European Commission

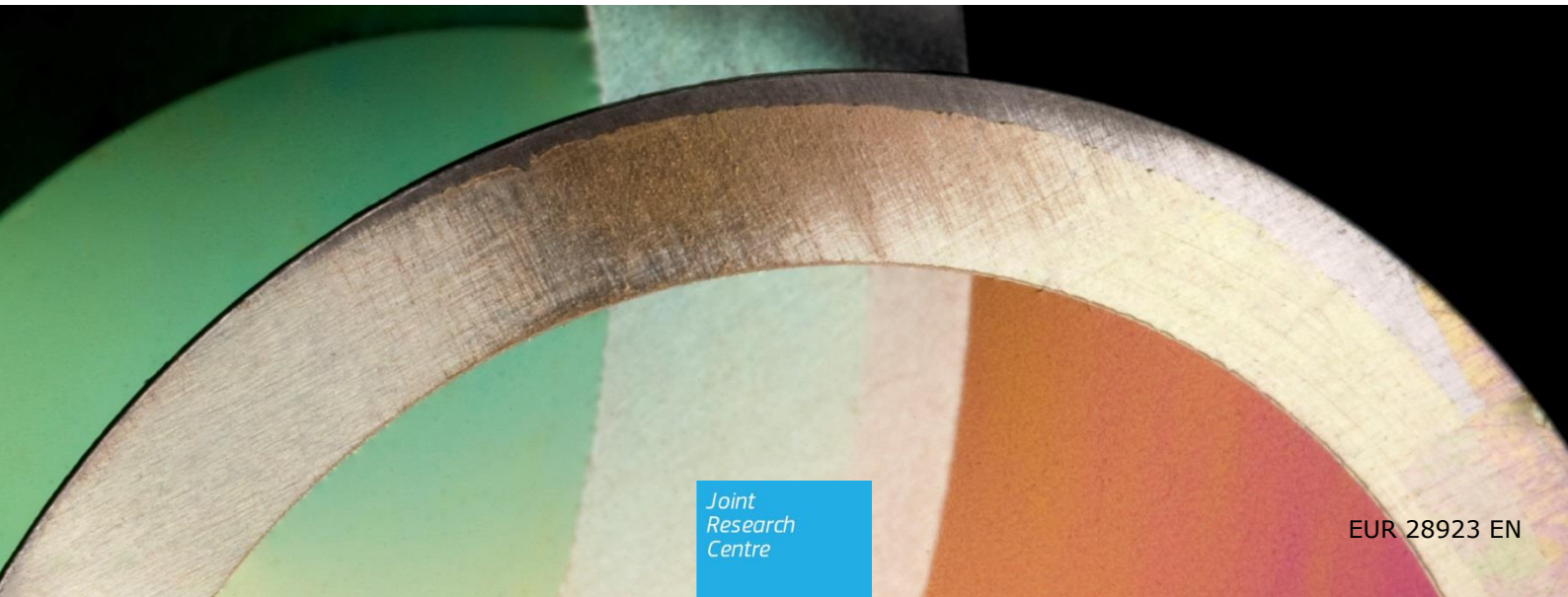
JRC SCIENCE FOR POLICY REPORT

Data science applications to connected vehicles

Key barriers to overcome

Gómez Losada, A.

2017



Joint Research Centre

EUR 28923 EN

Contact information
Name: Álvaro Gómez Losada
Address: European Commission, Joint Research Centre, Edficio Expo, c/ Inca Garcilaso 3, E-41092 Seville,Spain
Email: alvaro.gomez-losada@ec.europa.eu
Tel.: +34 954 488 480

JRC Science Hub
https://ec.europa.eu/jrc

JRC108572

EUR 28923 EN

PDF ISBN 978-92-79-77041-8 ISSN 1831-9424 doi:10.2760/822136

Luxembourg: Publications Office of the European Union, 2017

How to cite this report: Gómez Losada, A., *Data science applications to connected vehicles Key barriers to overcome*, EUR 28923 EN, Publications Office of the European Union, Luxembourg, 2017, ISBN 978-92-79-77041-8, doi:10.2760/822136, JRC108572

Data science applications to connected vehicles

The connected vehicles will generate huge amount of pervasive and real time data, at very high frequencies. This poses new challenges for Data science. How to analyse these data and how to address short-term and long-term storage are some of the key barriers to overcome.

Contents

Acknowledgements

The views expressed here are purely those of the author and may not, under any circumstance, be regarded as an official position of the European Commission.

Authors

Álvaro Gómez Losada, Joint Research Centre

Executive summary

As our environments become more connected in general, Intelligent Transportation Systems will play a central role in our cities and across borders, forming part of a new vision of "mobility as a service." Connected vehicle technology, with a prominent role in Intelligent Transportation Systems, will be capable of generating huge amounts of pervasive and real time data, at very high frequencies. These streaming data are the common type of data produced by Connected vehicles, and their analysis is of paramount importance for applications improving road safety, effective service delivery, eco-driving, traffic regulation and pollution reduction. This study aims to characterize this type of data from an analytical perspective, as well as to pose the challenges Data science faces in extracting knowledge from them in real time. Data generated by sensors and actuators in Connected vehicles include noisy, anomalous, redundant, rapidly changing, correlated and heterogeneous data. In such a context, numerous techniques have been proposed to adapt the data analytics in batch learning to these new dynamic and evolving streaming data, which are produced in huge volumes and transmitted at high velocity. The Internet of Vehicles has the potential to provide a pervasive network of Connected vehicles, smart sensors and road infrastructures, and big data has the potential to process and store that amount of data and information. Modelling, predicting, and extracting meaningful information in reasonable and efficient ways from big data represent a challenge for Data science in Connected vehicles.

Policy context

This report analyses the current technology context for Connected vehicles in the Data science domain. Seven years ago, the Intelligent Transportation Systems Directive (Directive 2010/40/EC) considered the processing and use of road, traffic and travel data as a part of the necessary steps for the deployment and use of Intelligent Transportation Systems applications. Most recently, in the Declaration of Amsterdam in 2016, European transport ministers urged the European Commission to develop a European strategy on cooperative, connected and automated vehicles. The European Strategy for Low-Emission Mobility (COM/2016/501 final) adopted in July 2016, highlighted the potential of these cooperative, connected and automated vehicles to reduce energy consumption and emission from transport. Data science plays an important role to develop a layer of innovative services and applications in Connected vehicle technology. In this study, an exploratory approach to identify key barriers for the correct implementation of Data science in Connected vehicles is presented, that stakeholders may want to consider as a scenario for an eventual full realisation of a safe an efficient deployment of Connected vehicles.

Key conclusions

Implementation of Connected vehicles in Intelligent Transportation Systems will revolutionize the way we drive. The key to the success of Connected vehicle lies in how well connectivity of vehicles and infrastructure works in real life. Correct analyses of the volume of data generated by Connected vehicle technology and meaningful knowledge being generated by such vast amounts of information will be among the critical success factors. Therefore, the potential of Data science to transform our ability to deliver solutions to the Connected vehicle technology is clearly envisaged. However, there are many issues that need to be resolved in order to achieve this opportunity's maximum potential. Mining pervasive data streams requires new and efficient algorithms executed in dynamic and changing environments under time and memory constraints. Also, the selection of the data pre-processing methods, data reduction and data fusion operations becomes a challenge due to their variant complexity. The faster training and improved generalization capabilities in the learning algorithms, but also, how to storage the amount of data generated by Connected vehicles, are among the issues that need to be resolved for the correct implementation of Data science in Connected vehicle technology.

