scenarios involving multiple difficulty factors, the main general observation is that starting the learning process from a highly imbalanced stream (minority class ratio $\leq 10\%$) is more difficult than starting from a balanced stream. For example, decreases of predictive performance in streams StatIm10+Split5+Im1+Rare[N] are higher than in Split5+Im1+Rare[N], for all percentages of rare examples N. Moreover, while analyzing particular plots one can notice that the decrease in predictive performance is stronger than in scenarios involving singular or paired difficulty factors. For instance consider a stream combining a $10\% \rightarrow 1\%$ drift of the minority class ratio with a minority class split into 5 sub-clusters and a $0\% \rightarrow 40\%$ increase of the proportion of rare examples (Fig. 14). Note that OB and VFDT are practically incapable of recognizing the minority class after the drift.

Furthermore, we compared complex drifting scenarios (where difficulties appear during a drift) to corresponding static streams (where all the difficulties appear from the beginning of the stream). A sample of this comparison is given in Table 6. One can easily that G-mean values for static streams are much lower than for the corresponding drifting streams. Moreover, there are quite strong differences in classifier performance. Again OB and VFDT failed to handle multiple difficulty factors, whereas UOB and ESOS work better. Similar observations were made for other proportions of rare/borderline examples.

We have also analyzed averaged values of G-mean and Recall of classifiers for various combinations of multiple drifts (Fig. 15). One can notice that, for all classifiers, scenarios involving various proportions of rare examples are at the top of the ranking. Moreover, the impact of multiple combinations of factors was more

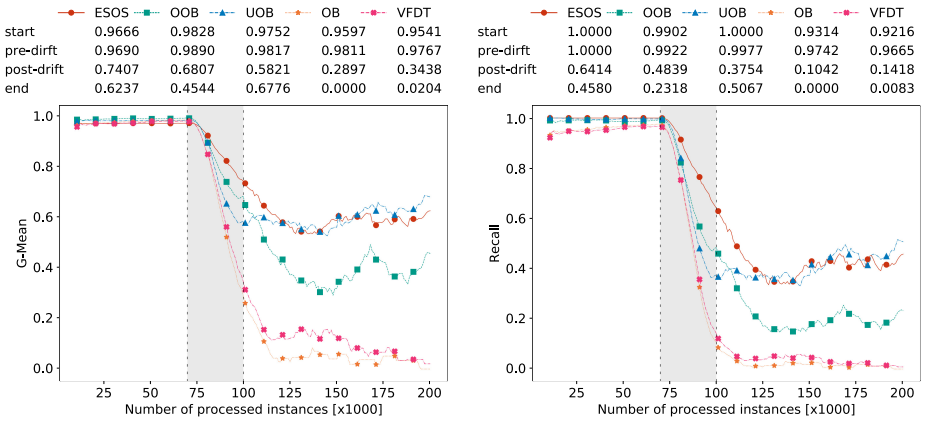


Fig. 14: G-mean (left) and Recall (right) for stream StaticIm10 + Split5 + Im1 + Rare40, which combines a 10% → 1% drift of the minority class ratio with a minority class split into 5 sub-clusters and a 0% → 40% increase of the proportion of rare examples.

Table 6: G-mean values of classifiers averaged over entire data streams with high imbalance, minority class split, and rare/borderline examples.

Mode	Configuration	OOB	UOB	OB	VFDT	ESOS
Static	Split5+Im1+Borderline40	0.486	0.875	0.000	0.000	0.700
Static	Split5+Im1+Rare40	0.430	0.688	0.000	0.050	0.540
Dynamic	Split5+Im1+Borderline40	0.811	0.875	0.683	0.694	0.720
Dynamic	Split5+Im1+Rare40	0.729	0.812	0.679	0.687	0.504

severe than that of pairs of factors (Fig. 13). Complex scenarios also intensified the impact of minority class composition and resulted in larger performance differences between classifiers. Whereas pairs of difficulty factors differentiated classifiers mostly at the ~ 0.050 G-mean levels, complex scenarios show differences of up to almost 0.400.

Moreover, the results of the Friedman test (Table 5) calculated on stream-averaged performance values show that when multiple difficulty factors are present in the stream, UOB is the best performing classifier in terms of both G-mean and Recall.

RQ3: Scenarios involving more than two factors intensify the impact of other data complexities. In multiple combinations, the impact of borderline examples is much more prominent when it co-occurs with other factors. Moreover, the most complex scenarios decrease performance measures more than in scenarios involving singular or paired difficulty factors, and strongly distinguish classifiers. Scenarios involving rare examples, minority class splits and high imbalance ratios are the most challenging combinations.

