Online Feature Selection with Capricious Streaming Features: A General Framework

Di Wu[a,e] *Member, IEEE*, Yi He[d], Xin Luo[a,b], *Senior Member, IEEE*, Mingsheng Shang[a], and Xindong Wu[c], *Fellow, IEEE*

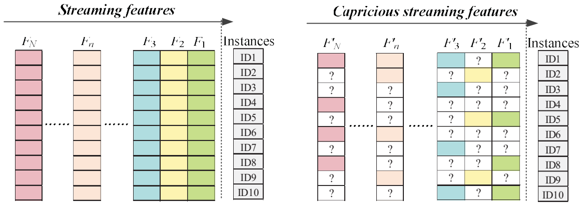
[a] Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing, 400714, China
[b] Department of Big Data Analyses Techniques, Cloudwalk, Chongqing 401331, China
[c] Mininglamp Academy of Sciences, Mininglamp Technology, Beijing, China
[d] University of Louisiana at Lafayette, Lafayette, 70503, USA
[e] University of Chinese Academy of Sciences, Beijing, 100049, China
{wudi, luoxin21, msshang}@cigit.ac.cn, {yi.he1, xwu}@louisiana.edu

***Abstract*—Online streaming feature selection has received extensive attention in the past few years. Existing approaches have a common assumption that the feature space of the fixed data instances increases dynamically without any missing entry. This assumption, however, does not always hold in many real-world applications. For example, in a credit evaluation system, we cannot collect the complete dynamic features for each person and/or enterprise. Motivated by this observation, this paper aims at conducting online feature selection from *capricious streaming features*, where features flow in one by one with some random missing entries while the number of data instances remains fixed. To do so, we propose a general framework named GF-CSF. The main idea of GF-CSF is to adopt latent factor analysis to preprocess capricious streaming features for completing their missing entries before conducting feature selection. Both theoretical and experimental analyses indicate that GF-CSF can efficiently improve any existing model of online streaming features selection to achieve online capricious streaming features selection.**

***Keywords*—big data, online feature selection, latent factor analysis, capricious streaming features.**

I. Introduction

High dimensionality is a typical characteristic of big data and ubiquitous in many fields [1]. Data with high dimensionality can cause the problems of high storage and computational cost, performance degradation on unseen data, and difficulty in facilitating data visualization and understanding [2]. Feature selection is one of the most powerful tools for addressing such problems [3]. Its task is to search for an optimal subset from original high-dimensional feature space under some criteria, such as maximizing relevance and minimizing redundancy to class labels in classification[4].



(a) Streaming features (b) Capricious streaming features

Fig. 1. The illustration for the different dynamic features scenarios. The observed features are marked in multiple colors and the unobserved features (missing entries) are represented by the symbol “?”.

Traditional feature selection approaches conduct feature selection under the condition that all candidate features are available [5]. Unfortunately, in real-world applications, the feature space often keeps expanding continuously [6]. Under such circumstances, it is infeasible to gather all the features before conducting feature selection [4]. To address this issue, *online streaming features selection* is achieved by some researchers [4, 7-10]. It can carry out feature selection in real-time rather than waiting for all the features. Representative algorithms on online streaming features selection include Grafting [7], Alpha-investing [8], OSFS [4], SAOLA [9], and OSFASW [10].

Although these algorithms are different from each other on model design, they all have a common assumption that the feature space of fixed data instances increases dynamically without any missing entry (called *streaming features* [4]), as illustrated in Fig.1(a). This assumption, however, does not always hold in many real-world applications where the dynamic features cannot be collected completely. For example, in a smart healthcare platform [11], since a patient’s features (describing the symptoms) come from different inspection devices (pulse monitors, thermometers, *etc.*) and service providers (labs, hospitals, *etc.*), it is impracticable to collect each feature for each patient. Motivated by this example, we formulate such dynamic features as *capricious streaming features*, i.e., features flow in one by one with some random missing entries while the number of data instances remains fixed, as illustrated in Fig.1(b).

To address the newly formulated problem, we propose a general framework, which is highly compatible with any

This work is supported in part by the U.S. National Science Foundation under Grant 1613950 and Grant 1763620, in part by the National Natural Science Foundation of China under Grants 61702475, 61772493, 61902370, 61602434, and 91646114, in part by the Natural Science Foundation of Chongqing (China) under grants cstc2019jcyjjqX0013 and cstc2019jcyj-msxmX0578, in part by the Chongqing Research Program of Key Standard Technologies Innovation of Key Industries under grant cstc2017zdcy-zdyfX0076, in part by the Chongqing Municipal Science and Technology Commission under Grant No.cstc2015jcyjA10062, and in part by the Pioneer Hundred Talents Program of Chinese Academy of Sciences (*X. Luo and M. Shang are the Corresponding Authors of this study*).

978-1-7281-0858-2/19/$31.00 ©2019 IEEE