Main findings

Data streams from Connected vehicles pose new challenges for machine learning and data mining as the traditional algorithmic methods have been designed for static datasets and are not capable of efficiently analyzing a fast-growing amount of data. The main issue is that Connected vehicles data are characterized by a large number of features producing a continuous and transient flow of data, in a dynamic, non-stationary environment. The following are the main limitations and characteristics that should be taken into consideration to achieve maximum potential in the application of Data science to Connected vehicles:

— Data generated by Connected vehicles are characterized by a multitude of formats and data types and therefore have a pronounced volume and variety dimensions. To deal with data size and heterogeneity, the following must be addressed: how to improve the quality of data in real time (filtering), how to summarize and sketch them, how to unify data (fusion methodologies) and processing models, how to implement knowledge creation and reasoning, and how to address short-term and long-term storage.

— The application of data mining techniques to streaming data generated by sensors in Connected vehicles may deliver approximate results. There are many sources of uncertainty that must be considered carefully in decision making. The assessment, representation and propagation of uncertainty must be understood, as well as the development of robust optimization methods and the design of optimal sequential decision making.

— Data in Connected vehicles are generated and collected at high speed. Algorithms that process data streams must face limited computational resources like memory and time, as well as constraints to make predictions within a reasonable time. This poses the need for efficient resource-aware algorithms providing fast answers, using few memory resources.

— Presumably, static data mining and machine learning models built from fixed training sets are not prepared to process the highly-detailed data available, neither are they able to continuously maintain a predictive model consistent with the actual state of the nature surrounding a Connected vehicle, nor to react quickly to changes. Since the nature of data in a Connected vehicle environment is changing and evolving continuously over time, advancements in adaptive algorithms should be taken into account:

  - The *stability-plasticity dilemma*, which asks how a learning system can be designed to remain stable and unchanged by irrelevant events while being plastic to new, important data in the Connected vehicle environment. This is related to the continuous adaptations of the decision models, and therefore it is important to ascertain which data to remember or forget, and how and when to do the model upgrade.

  - Data mining and machine learning require continuous processing of the incoming data monitoring trends, and detecting changes. The phenomenon called *concept drift* is related to changes in the distribution of data, which occur in the streams over time. This concept might deteriorate the performance of built models.

— In dynamic streaming data like those generated in Connected vehicles, the concept of irrelevant or redundant features is now restricted to a certain period of time. Features previously considered irrelevant might become relevant, and vice-versa, to reflect the dynamic of the process generating data.

— Although there are an increasing number of streaming learning algorithms, the metrics and the design of experiments for assessing the quality of learning models is still an open issue. The design of experimental studies is of paramount importance. The continuous evolution of the decision models and the non-stationary nature of data streams are two important aspects in the evaluation methodologies. Discussions

on best practices for performance assessment and differences in performance when learning dynamic models that evolve over time should be addressed.

Related and future JRC work

This work was carried out in the context of JRC's exploratory research project ART (Autonomous Road Transport). Next steps will concern the application of Data science methodologies in a Connected vehicle's real data scenario, from a modelling and simulation perspective. A JRC Science Policy Report dealing with the Connected vehicle topic is the one entitled "The r-evolution of driving: from Connected Vehicles to Coordinated Automated Road Transport (C-ART)" which was released on May 2017 and presented on the "Challenge & Opportunities of Coordinated Automated Road Transport (C-ART)" in Brussels (12th June, 2017).

Quick guide

This study examines the analytical lifecycle of streaming data generated in Connected vehicles by means of a selected literature analysis. The scope of the different studies (articles, monographs, reports, books, conference proceedings and web sites) collected covers the following topics:

— Data analytics in Intelligent Transportation Systems.

— Intelligent Transportation Systems sensors for traffic management and Connected vehicles.

— Data mining techniques in sensor networks.

— Machine learning and Knowledge discovery databases in streaming data.

— Data science and big data computing.

— Data quality and data lifecycles in Intelligent Transportation Systems.

— Data visualization.

Although they are interesting topics, aspects such as image processing and recognition, the ownership, privacy and security of data, large data storage, and communication technologies in Connected vehicles have not been discussed in this document. These aspects cover a wide range of areas that can be analyzed from multiple points of view, representing disciplines in their own rite. These issues are considered to be tangential to the scope of this document. Besides, addressing their proper analysis would have implied extending considerably its length beyond an exploratory nature.

1 Introduction

The term Connected vehicle (CV) refers to applications, services, and technologies that connect a vehicle to its surroundings (e.g., other vehicles, roadside infrastructure, traffic management centres). CV enables us to instrument our physical environment with complex sensor and actuators and creating a connected world that generates huge volumes of interconnected data. The importance of this trend can be seen in the growing momentum of exemplars such as the Internet of Things, smart environment and smart cities. CV has recently been further enhanced by the concept of Internet of Vehicles (IoV), where the smart car is envisaged.

CVs are elements of Intelligent Transportation Systems (ITS). ITS are an innovative technology, which has emerged for improving the safety, operation and environmental impact of transportation networks. ITS promise to be one of the most dynamic and innovative specialization fields with a large expected industrial and business growth over the next decade.

There are, however, many issues that need to be resolved to achieve maximum potential, including privacy and security issues, data processing and storage, development of standards and regulation across all platforms, establishing new communications protocols and systems architectures, and the creation of new services and applications. At the level of technology, they are summarized in two: (i) an adequate infrastructure for mobility (including sensor and communication infrastructure), and (ii), an adequate information processing infrastructure that includes data management and analysis capacity (e.g., software, tools, models) to extract and process relevant and timely mobility intelligence.

Data science (DS) is an emerging field, in industry and academia, which groups multiple perspectives. Although the terms Big Data and DS are often used interchangeably, the two concepts have fundamentally different roles to play. While Big Data refers to collection and management of large amounts of varied datasets from diverse sources, DS looks to creating models and providing tools, techniques and scientific approaches to capture the underlying patterns and trends embedded in these datasets, mainly for the purposes of strategic decision making.

CVs generate huge volume of data at very high frequencies from a variety of sources, on a real time-basis. From a single source, a typical example is the *basic security message*, that is emitted 10 times per second and forms the basic data stream which other vehicles analyse to determine when a potential conflict exists. This huge volume of real-time data must be communicated, aggregated, analysed and managed. Cloud datacentres imply a large average separation between the CVs and their clouds, lack of context awareness and increasing the average network latency. New paradigms have emerged to bring the cloud services and resources closer to the user proximity by leveraging the available resources in the edge networks. These new paradigms and the data stream processing suppose standard DS must face a new context.

Some authors claim that current Artificial Intelligence concept is not suitable to manage the challenges IoV will bring in a near future. In fact, the beginning of the Ambient Intelligence (Carbone et al., 2016) as an upgraded level of such concept could emerge in coming years to face the new IoV ecosystem. The application of DS to CVs demands a new focus on how we capture, process, and use data in pervasive environments. This new field of research has been called pervasive DS (Davies and Clinch, 2017) which exists at the intersection of pervasive computing and DS, characterized by a focus on the collection, analysis (inference) and use of data in pursuit of the vision of ubiquitous computing.

The next sections aim to provide a general overview of the application of DS to CVs. They are organized as follow. The contributions to DS to CVs are presented in Section 2. The properties of the CVs data are given in Section 3, and in Section 4, the data streams, as the basic representation of CVs data, are analysed. The Knowledge Discovery Databases

(KDD) in data streams is described in Section 5, and Section 6 presents some basic machine learning algorithms. Finally, in Sections 7 and 8, the concepts of Data lifecycle and Visual analytics are briefly commented, respectively.