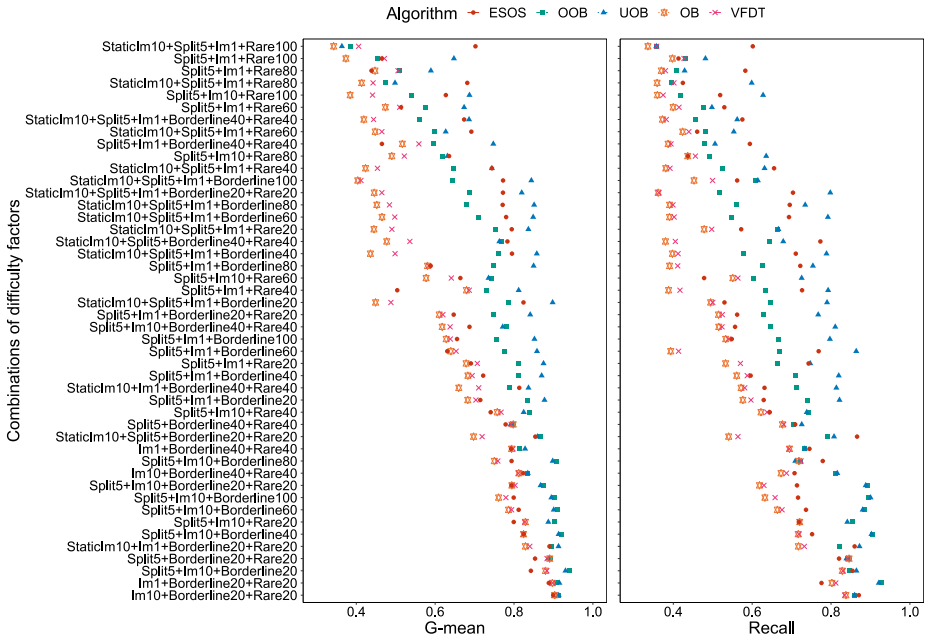


Fig. 15: Comparison of the effect of complex scenarios on G-mean and Recall of the analyzed classifiers. Scenarios ranked according to the median performance of all classifiers on a given scenario. Borderline`[N]`/Rare`[N]`: denotes a drift to `N`% borderline/rare examples, StaticIm`[N]`: `N`% minority examples before the drift, Im`[N]`: drift to `N`% minority ratio, Split`[N]` splitting into `N` sub-clusters, +: combination of factors.

5.5 Real-world imbalanced data streams

The final set of experiments involved four real-world imbalanced streams (Table 7). Although synthetic streams are better suited for performing controlled experiments and analyzing the impact of particular difficulty factors, real-world datasets are also useful to verify which of these difficulties and classifier reactions can be expected in practical applications. As there are no benchmark imbalanced data streams, we looked for datasets which at least partly contained the considered data difficulty factors. Finally, we have selected four binary classification datasets, which cover a spectrum of applications involving tweets (`Twitter` [53]), reviews (`Tripadvisor` [74]), product descriptions (`Amazon` [5]), and credit scoring data (`PAKDD` [72]).

The occurrence of minority class types and imbalance ratios were estimated in successive blocks of examples in the streams. Types of minority examples were estimated with the method proposed in [56], which analyzes labels of the neighbours of examples (see the method's brief description in Section 3.2.4). We have also estimated the number of sub-clusters and their positions in the attribute space over time. To achieve this, we clustered minority class examples in each successive block using the affinity propagation algorithm (parameterized as proposed in [22])

Table 7: Real-word datasets' characteristics. The minority class ratio, as well as the percentages of safe, borderline, rare, and outlier examples given as min–max ranges based on estimations of blocks of 2,000 examples. The ranges show the amplitude of global imbalance and minority class composition drifts.