2 Applications of Data science to Connected vehicles

The impacts and potential operational benefits of the application of DS to CVs may be classified on mobility of people and goods, safety, environmental benefits and driver support. The following aspects may become feasible by the intelligence derived from DS. Most of them are based on descriptive, diagnostic or predictive analytics:

— **Mobility**

- Detect and understand patterns and trends in mobility data.
- Adjusting traffic signals for dynamically managing transit operations, or for emergency routing.
- Reduce main city roads' congestion (and therefore air pollution[1]) by predicting traffic flow.
- Traffic management during planned or unplanned events.
- Provide shortest or alternative routes (re-routing) between origin-destination pairs considering different factors (e.g., distance, time, energy consumption, air pollution).
- Carpooling recommendations.

— **Safety**

- Variable speed limit systems for ensuring traffic safety.
- Driver behaviour and performance analysis, to detect improper driving events by different causes (e.g. various environmental, vehicle and roadway, traffic and physical and psychological conditions).
- Assist the driver in optimum CV operation, and therefore, increasing resource economy and vehicle lifetime (also in different weather scenarios).
- Detect critical elements of the CV.
- Predict the effect of environmental conditions sensed by the CV on vehicle occupants (especially to those with known chronic affections).
- Infer real-time environmental conditions according to CVs collected data.
- Lane-changing assistance.
- Identification of drivable road surface and road boundaries.
- Understand interactions between drivers and pedestrian at signalized intersections.

— **Environmental benefits**

- Reduce supply chain waste and air pollution by associating deliveries and optimizing shipping movements.

— **Support**

- Cooperative adaptive cruise control.
- En-route guidance to parking spaces.
- Ability of CVs users to find location-based information of interest.
- Driver behaviour analysis (e.g., in the insurance domain, for calculating a safety score for the driver: pay-how-you-drive instead of insurance premiums based on population groups).

[1] After deploying intelligent decision systems based on DS and smart transportation technologies across the city of Pittsburgh (Pennsylvania, USA), Traffic21 pilot project reduced traffic jams and waiting times resulting in emissions dropped by over 20 %.

- Qualitative assessment of text-based content from traffic information dissemination in social media (e.g., drivers posting on events affecting traffic conditions) or others.
- Vehicle predictive maintenance.

3 Properties of the Connected vehicle data

The characterization of data in a CV environment may be done from different points of view. According to the data source, the origin of data can be from roadways, vehicle-based data, traveller-based data or wide area data (photogrammetry). MDOT (2014) distinguishes between digitally generated, passively produced, automatically collected, geographically or temporally trackable, or continuously analysed data.

In this section, the selected properties of the CVs data will be used to provide a first insight of the application of DS to CVs technology, and pose some of the challenges this application is facing. The first three properties are inspired from the V's defining big data. While some of the properties might be considered present everywhere, others are not general and depend on the context of the monitored phenomena; besides, some of the given properties are closely functional-related each other. Below is a summary of these properties:

1. The high *Volume* of data transmitted by a CV is determined by its grade of the connectivity and how this is implemented. This volume cannot be accurately calculated since the connectivity may work at the level of the vehicle, the transportation system, or both. A fully CV should support the interactions with their internal and external environments. The amount of data generated by a CV has been estimated upwards of 560 GB/day (Google, 2017).

   At the vehicular-internal level, the number of sensors is projected to reach the number of 200 per vehicle by 2020 (ASEE, 2017). The sampling rate and the density of the sensor network deployed in a CV will determine the amount of monitored data which will be generated. In real-time mobility data, to sample an accelerometer sensor at the rate of 100 times per second (Hz) may generate about 5MB/h of data (Kargupta, 2017).

   Other example is GPS data. The minimum size of a GPS record is 20 bytes (2 8-byte values of type double for latitude and longitude, and 1 4-byte value for time stamp), and data are collected at most once every 10 second. For a city with 20000 GPS-enabled ITS devices, daily 3219 GB of data can be generated which it is needed further be processed to extract relevant localization information (Khan et al., 2017).

   On a vehicle-external level, the amount of data a CV may transmit to other vehicle (V2V) or to infrastructure (V2I) depends on many factors, including the amount of information transmitted, its temporal resolution and the span of interaction. Wireless access technologies can provide services to both V2V and V2I up to 1 km and supports data rate up to 27 MB/sec. Haroun et al. (2016) estimated in 5 KB/sec a feasible amount of data in V2I communications.

2. *Velocity* refers to the data acquisition rate, with an emphasis on real-time (streaming) analytics. Based on the context, the transmitted data may be static or dynamic and change over time. The first are also referred as *data at rest*, and the latter, data in *motion* or *in transit*.

   Given that most of the data are acquired in real time and field settings, several issues arise about missing values and noise. The concepts of *snapshot* vs *continuous* or *send* vs *receive* are useful to explain velocity on mobile data communications. One issue related to velocity in CVs is the data latency or the time between when the data are collected and the time data can be shared with others.

The Society of Automotive Engineers (SAE) has classified the automotive applications into Classes A, B, and C with increasing order of criticality on real-time and reliability constraints (Wang et al., 2017):

- Class A, low speed (<10 kbps) for convenience features (e.g. door locks, opening/closing windows, seat position motors, occupancy control).
- Class B, medium speed (between 10 kbps and 125 kbps) for general information transfer, such as emission data and instrumentation, and typically supports the vast majority of non-diagnostic, noncritical communications.
- Class C, high speed (>125 kbps) for real-time control, such as suspension, traction, engine, brake and transmission controls. Higher performance communication classification in the range of 1 Mbps to 10 Mbps is expected in the future devoted to multimedia data.

3. *Variety* refers to the fact data are acquired from a diversity of sources. Data generated on-board (vehicle data) or off-board (from transport infrastructures, other CVs or society) are dissimilar in formats (image, audios, server logs, etc.), types (structured, semi-structured, unstructured or all the previous mixed), scales (sizes) and obtained at varied frequencies. Most of types of data in connected transport systems are of unstructured nature. Integrating these multiple data representation into useful, homogenous and structured data sets is a challenge on its own.
4. *Distributed.* There is no universal way to retrieve and transform the data automatically and universally into a unified data source for useful analysis. In a full-scale and mature IoV scenario, centralized real-time processing of large and heterogeneous set of data streams is not feasible. This motivates the need for alternative edge paradigms that aim to bring cloud-computing capabilities and services closer geographically to the CVs sensors (the edge of the network) than to the clouds. Any device with computing, storage, control and network connectivity services can be an element of the edge network, including CVs and infrastructure by the roads. Edge computing enables each edge device to play its own role of determining what information should be stored or processed locally and what needs to be sent to the cloud for further use. This technology provides a trustworthy and geographically localized application environment that is aware of the true network conditions of different CVs equipment in its proximity (network context awareness).
5. Data generated in CVs exhibit either *temporal correlation, spatial correlation* or *both*, since they may be obtained at different time points and positions. In the time series framework, CVs data exhibit Markovian behaviour, since the sensor value at a timestamp is only function of the previous sensor value at the previous timestamp. Correlation in time series must be considered when the aim is not describe the state of the system at a specific point in time. In the spatial context, CVs characterize for generating data with a position, dimension and orientation. To be able to spatially analyse data from the CVs for realizing time-spatial decision-making, the data has to be geo-referenced to create *spatial big data*.

   Spatial big data poses statistical and computational challenges due to spatial characteristics, including spatial autocorrelation, anisotropy, heterogeneity, and multi-scale and resolutions. In spatial statistics, the spatial dependence is called spatial autocorrelation, in that near things are more related than distant things. DS techniques that ignore spatial autocorrelation mistakenly assume independent and identical distribution (i.i.d) of data and therefore often generate inaccurate hypothesis or models. The spatial anisotropy extents the concept of spatial dependence considering this dependence varies across different directions (not isotropic). This is often due to irregular geographical terrains, topographic features and political boundaries. The heterogeneity in spatial statistics takes into account that spatial data samples do not follow an identical distribution across the entire space. This causes that a global models learned from samples in the entire study area may not be effective in different local regions. The last challenges in spatial big

data are that data often exists in multiple spatial scales (e.g., local, regional, global) and resolutions (e.g., sub-meter to kilometres), but also, the temporal coverage of these data can be different.

A stream is a sequence of data elements ordered by time. The structure of a stream could consist of discrete signals, event logs, or any combination of time series data. In terms of representation, a data stream has an explicit timestamp associated with each element, which serves as a measurement of data order. This type of data, due to its importance in the context of CVs, is presented next.

4 Data streams in Connected vehicles

Most machine learning and data mining approaches assume that examples (records, cases, observations or instances) are independent, identically distributed and generated from a stationary distribution. In that context, standard data mining techniques use finite training sets and generate static models. Although further discussed later, sensor networks may be geographically distributed and produce high-speed data streams. They act in dynamic environments and are influenced by adverse conditions. Data collected from a sensor network are correlated and the distribution is non-stationary. Therefore, models should change with time (Ganguly et al., 2009). From a data mining perspective, sensor network problems are characterized by a large number of variables (sensors), producing a continuous flow of data, in a dynamic non-stationary environment. In streaming scenarios, recent data are generally considered more important than older data (this conception is later discussed in Section 7, Data Lifecycle). This is because the data generating process may change over time, and the older data are often considered obsolete from the perspective of analytical insights (Aggarwal, 2015).

The main characteristics of data stream are now discussed. The differences between data stream and data stream in sensor networks are afterwards commented. However, in these sections, since data streams from sensor networks are also a representation of data streams, these terms will be used interchangeably for the sake of simplicity.

4.1 Characteristic of data streams

A data stream can be read only once, or rarely a small number of times, using limited computing and storage capabilities. In the data stream model the data arrive (transiently and automatically) at high speed so the algorithms used for mining the data streams must process them with constraints of space and time ("resource aware algorithms"). Some other relevant differences between traditional data and data streams are shown in Table 1 and include:

- Data elements in the stream arrive online, in a multiple, continuous, time-varying manner.
- Data stream is time ordered data, either explicit with a time stamp or implicit based on arrival order.
- The system has no control over the order in which data elements arrive, either within a data stream or across data streams.
- Changing probability distributions of the data instead of i.i.d data.
- Data streams are potentially unbounded in size.
- Once an element from a data stream has been processed it is discarded or archived. It cannot be retrieved easily unless it is explicitly stored in memory, which is small relative to the size of the data streams.
- The algorithm processing the stream must update its model incrementally as each data is inspected. An additional desirable property is the "anytime property", by which it is required that the model is ready to be applied at any point between data (real-time response).

Table 1. Comparison between traditional and stream data processing.

Characteristics	**Traditional data mining**	**Data stream mining**
Number of passes	Multiple	Single
Processing time	Unlimited	Restricted
Memory usage	Unlimited	Restricted

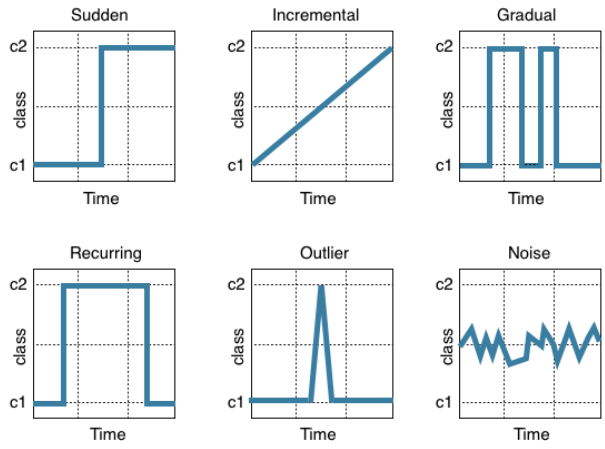
Distributed	No	Yes
Data size	Finite data set	Continuous flow
Data distribution	i.i.d	Non-i.i.d
Data evolution	Static	Non-stationary
Order of observations	Independent	Dependent
Model stability	Static	Evolving
Data labelling	Non-costly	Costly
Type of results	Accurate	Approximate

Source: Own elaboration (adapted from Gama, 2013, Nguyen et al., 2015, and Krawcyk et al., 2017).

The changes in the physical world are reflected in terms of the changes in data or model built from data. Change detection in stream data aims to identifying differences in the data by observing it at different times and/or different locations in space. The ability to identify trends, patterns, and changes in the underlying process generating data contributes to the success of processing and mining massive high-speed data streams.

In recent years many change detection methods have been proposed for streaming data. Machine Learning algorithms learn from observations described by a finite set of attributes. In real world problems, there can be important properties of the domain that are not observed. There could be hidden variables that influence the behaviour (properties) of nature. The reasons of change in streaming data is diverse and abundant, although it is identified two dimensions for analysis, namely, the *causes of change*, and the *rate of change*. The causes of change are influenced by modifications in the context of learning (because of changes over time in hidden variables), but also, from changes in the characteristic properties of the observed variables. As a result, concepts (target variables) learned at one time can become inaccurate. The second dimension is related to the rate of change. The term *Concept Drift* is more associated to gradual changes in the target concept, while the term *Concept Shift* refers to abrupt changes. Usually, detection of abrupt changes is easier and requires few examples for detection. Gradual changes are more difficult to detect. The latter is one of the biggest challenges for data stream mining, as the data are dynamic and depend on many factors that can keep changing fast. Figure 1 shows, adopting a two-classes classification view, six basic structural types of changes that may occur over time:

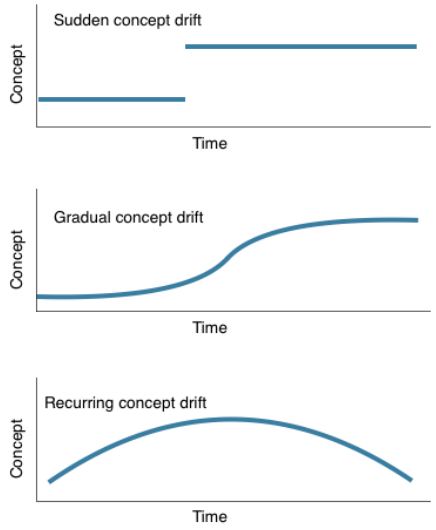
Figure 1. Types of drift.



Source: Own elaboration (adapted from Ramírez-Gallego et al., 2017).

Some of these changes, in a continuous approach, are reproduced in Figure 2:

Figure 2. Types of drift in a continuous scheme.



Source: Own elaboration (adapted from Hoens et al., 2012).

By adding some notation, data streams can be considered infinite sequences of (x,y) pairs, where x if the feature vector and y the class label. Zhang et al. (2008) observed that $p(x|y) = p(x|t)\ p(y|x)$, with t a given time stamp, and categorized concept drift in two types:

— *Loose concept drifting* when concept drift is caused only by the change of the class prior probability $p(y|x)$ as time elapse.

— *Rigorous concept drifting* when concept drift is caused by the change of class prior probability $p(y|x)$ and the conditional probability $p(x|t)$.

In Table 1, the type of results in data stream mining is defined as approximate. With new data constantly arriving even as old data are being processed, the amount of computation time per data element must be low. Furthermore, since the amount of memory is bounded, it may not be possible to produce exact answers. High-quality approximate answers can be an acceptable solution.