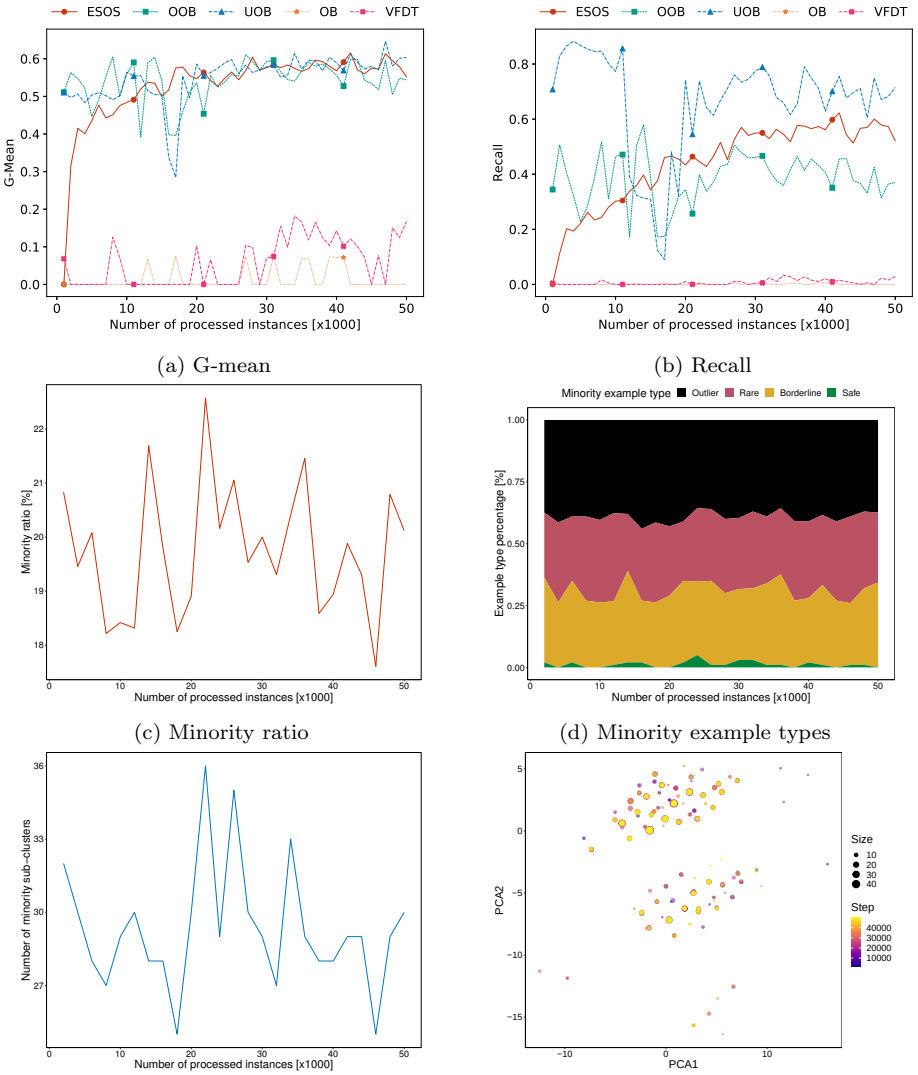
	`Amazon` [5]	`PAKDD` [72]	`Tripadvisor` [74]	`Twitter` [53]
Examples	8,000	49,997	20,491	9,090
Classes	2	2	2	2
Features	30	34	30	30
Estimated clusters	12–16	25-36	20–31	20–27
Minority ratio	12–16%	18–23%	20–34%	13–19%
Safe	0–1%	0–5%	30–45%	1–7%
Borderline	11–29%	25–37%	33–39%	20–34%
Rare	35–43%	36–41%	12–19%	29–35%
Outlier	30–51%	23–36%	9–18%	29–45%

and removed clusters smaller than six examples. Then, we visualized the relative positions of cluster exemplars using PCA precomputed on the entire dataset and applied to minority cluster representatives in each block.

As Table 7 shows, the selected streams all showcase multiple data difficulties. Interestingly, all datasets contain all types of minority examples, with a relatively high number of outliers. More precisely `Amazon`, `PAKDD`, and `Twitter` have very low proportions of safe examples, with minority examples mostly attributed to rare and outlier example types. `Tripadvisor` has a higher proportion of safe and borderline examples. Importantly, an analysis of the minority class over subsequent blocks of examples has shown that the global imbalance ratio, example type proportions, and minority class composition vary over the course of the streams. This is reflected by the minimum and maximum ratios shown in Table 7, and by plots presented in Figs. 16c– 16f[4]. Although, these real-world streams are partly similar to the scenarios with multiple data factors discussed in the precious section, they should be considered as even more difficult ones, due to the very low number of safe examples. Furthermore, discovering a quite high number of clusters may indicate quite strong decomposition of classes, which also change over streams; see Table 7. It constitutes additional difficulties for learning classifiers.

Figure 16 shows the G-mean and Recall of the analyzed classifiers on the `PAKDD` dataset. There is a very clear (∼0.500) difference in G-mean between specialized (OOB, UOB, ESOS) and non-specialized (OB, VFDT) classifiers. However, the G-mean values are generally much lower than those seen on synthetic streams, even though the stream is not that highly imbalanced. We could hypothesize that the difficulty of classifying real-world imbalanced data does not lie in the global imbalance ratio alone. Indeed, the `PAKDD` dataset has a drifting global imbalance ratio (Fig. 16c), has a very low proportion of safe minority examples (Fig. 16d), and minority sub-cluster appeared and disappear over time (Fig. 16e–16f). It is also worth noting that ESOS performs particularly well, compared to previous experiments on single factors, even in the presence of a drift visible around the 15k example. ESOS is indeed the only classifier equipped with a drift detector.

[4] Similar plots for other real-world data streams are provided in Section E of the supplement.