All these data stream characteristics pose the need for other algorithms than ones previously developed for *batch learning*, where data are stored in finite, persistent data repositories. In non-stationary data streams, data from the past can become irrelevant or even harmful for current situations, deteriorating predictions of the algorithms used. Data management approaches can play the role of a forgetting mechanism when old data instances are discarded.

4.2 Data Stream in sensor networks

A sensor network consists of spatially distributed autonomous sensors that cooperatively monitor an environment. Each of these computational devices may be equipped with sensing, processing and communication facilities. The processing part is able to do computation on the sensed values and/or other received values from the neighbours. The communication part is able to listen and send to other sensor nodes.

The distinction between traditional data stream processing and sensory data stream processing is important because sensory data streams have their own features. Elnahrawy (2003) distinguished sensor streaming from traditional streaming in the following way:

— The sensor data are a sample of the entire population. On the other hand, traditional streaming data represent the entire population of the data.

— The sensor data streams are considered noisy and faulty by comparison with other traditional streaming data.

— The sizes of sensor data streams are usually less than traditional streaming data.

Other set of differences can be addressed considering the intrinsic nature of sensor data:

— Spatial, temporal, and spatio-temporal attributes play a major role in sensor networks. Sensor data are usually meaningless unless it is associated with the time and location of the information.

— Redundancy in sensor networks is common due to the strong spatial and temporal correlations.

— Existence of missing data (absent readings). Redundancy can be used to impute missing values.

— Real-time data cleaning is required to build an accurate model.

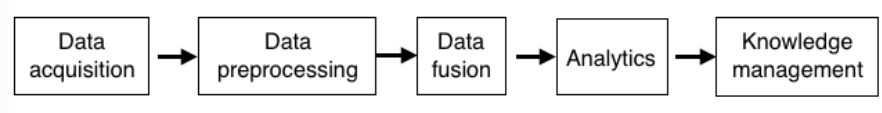
— Sensor nodes have limited availability of computational resources.

Information extraction methods that translate raw output from sensors are as numerous as the sensors themselves. In most cases, raw information from sensors has little meaningful end-use value and needs to be processed in various ways for different purposes. The term KDD became popular at the first KDD workshop in 1995 in Montreal (Fayyad et al., 1996) and refers to the overall process of discovering useful knowledge from data. The application of KDD to streaming data, as those produced in CVs, is described next.

5 Knowledge discovery from data streams

The data mining process for classical knowledge discovery is commonly partitioned into several stages. In the data stream context, the KDD process must be thought anew to process data from sensor networks in (near) real time (Figure 3, adapted from Rehman et al., 2017). These new KDD systems are constrained by three main limited sources: time, memory and sample size (Domingos and Hulten, 2000). These constraints have resulted in the development of different kinds of windowing techniques, sampling and other summarization approaches. But also, in the development of several algorithms for data mining which are modified versions of clustering, regression, and anomaly detection techniques from the field of multidimensional data series analysis in other scientific fields (Appice et al., 2014).

Figure 3. KDD in data streams.



Source: Own elaboration (adapted from Rehman et al., 2017).

Ideally, we would like to have KDD systems that operate continuously and indefinitely, incorporating examples as they arrive, and never losing potentially valuable information. Such desiderata are fulfilled by incremental learning (*aka* online, successive or sequential methods), on which a substantial literature exists.

The new KDD stages are not completely disjoint from each other, and there are techniques that perform functions common to different stages. Additionally, there are also interactions between the techniques used in the different stages that can affect end-to-end results. A simplified KDD approach for data streams is described next in a stepwise fashion (modified from Rehman et al., 2017):

1. *Data acquisition and selection*, and in particular *stream processing***,** are challenging tasks because of massive heterogeneity. Once stream data are obtained from multiple sensors in a CV, complexity reduction is a mandatory step. This reduction tries to maintain the original structure and meaning of the inputs, but at the same time obtaining a much more manageable size. Faster training and improved generalization capabilities of learning algorithms, as well as better understanding and interpretability of results, are among the many benefits of data reduction (Ramírez-Gallego et al., 2017). This approach requires controlling a trade-off between accuracy and the amount of memory used to store the reduced data.

   Several techniques have been developed for storing *summaries* or *synopsis information* about previously seen data (e.g. sketching algorithms, element-counting data structures). Quite often, the interest is to compute some statistical property of the streaming data. The recursive version of the sample mean, or the incremental version of the standard deviation can be used, to cite a few, to obtain simple statistics of data streams. Also, *Hoeffding bounds* provide confidence bounds on the mean of a distribution (given enough independent observations, the true mean of a random variable will not differ from the estimated mean by more than a certain amount). The main interest of these statistics is that they allow maintaining their exact values over an eventually infinite sequence of data without storing them in memory. Many different kinds of synopsis can be constructed depending upon the application at hand. The nature of the synopsis highly influences the type of insights that can be mined from it.

   Sampling at periodic intervals a data stream is a common technique for reducing its data. To obtain an unbiased sampling is basic in data streams. In statistics, most sampling techniques require to know the length of the stream. The key problems are

the sample size and sampling method. The *reservoir sampling* technique is the classic algorithm to maintain an online random sample. The basic idea consists of maintaining a sample of fixed size, called the *reservoir*. As the stream flows, every new element has a certain probability of replacing an old element in the reservoir. Other sampling strategies can be found in literature (e.g., Min-Wise sampling, load shedding).

Most of the time, there is no interest in computing statistics over all the past, but only over the recent past in the data stream. *Data windows* are a way of looking at relevant slices of a data stream. Windowing models can be used to limit the amount of processed data based on different characteristics. Some of most common of these models are the *landmark*, *tilted*, *sliding* and *damped windows*. The simplest approach is the sliding windows, based on *first in*, *first out* data structure.

(a) Sliding Window. Given a window with width w and current time point t, the interest is in the frequent patterns occurring in the window $[t - w + 1, t]$. As time advances, the window will keep its width and move along with the current time point. In this model, there are no interest in the data that arrived before the time point $t - w + 1$.

(b) Landmark Window identifies relevant points (the *landmark*) in the data stream and the aggregate operator uses all records seen so far after the landmark.

(c) Time-Tilted Window. In this model, the interest is in frequent item sets over a set of windows of varying width. Each window corresponds to different time granularity based on their novelty.

(d) Damped Window Model. This model assigns greater weight to more recently arrived transactions. A simple way to do that is to define a *decay rate* δ, $0 < \delta \leq 1$. As each new data transaction arrives, the support levels of the previously recorded patterns are multiplied by δ to reduce their significance.

2. *Data pre-processing* is the stage that includes operations to prepares the data for further analysis and improve the quality of the data stream. The heterogeneity in pre-processing operations arise when mobile data stream mining methods need to handle missing values, remove noise, and detect anomalies and outliers from the data stream (Rehman et al., 2017).

(a) Noise filtration. Noise refers to the inclusion of extraneous and irrelevant information in mobile data streams. The data streams become noisy due to multiple reasons such as improper placement of sensors, wrong sensor configurations, and inducement of environmental noise among others.

(b) Outliers detection. Outliers (in an univariate or multivariate scheme) refer to misreported data points in the acquired data streams. Numerous classification and clustering methods are used to detect and remove the outliers.

(c) Anomaly detection. Anomaly detection refers to the presence of anomalous data points in acquired data streams. The anomaly detection helps in improving the quality of knowledge patterns.