(a) G-mean (b) Recall

(c) Minority ratio (d) Minority example types

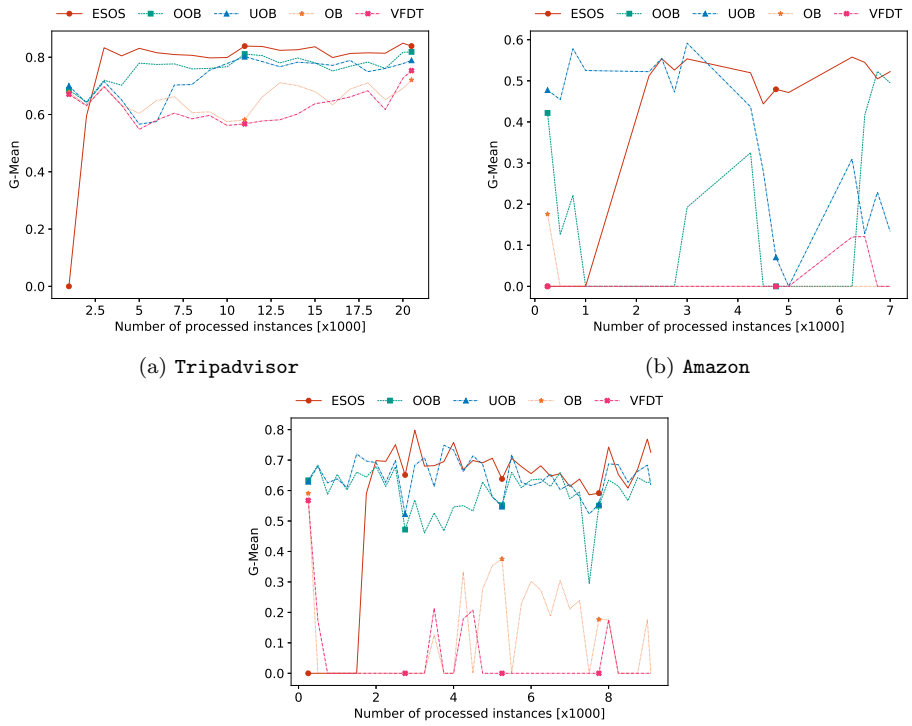
(e) Number of sub-clusters (f) Sub-cluster exemplars over time

Fig. 16: PAKDD (a) G-mean, (b) Recall, (c) minority ratio (d) example type proportions, (e) sub-cluster count, (f) PCA-visualized sub-cluster positions with color representing time and point sizes representing cluster sizes.

Interestingly, similar classifier reactions could be observed for the most complex synthetic streams (e.g., Fig. 14).

Plots for the remaining real-world datasets showed similar patterns. Due to the much lower number of safe examples, classifier G-mean values were generally lower than those observed for synthetic streams. Moreover, the gap between specialized and non-specialized classifiers was always substantial (Fig. 17). The remaining real-

world dataset also suffered from drifting global class imbalance (Supplementary Fig. S6), changing example type proportions (Supplementary Fig. S7), and changing minority sub-clusters (Supplementary Fig. S8). On the `Tripadvisor` dataset, the classifiers obtained higher measure values than on the remaining real world streams. This is consistent with the fact that `Tripadvisor` contains a higher proportion of safe examples and is less imbalanced (Table 7). `Amazon` proved to be the most difficult stream. Poor classifier performance could be explained by a very unsafe distribution of the minority class types. This is the only data with nearly no safe examples and the highest number of rare examples and outliers. It is also the dataset with the highest class imbalance ratio and the estimated minority sub-clusters show strong movement over time (Supplementary Fig. S8). As a result, VFDT and OB were unable to learn classes from this relatively short stream. Furthermore, all the analyzed real-world streams were affected by more drifts than it was considered in the synthetic streams. Finally, it is also interesting to notice that, even though ESOS needed more time to achieve its final performance level it performed better than OOB and UOB (or came close second) for all of the real streams.



(a) `Tripadvisor` (b) `Amazon`

(c) `Twitter`

Fig. 17: Classifier G-mean for the `Tripadvisor`, `Amazon`, and `Twitter` datasets.

Table 8 presents averaged values of performance measures calculated over entire streams. For both G-mean and Recall UOB ranked first, followed by ESOS. Due to the small number of datasets, the statistical significance of these differences cannot be evaluated. However, the difference between mean performance on each dataset (effect size) is large when one compares specialized (OOB, UOB, ESOS) and non-specialized (OB, VFDT) classifiers.

Table 8: Average classifier performance values on the real-world data streams.

Data stream	Metric	OOB	UOB	OB	VFDT	ESOS
`Amazon`	G-mean	0.241	0.394	0.021	0.021	0.402
`PAKDD`		0.542	0.549	0.013	0.059	0.530
`Tripadvisor`		0.764	0.731	0.656	0.627	0.771
`Twitter`		0.600	0.654	0.155	0.075	0.598
Averaged Friedman ranks		2.25	1.75	4.38	4.63	2
`Amazon`	Recall	0.086	0.708	0.002	0.002	0.391
`PAKDD`		0.386	0.663	0.001	0.007	0.444
`Tripadvisor`		0.699	0.658	0.468	0.431	0.716
`Twitter`		0.435	0.574	0.040	0.019	0.561
Averaged Friedman ranks		2.75	1.5	4.38	4.63	1.75

Real-world streams: The analyzed datasets resembled the most complex scenarios in that they combine drifting class imbalance and drifts of unsafe minority examples. However the number of drifts is higher. Moreover, the real-world datasets contained a high percentage of outlier examples, making them even harder. As in the complex synthetic scenarios, specialized classifiers clearly outperformed non-specialized classifiers. UOB and ESOS ranked highest in terms of average G-mean and Recall, with the latter learning at a slower pace, but being less susceptible to drifts.

6 Conclusions

The main goal of this paper, was to highlight the importance of other difficulty factors than the global imbalance ratio in drifting imbalanced data streams. To this end, we have proposed a new categorization of concept drifts for imbalanced streams, which includes criteria of *imbalance ratio*, *locality*, *class composition*, and *minority example distribution*. To the best of our knowledge, the three latter criteria were not previously adopted to systematically characterize concept drifts. Moreover, we have developed and made publicly available a data stream generator that follows the proposed drift categorization. Finally, using 385 generated synthetic streams and four real-world datasets we have carried a comprehensive set of experiments to study the impact of drifts and difficulty factors on the performance of online stream classifiers.

The obtained experimental results led to the following main observations:

- Specialized imbalanced stream classifiers (OOB, UOB, ESOS-ELM) coped well with global class imbalance (except the highest ones), whereas non-specialized classifiers (OB, VFDT) performed worse.
- Among the single data difficulty factors, complex distributions of minority examples in balanced data streams were more challenging than static imbalanced data streams. In particular, experiments with rare type examples were the only scenarios where the classifiers did not recover from in any way. Interestingly, all classifiers were fairly robust to larger proportions of borderline examples in balanced data streams.
- High imbalance ratios amplified the detrimental effect of other difficulty factors.
- All classifiers were able to successfully benefit from temporary periods of less extreme class imbalance or less challenging data difficulty factors, improving overall predictive performance compared to cases where high levels of class imbalance or challenging data difficulty factors were present from the beginning.
- Rare examples were the most influential stream characteristic when combined with other factors. Out of the analyzed classifiers, UOB was the best choice for complex streams with multiple difficulty factors.
- Real-world datasets combine multiple data difficulty factors, being more difficult than synthetic streams. In particular, the proportion of safe minority examples in the analyzed datasets was very low and the minority ratio and class composition drifted quite frequently over time. Real-world streams distinguished classifiers similarly to complex synthetic scenarios with multiple elements in streams, albeit with a larger performance gap between specialized and non-specialized classifiers. UOB and ESOS would be the classifiers of choice for real-world datasets.

Future research for imbalanced data stream classifiers should focus on other difficulty factors than just global class imbalance. None of the analyzed classifiers were able to cope with rare examples and outliers combined with minority class splits. This calls for new online methods that take into account the composition and the types of examples of the minority class. The need for research in this area is further emphasised by the fact that real world data streams are likely to involve more complex scenarios, as demonstrated in this study.

The methodological similarity between the proposed class composition criterion and stream cluster transitions types [66, 67], suggests that stream clustering methods could be instrumental in tracking minority class evolution. The characteristic of different minority example types could be taken into account by differentiating resampling methods based on whether examples are safe, borderline, rare or outlying. This could be achieved by separate minority example memories, similar to minority class buffers known from block-based classifiers [14, 15]. We also note that currently drift detectors monitor only classifier performance, whereas they could potentially track minority class composition and changes of example type proportions. Finally, another challenge is to generalize the proposed categorization and design classifiers for multiple imbalanced classes.

Acknowledgements This work was partly supported by PUT Institute of Computing Science Statutory Funds. Leandro Minku was funded by EPSRC Grant Nos. EP/R006660/1 and EP/R006660/2. Moreover the research of J.Stefanowski was partially supported by TAILOR, a project funded by EU Horizon 2020 research and innovation programme under GA No 952215.