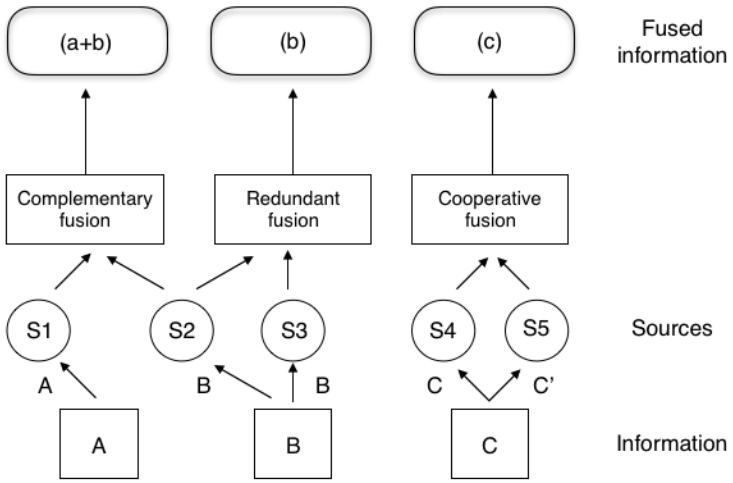
(d) Feature extraction. Massive data streams need to handle efficiently. The feature extraction methods help in extracting features (attributes) from incoming data. Feature extraction methods convert data streams from unstructured and semi-structured formats into structured data formats.

(e) Sparsity handling. Highly sparse data may hamper the performance of far-edge mobile devices in some cases. Similarly, low sparsity also degrades the performance of data stream mining methods. Therefore, handling sparsity and maintaining an adequate level sparsity in data stream mining applications help in improving the quality of knowledge patterns.

A survey on data pre-processing techniques for data stream mining can be found in Ramírez-Gallego et al. (2017).

3. *Data fusion* is required since CVs sensors generate data streams with different sampling frequencies. Besides, sensor or data fusion is a principled way of combining data from multiple sensors to yield better results than any single sensor could produce on its own (Castanedo, 2013). In general, a sensor model will include an error distribution which is essentially a formalization of how good the sensor is. Several criteria to classify data fusion techniques are possible. Based on the relations of the input data sources, the following data fusion techniques were proposed by Durrant-Whyte (1988) (Figure 4):

Figure 4. Data fusion techniques.



Source: Own elaboration (adapted from Castanedo, 2013).

Other terms associated with data fusion that typically appear in the literature include decision fusion, data combination, data aggregation, multi-sensor data fusion, and sensor fusion. In this sense, the term information fusion implies a higher semantic level than data fusion.

4. *Modelling* (or *learning*) is the stage that includes knowledge discovery algorithms and is regarded as one of the main steps in KDD from data streams. The heterogeneity in learning arises in terms of learning types and learning models. The learning type varies in terms of supervised, unsupervised and semi-supervised models, and learning models are summarized in Section 6. Domingos and Hulten (2001) identify desirable properties of learning systems for efficient mining continuous, high-volume, open-ended data streams:

   (a) Require small constant time per data example.

   (b) Use fix amount of main memory, irrespective to the total number of examples.

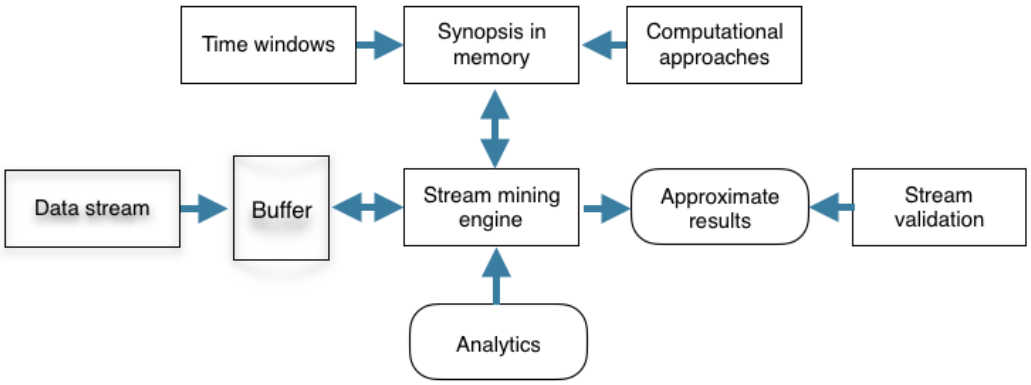
   (c) Built a decision model using a single scan (pass) over the training data. Ideally, it should produce a model that is equivalent (or nearly identical) to the one that would be obtained by the corresponding ordinary database mining algorithm.

   (d) Generating an *anytime model* independent from the order of the examples.

   (e) Ability to deal with concept drift. The model, at any time, should be up-to-date, but also include all information from the past that has not become outdated.

   Learning systems process examples at the rate they arrive using a single scan of data and fixed memory. They maintain a decision model at any time and are able to adapt the model to the most recent data. In Figure 5 is adapted the general model for data stream mining proposed by Nguyen et al. (2015):

Figure 5. A general model for data stream mining in CVs.



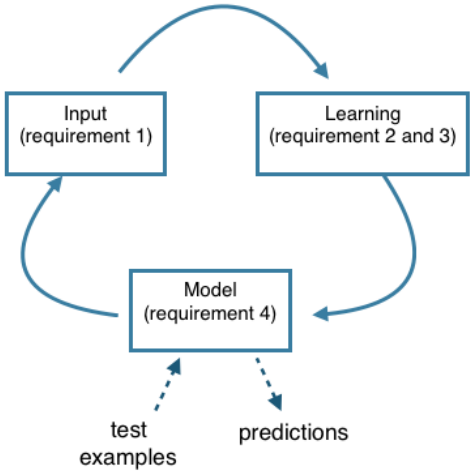
Source: Own elaboration (adapted from Nguyen et al., 2015).

When a data stream arrives, a buffer is used to store the most recent data. The stream mining engine reads the buffer to create a synopsis of the data in memory. In order to maintain the synopsis, the system may apply different time window and computational approaches. When certain criteria are triggered, for example, a user's request or after a certain time lapse, the stream mining engine will process the synopsis and output approximate results. In general, most data stream algorithms are derived and adapted from traditional mining algorithms. Lastly, stream validation methods (later described) are applied to evaluate the performance of data stream algorithms.

Broadly speaking, there are two *computational approaches* to process the data streams: *incremental learnings* and *two phases learning* (*aka* online-offline learning):

- In the incremental learning the model evolves to adapt to changes in incoming data: the learning process takes place whenever new examples emerge, and then adjusts to what has been learned from the new examples. There are two schemes to update the model: by data instance and by window. It has the advantage of providing mining results instantly, but requires more computational resources. Figure 6 illustrates the incremental learning approach following three steps in a repeating cycle, adapted from Bifet and Kirby (2009): (i) the algorithm is passed the next available example from the stream (requirement 1), (ii) the algorithm processes the example, updating its data structures -it does so without exceeding the memory bounds set on it (requirement 2), and as quickly as possible (requirement 3)-, and (iii) the algorithm is ready to accept the next example. On request it is able to supply a model that can be used to generate looking-ahead predictions (requirement 4).