References

1. Ancy S, Paulraj D (2020) Handling imbalanced data with concept drift by applying dynamic sampling and ensemble classification model. Computer Communications 153:553–560
2. Bifet A, Holmes G, Kirkby R, Pfahringer B (2010) MOA: Massive Online Analysis. J Mach Learn Res 11:1601–1604
3. Błaszczyński J, Stefanowski J (2015) Neighbourhood sampling in bagging for imbalanced data. Neurocomputing 150 A:184–203
4. Błaszczyński J, Stefanowski J (2018) Local Data Characteristics in Learning Classifiers from Imbalanced Data. In: Kacprzyk J, Rutkowski L, Gaweda A, Yen G (eds) Advances in Data Analysis with Computational Intelligence Methods. Springer series Studies in Computational Intelligence., Springer, pp 51–85
5. Blitzer J, Dredze M, Pereira F (2007) Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In: ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic
6. Branco P, Torgo L, Ribeiro R (2016) A survey of predictive modeling under imbalanced distributions. ACM Comput Surv 49(2):31
7. Breiman L (1996) Bagging predictors. Machine Learning 24(2):123–140
8. Brzezinski D, Stefanowski J (2014) Reacting to different types of concept drift: The accuracy updated ensemble algorithm. IEEE Trans on Neural Netw Learn Syst 25(1):81–94
9. Brzezinski D, Stefanowski J (2017) Prequential auc: Properties of the area under the roc curve for data streams with concept drift. Knowledge and Information Systems 52(2):531–562
10. Brzezinski D, Stefanowski J (2018) Ensemble Classifiers for Imbalanced and Evolving Data Streams, World Scientific, pp 44–68. DOI 10.1142/9789813228047_0003
11. Brzezinski D, Stefanowski J, Susmaga R, Szczech I (2018) Visual-based analysis of classification measures and their properties for class imbalanced problems. Information Sciences 462:242–261
12. Brzezinski D, Stefanowski J, Susmaga R, Szczech I (2019) On the dynamics of classification measures for imbalanced and streaming data. IEEE Transactions on Neural Networks and Learning Systems
13. Cabral G, Minku L, Shihab E, Mujahid S (2019) Class imbalance evolution and verification latency in just-in-time software defect prediction. In: Proceedings of the International Conference on Software Engineering (ICSE)
14. Chen S, He H (2009) Sera: Selectively recursive approach towards nonstationary imbalanced stream data mining. In: Proceedings of the 2009 International Joint Conference on Neural Networks, pp 522–529
15. Chen S, He H (2011) Towards incremental learning of nonstationary imbalanced data stream: a multiple selectively recursive approach. Evolving Systems pp 35–50
16. Demsar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30
17. Ditzler G, Polikar R (2013) Incremental learning of concept drift from streaming imbalanced data. IEEE Transactions on Knowledge and Data Engineering

25(10):2283–2301
18. Ditzler G, Roveri M, Alippi C, Polikar R (2015) Learning in nonstationary environments: A survey. IEEE Comp Int Mag 10(4):12–25
19. Domingos P, Hulten G (2000) Mining high-speed data streams. In: Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 71–80
20. Fernández A, García S, Galar M, Prati RC, Krawczyk B, Herrera F (2018) Learning from Imbalanced Data Sets. Springer
21. Fernández A, García S, Herrera F, Chawla NV (2018) SMOTE for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary. J Artif Intell Res 61:863–905
22. Frey BJ, Dueck D (2007) Clustering by passing messages between data points. Science 315(5814):972–976
23. Gama J (2010) Knowledge Discovery from Data Streams. Chapman and Hall
24. Gama J, Castillo G (2006) Learning with local drift detection. In: International Conference on Advanced Data Mining and Applications, pp 42–55
25. Gama J, Sebastião R, Rodrigues PP (2013) On evaluating stream learning algorithms. Mach Learn 90(3):317–346
26. Gama J, Zliobaite I, Bifet A, Pechenizkiy M, Bouchachia A (2014) A survey on concept drift adaptation. ACM Comput Surv 46(4):44:1–44:37
27. Gao J, Fan W, Han J, Yu PS (2007) A general framework for mining concept-drifting data streams with skewed distributions. In: Proceedings of the 2007 SIAM International Conference on Data Mining, pp 3–14
28. Gao J, Ding B, Han J, Fan W, Yu PS (2008) Classifying data streams with skewed class distributions and concept drifts. IEEE Internet Computing 12(6):37–49
29. Garcia V, Sanchez J, Mollineda R (2007) An empirical study of the behaviour of classifiers on imbalanced and overlapped data sets. In: Proc. of Progress in Pattern Recognition, Image Analysis and Applications , LNCS, Springer, vol 4756, pp 397–406
30. Ghazikhani A, Monsefi R, Yazdi H (2013) Recursive least square perceptron model for non-stationary and imbalanced data stream classification. Evolving Systems 4(2):119–131
31. Ghazikhani A, Monsefi R, Yazdi H (2014) Online neural network model for non-stationary and imbalanced data stream classification. International Journal of Machine Learning and Cybernetics 5(1):51–62
32. Goldenberg I, Webb G (2019) Survey of distance measures for quantifying concept drift and shift in numeric data. Knowledge and Information Systems 60:591–615
33. Gomes H, Barddal J, Enembreck F, Bifet A (2017) A survey on ensemble learning for data stream classification. ACM Computing Surveys 50(2):23:1–36
34. He H, Ma Y (eds) (2013) Imbalanced Learning: Foundations, Algorithms, and Applications. Wiley-IEEE Press
35. Hoens T, Chawla V (2013) Learning in non-stationary environments with class imbalance. In: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, pp 168–176
36. Japkowicz N, Stephen S (2002) The class imbalance problem: A systematic study. Intell Data Anal 6(5):429–449

37. Jo T, Japkowicz N (2004) Class imbalances versus small disjuncts. SIGKDD Explorations 6(1):40–49
38. Khamassi I, Sayed-Mouchaweh M, Hammami M, Ghédira K (2018) Discussion and review on evolving data streams and concept drift adapting. Evolving systems 9(1):1–23
39. Krawczyk B, Skryjomski P (2017) Cost-sensitive perceptron decision trees for imbalanced drifting data streams. In: Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18-22, 2017, Proceedings, Part II, pp 512–527
40. Krawczyk B, Minku L, Gama J, Stefanowski J, Woźniak M (2017) Ensemble learning for data stream analysis: A survey. Information Fusion 37:132–156
41. Kubat M, Matwin S (1997) Addressing the curse of imbalanced training sets: one-side selection. In: Proc. of the 14th Int. Conf. on Machine Learning ICML-97, pp 179–186
42. Lango M (2019) Tackling the problem of class imbalance in multi-class sentiment classification: An experimental study. Foundations of Computing and Decision Sciences 44(2):151–178
43. Laurikkala J (2001) Improving identification of difficult small classes by balancing class distribution. Tech. Rep. A-2001-2, University of Tampere, DOI 10.1007/3-540-48229-6_9
44. Levin D, Peres Y, Wilmer E (2008) Markov Chains and Mixing Times
45. Lichtenwalter R, Chawla N (2010) Adaptive methods for classification in arbitrarily imbalanced and drifting data streams. In: New Frontiers in Applied Data Mining – Lecture Notes in Computer Science, vol 5669, pp 53–75
46. Lopez V, Fernandez A, Garcia S, Palade V, Herrera F (2014) An Insight into Classification with Imbalanced Data: Empirical Results and Current Trends on Using Data Intrinsic Characteristics. Inf Sci 257:113–141
47. Lu Y, Cheung YM, Tang Y (2017) Dynamic weighted majority for incremental learning of imbalanced data streams with concept drift. In: International Joint Conference on Artificial Intelligence, pp 53–75
48. Lyon RJ, Brooke JM, Knowles JD, Stappers BW (2014) Hellinger distance trees for imbalanced streams. CoRR abs/1405.2278, 1405.2278
49. Minku L, Yao X (2012) DDD: a new ensemble approach for dealing with concept drift. IEEE Transactions on Knowledge and Data Engineering 24(4):619–633
50. Minku L, White A, Yao X (2010) The impact of diversity on on-line ensemble learning in the presence of concept drift. IEEE Transactions on Knowledge and Data Engineering 22:730–742
51. Minku LL (2019) Transfer learning in non-stationary environments. In: Sayed-Mouchaweh M (ed) Learning from Data Streams in Evolving Environments: Methods and Applications, Springer International Publishing, Cham, pp 13–37
52. Mirza B, Lin Z, Liu N (2015) Ensemble of subset online sequential extreme learning machine for class imbalance and concept drift. Neurocomputing 149:316–329
53. Nakov P, Ritter A, Rosenthal S, Sebastiani F, Stoyanov V (2016) Semeval-2016 task 4: Sentiment analysis in twitter. In: Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016, pp 1–18