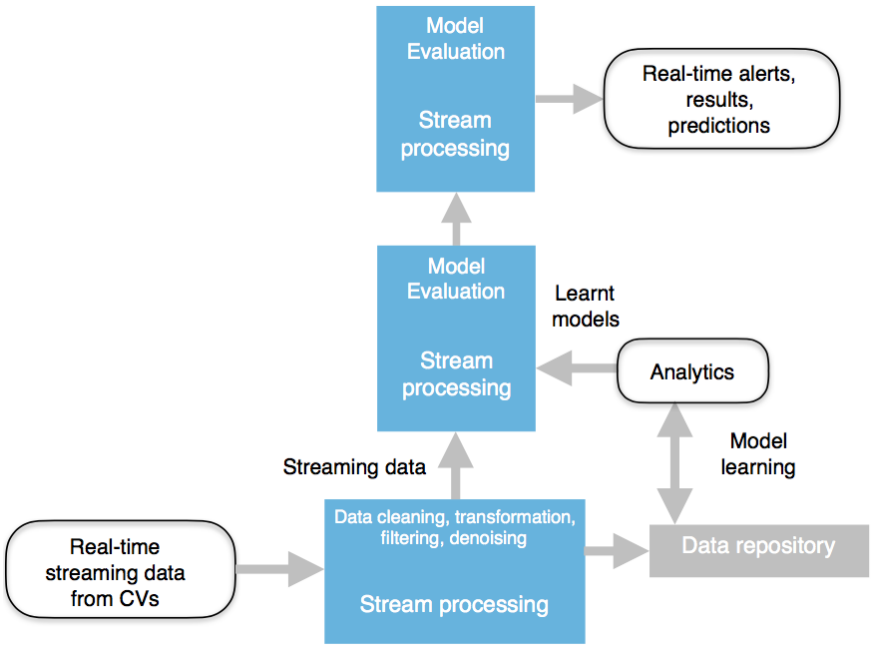
Figure 6. Incremental learning.



Source: Own elaboration (adapted from Bifet and Kirby, 2009).

- Two-phase learning is a common computational approach in data streams. In the first phase (online phase), a synopsis of data is updated in a real-time manner. In the second phase (offline phase), the mining process is performed on the stored synopsis whenever a user sends a request.

Figure 7. Online-offline learning for continuous data analysis



Source: Own elaboration (adapted from Andrade et al., 2014).

Figure 7 illustrates how offline model learning and online scoring can be performed. In this figure, it is shown historical data being processed offline by traditional data mining tools (as proposed by these authors) producing the data mining model. Subsequently the model is imported to score the new, incoming data. In this diagram, it is also seen that (some of the) newly arrived data is stored, thereby allowing the periodic recreation of the model by re-running the modelling step.

In the data stream mining literature, most algorithms incorporate one or more of the following ingredients: windows to remember recent examples; methods for detecting distribution change in the input, and methods for keeping updated estimations for some statistics of the input. Either both in the incremental learning or two-phases learning approaches, it is important to set (i) what data to remember or forget, (ii) when to do the model upgrade, and (iii) how to do the model upgrade.

The competing motivations of these goals give rise to the *stability–plasticity dilemma*, which asks how a learning system can be designed to remain stable and unchanged to irrelevant events (e.g., outliers), while plastic (i.e., adaptive) to new, important data (e.g., changes in concepts) (Hoens et al., 2012).

5. *Knowledge management*. This stage includes consolidating discovered knowledge and incorporating this knowledge to another system for further action. Integration, storage, and utilization of knowledge patterns in mobile data stream applications take place at various places (Rehman et al., 2017):

   (a) On-device. The on-board storage refers to the storage capabilities of far-edge devices that are used to store locally uncovered knowledge patterns. In addition, the synchronized knowledge patterns for personalized user experience are also stored on-board far-edge mobile devices.

   (b) On-edge. The service provision from edge servers enables data reduction. The location-aware aggregations of knowledge patterns facilitate in reduced data transfer in remote environments and minimize bandwidth utilization.

   (c) Remote. Conventionally knowledge patterns are integrated and stored in remote data stores, which include cloud data centre, clusters, grids, and application servers. Remote knowledge aggregation is useful for global knowledge discovery.

6. *Evaluation* is the stage that includes operations for the use of a mining model and for the interpretation of the results produced by the modelling process. The two mains aspects that make the difference with respect the evaluation in batch learning, in which the learning process is made from a single batch of examples, are the continuous evolution of decision models and the non-stationary nature of data streams, and therefore, the evaluation based on large test or validation streaming data sets could not be viable. Several approaches have been proposed to deal with this problem, and currently, evaluation of algorithms in streaming data represents an active field of research. Most of the evaluation methods and metrics were designed for the static case and provide a single measurement about the quality of the applied models. Many methods and metrics are possible and can be found in literature to perform evaluation on classification and regression tasks in streaming data (Muthukrishnan, 2005). Other considerations to evaluate algorithms in a data stream context are not just based on the accuracy of their predictions, but also, on their memory requirements and runtime.

6 Basic learning algorithms

There is a wide array of applications to which machine learning in stream data can be applied. Broadly speaking, in all these applications it can be identified several types of learning-related issues, which are:

- Classification – to assign each item from a data set to a specific category.
- Regression and time series analysis – to predict a real value for each item.
- Ranking – to return an ordered set of features based on some user-defined criterion.
- Dimensionality reduction – to use for transforming initial large feature spaces into a lower-dimensional representation so that it preserves the properties of the initial representation.
- Clustering – to group items based on some predefined distance measure.
- Anomaly detection – to conduct observation or series of observations which do not resemble any pattern or data item in a data set.

One crucial question is how to select an appropriate machine learning algorithm in the context of CVs. The answer indeed depends on the size, quality, the nature of the data and the intended use of the solution. Although in Section 5.4 were described the desirable properties of a learning systems for streaming data, it is worth to add the following factors to be considered when choosing the right learning algorithm (Akerkar and Sajja, 2016):

- Accuracy. Whether obtaining the best score is the aim or an approximate solution while trading off overfitting.
- Training time. The amount of time available to train the model. Some algorithms are more sensitive to the number of data points than others. When time is limited and the data set is large, it can drive the choice of algorithm.
- Linearity. Linear classification algorithms assume that classes can be separated by a straight line. Though these assumptions are good for certain problems, they bring accuracy down on some.
- Number of parameters. Parameters, such as error tolerance or number of iterations, or options between variants of how the algorithm performs, affect an algorithm's behaviour. The training time and accuracy of the algorithm can occasionally be rather sensitive to getting just the right settings.
- Number of features. For certain types of data, the number of features can be very large compared to the number of data points. The large number of features can overload some learning algorithms, making training speed rather slow.

Many streaming applications contain multidimensional discrete attributes with very high cardinality. In such cases, it becomes difficult to use conventional data learning algorithms because of memory limitations. It is not the purpose of this section to extensively cover the machine learning techniques and theories to derive intelligence from data streams produced in CVs beyond the very basics. A review of these techniques can be found in Gama and Gaber (2007), Gama (2010, 2013), Hoens et al. (2012), Aggarval (2015), and Anand et al. (2017). The next first three algorithms are solely devoted to classification.

- *Nearest neighbour* classifiers are defined by their characteristic of classifying unlabelled examples by assigning them the class of similar labelled examples. The nearest neighbours approach to classification is exemplified by the k-nearest neighbours algorithm (k-NN). k-NN is a simple algorithm that stores all available cases and classifies new cases by a majority vote of its *k* neighbours.
- *Bayes classification*. The Bayes classifier is based on the Bayes theorem for conditional probabilities. This theorem quantifies the conditional probability of a

random variable (class variable), given known observations about the value of another set of random variables (feature variables). It is assumed that the data points within a class are generated from a specific probability distribution such as the Bernoulli distribution or the multinomial distribution. A naïve Bayes assumption of class-conditioned feature independence is often (but not always) used to simplify the modelling. Naïve Bayes is a rare example of an algorithm that needs no adaptation to deal with data streams (Bifet and Kirby, 2009).

— *Support vector machines* (SVMs) are naturally defined for binary classification of numeric data. The binary-class problem can be generalized to the multiclass case. Categorical feature variables can also be addressed by transforming categorical attributes to binary data with the binarization approach. It is assumed that the class labels are drawn from $\{-1, 1\}$. SVMs use separating hyperplanes as the decision boundary between the two classes. In the case of SVMs, the optimization problem of determining these hyperplanes is set up with the notion of margin.