54. Napierała K, Stefanowski J (2012) BRACID: a comprehensive approach to learning rules from imbalanced data. J Intell Inf Syst 39:335–373
55. Napierała K, Stefanowski J (2012) The influence of minority class distribution on learning from imbalance data. In: Proc. of the 7th Int. Conf. HAIS 2012, pp 139–150
56. Napierala K, Stefanowski J (2016) Types of minority class examples and their influence on learning classifiers from imbalanced data. Journal of Intelligent Information Systems 46(3):563–597
57. Napierała K, Stefanowski J, Wilk S (2010) Learning from imbalanced data in presence of noisy and borderline Examples. In: Proc. of 7th Int. Conf. RSCTC 2010, LNAI, Springer, vol 6086, pp 158–167
58. Nickerson A, Japkowicz N, Milios E (2001) Using unsupervised learning to guide re-sampling in imbalanced data sets. Proceedings of the Eighth International Workshop on AI and Statitsics
59. Olaitan OM, Viktor HL (2018) SCUT-DS: learning from multi-class imbalanced canadian weather data. In: Foundations of Intelligent Systems - 24th International Symposium, ISMIS 2018, Limassol, Cyprus, October 29-31, 2018, Proceedings, pp 291–301
60. Oza NC, Russell S (2001) Online bagging and boosting. In: Jaakkola T, Richardson T (eds) Eighth International Workshop on Artificial Intelligence and Statistics, Morgan Kaufmann, Key West, Florida. USA, pp 105–112
61. Oza NC, Russell SJ (2001) Experimental comparisons of online and batch versions of bagging and boosting. In: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, San Francisco, CA, USA, August 26-29, 2001, pp 359–364
62. Prati R, Batista G, Monard M (2004) Class imbalance versus class overlapping: an analysis of a learning system behavior. In: Proc. 3rd Mexican Int. Conf. on Artificial Intelligence, pp 312–321
63. Ren S, Liao B, Zhu W, Li Z, Liu W, Li K (2018) The gradual resampling ensemble for mining imbalanced data streams with concept drift. Neurocomputing 286:150–166
64. Sarnelle J, Sanchez A, Capo R, Haas J, Polikar R (2015) Quantifying the limited and gradual concept drift assumption. In: Proceedings of the 2015 International Joint Conference on Neural Networks (IJCNN)
65. Sousa MR, Gama J, Brandao E (2016) A new dynamic modeling framework for credit risk assessment. Expert Systems with Applications 45:341–351
66. Spiliopoulou M, Ntoutsi E, Theodoridis Y, Schult R (2006) MONIC: modeling and monitoring cluster transitions. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 706–711
67. Spiliopoulou M, Ntoutsi E, Theodoridis Y, Schult R (2013) MONIC and followups on modeling and monitoring cluster transigons. In: European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD), pp 622–626
68. Stefanowski J (2013) Overlapping, rare examples and class decomposition in learning classifiers from imbalanced data. In: Ramanna S, Jain LC, Howlett RJ (eds) Emerging Paradigms in Machine Learning, vol 13, Springer, pp 277–306
69. Stefanowski J (2016) Dealing with data difficulty factors while learning from imbalanced data. In: Mielniczuk J, Matwin S (eds) Challenges in Computa-

tional Statistics and Data Mining, Springer, pp 333–363
70. Street WN, Kim Y (2001) A streaming ensemble algorithm (SEA) for large-scale classification. In: Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, pp 377–382
71. Sun Y, Tang K, Minku L, Wang S, Yao X (2016) Online ensemble learning of data streams with gradually evolved classes. IEEE Transactions on Knowledge and Data Engineering 28(6):1532–1545
72. Theeramunkong T, Kijsirikul B, Cercone N, Ho TB (2009) PAKDD data mining competition
73. Toffoli T, Margolus N (1987) Cellular Automata Machines: A New Environment for Modeling. MIT Press
74. Wang H, Lu Y, Zhai C (2010) Latent aspect rating analysis on review text data: a rating regression approach. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, July 25-28, 2010, pp 783–792
75. Wang S, Minku L, Yao X (2013) Concept drift detection for online class imbalance learning. In: Proceedings of the 2013 International Joint Conference on Neural Networks (IJCNN'13), pp 1–8
76. Wang S, Minku LL, Yao X (2015) Resampling-based ensemble methods for online class imbalance learning. IEEE Trans Knowl Data Eng 27(5):1356–1368
77. Wang S, Minku L, Yao X (2016) Dealing with multiple classes in online class imbalance learning. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16), pp 2118—2124
78. Wang S, Minku LL, Yao X (2018) A systematic study of online class imbalance learning with concept drift. IEEE Trans Neural Netw Learning Syst 29(10):4802–4821
79. Webb GI, Hyde R, Cao H, Nguyen H, Petitjean F (2016) Characterizing concept drift. Data Min Knowl Discov 30(4):964–994
80. Webb GI, Lee LK, Goethals B, Petitjean F (2018) Analyzing concept drid and shid from sample data. Data Min Knowl Discov 32(5):1179–1199
81. Weiss G (2010) The Impact of Small Disjuncts on Classifier Learning, vol 8, pp 193–226
82. Wu K, Edwards A, Fan W, Gao J, Zhang K (2014) Classifying imbalanced data streams via dynamic feature group weighting with importance sampling. In: Proceedings of the 2014 SIAM International Conference on Data Mining, pp 722–730
83. Zhang H, Liu W, Wang S, Shan J, Liu Q (2019) Resample-based ensemble framework for drifting imbalanced data streams. IEEE Access 7:65103–65115
84. Zliobaite I (2013) Controlled permutations for testing adaptive learning models. Knowl Inf Syst pp 1–14
85. Zliobaite I, Budka M, Stahl F (2015) Towards cost-sensitive adaptation: When is it worth updating your predictive model? Neurocomputing 150:240–249
86. Zliobaite I, Pechenizkiy M, Gama J (2015) An overview of concept drift applications. In: Japkowicz N, Stefanowski J (eds) Big Data Analysis: New Algorithms for a New Society, Springer