  It is worth to mention that in classification, a challenge arises when it is assumed that the prevalence of each class in the dataset is, and will remain, equivalent. While class prevalence in traditional data mining problems remains constant, such an assumption is particularly impractical in streaming data applications, where the class distributions can become severely imbalanced. Thus the positive (rare) events, which are underrepresented in a static dataset, can become even more severely underrepresented in streaming data. Hence, when combined with potential concept drift, class imbalance poses a significant challenge that needs to be addressed by any algorithm that proposes to deal with learning from streaming data.

— *Decision trees* are supervised learning methods that gather powerful and popular tools for classification and prediction. In tree-based methods the decision trees represent *rules*. In these algorithms, it is split the population into two or more homogeneous sets. This is done based on most significant attributes/independent variables to make as distinct groups as possible. *Hoeffding trees* (which borrows the Hoeffding bound) build models that can be proven equivalent to standard decision trees if the number of examples is large enough. Very fast decision trees are based on the principle of Hoeffding trees.

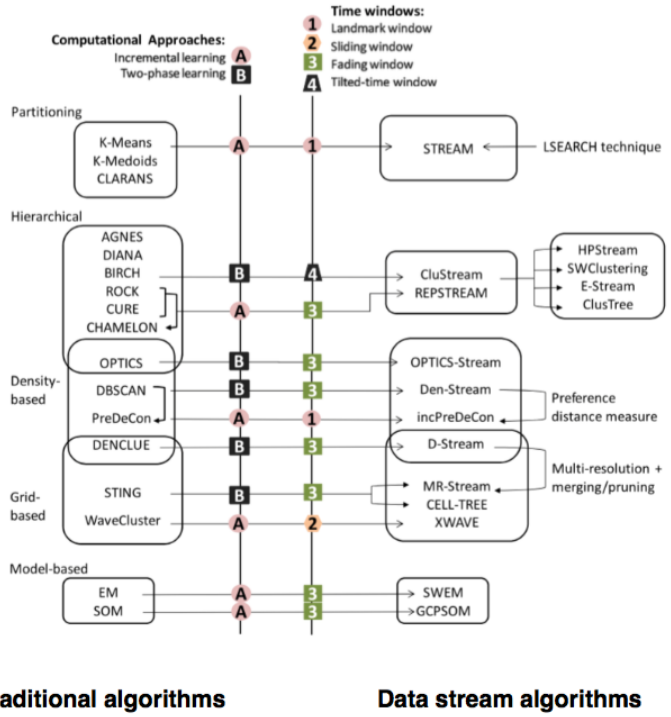
— *Linear regression*. Regression is concerned with specifying the relationship between a single numeric dependent variable (the value to be predicted) and one or more numeric independent variables (the predictors). The simplest forms of regression assume that the relationship between the independent and dependent variables follows a straight line. In the case of two or more independent variables, this is known as *multiple linear regression*. Both of these techniques assume that the dependent variable is measured on a continuous scale. Regression can also be used for other types of dependent variables and even for some classification tasks. For instance, logistic regression is used to model a binary categorical outcome, while Poisson regression models integer count data. The method known as multinomial logistic regression models a categorical outcome; thus, it can be used for classification. The same basic principles apply across all the regression methods.

— An *Artificial neural network* is a predictive model inspired by the way the brain operates (neurons). To train a neuron, along with a sample data set, the learning strategy also must be determined. That is, the neuron must know when to send an output and when not. Such a large number of neurons are interconnected through weighted connections between individual neurons. These neurons work in a very simple way, but all together, in a parallel fashion. The abilities of making intelligent decision-making and learning come from their parallel working. Each neuron calculates an output at its local level, which at the end is summed up as a global solution. Neural networks may consist of multiple layers of neurons. *Deep learning* is a technique of machine learning that consists of many hierarchical layers to process the information. This technique is its extremely computationally intensive and slow to train, particularly if the network topology is complex.

— *Random Forest* (RF) is a hallmark phrase for an ensemble of decision trees. RF is a collection of decision trees known as "Forest". RF is well suited to small sample size and large number of attributes problems. RF comes at the expense of some loss of interpretability, but obtain high performance on classification and predictions problems.

Several algorithms devote to clustering data streams. Two examples are the Stream and CluStream algorithms (Aggarwal, 2015). In both cases, they are based on the *k-medians* clustering methodology. Beringer and Hüllermeier (2006) proposed an online version of *k-means* for clustering parallel data streams (online k-means), using a Discrete Fourier Transform approximation of the original data. Clustering *On Demand* is another framework for clustering streaming series which performs one data scan for online statistics collection, designed to address the time and the space constraints in a data stream environment (Ganguly et al., 2009). In Figure 8, the relation between some traditional clustering algorithms and those used to deal with stream data is presented.

Figure 8. Relationship between traditional and stream clustering algorithms.



Source: Own elaboration (adapted from Nguyen et al., 2014).

The term *ensemble methods* refers basically to combine the output (numeric or categorical) of others learner models. All the ensemble methods are based on the idea that by combining multiple weaker learners, a stronger learner is created. A number of techniques have evolved to support ensemble learning, the best known being *bagging* (bootstrap aggregating) and *boosting*. A number of differences exist between bagging and boosting. Bagging uses a majority-voting scheme, while boosting uses weighted majority voting during decision-making. Predictions generated by boosting are independent of each other, while those of bagging are dependent on each other. *Stacking* approaches will not be discussed. A survey on ensemble learning for data stream analysis can be found in Krawcyz et al. (2017). Zang et al. (2014) performed a comparative study about incremental and ensemble learning on data streams.

7 Data lifecycle

The data lifecycle is an abstract view which describes the various processes the data goes through from its inception to end of life. In and ITS-enables technology, DS can provide adaptable and dynamic intelligence to make decisions on real time. However, making decisions based solely on data that was created in the recent past (e.g. hours) can impede such decisions because lacks the historical context needed to see and verify trends (e.g., slow-changing phenomena). In fact, old data can be more valuable than new data because it has survived the test of both time and use (Plale and Kouper, 2017). To appreciate historical and recent data as similarly valuable is not a trivial task. Knowing what data to use, and how to compare past and present measurement is essential in assessing changes in ITS, but also how to use data for accurate predictions. All these aspects require considering data and knowledge derived from them within a lifecycle (Cavanillas et al., 2016). Therefore, the KDD approach described earlier is just one stage (broadly speaking, the one related to data analysis) within the ITS data lifecycle.

Many lifecycle models exist, including domain-specific, regional and industry-specific models. Next are included four data generic lifecycle (U.S. Geological Survey –USGS-, UK Digital Curation Centre –DCC-, Higgins -2008-, DataOne and SEAD -Myers et al., 2015-) and its comparison (Plale and Kouper, 2017), indicating the stages composing each of them:

Table 2. Comparison of data lifecycle models.

Characteristics	**USGS**	**DCC**	**DataOne**	**SEAD**
Sequential stages	Plan	Conceptualize	Plan	
	Acquire	Create	Collect	Create
		Appraise & select	Assure	
	Process	Ingest	Describe	
	Analyze			Analyze
	Preserve	Preservation action	Preserve	
	Publish/share	Store	Discover	Publish
		Access, use & resuse	Integrate	
		Transform	Analyze	Reuse
Cross-cutting or complementary aspects	Metadata	Metadata	Metadata	Metadata
	Quality	Preservation	Quality	Provenance
	Security	Community	Preservation	Curation
		Curation		

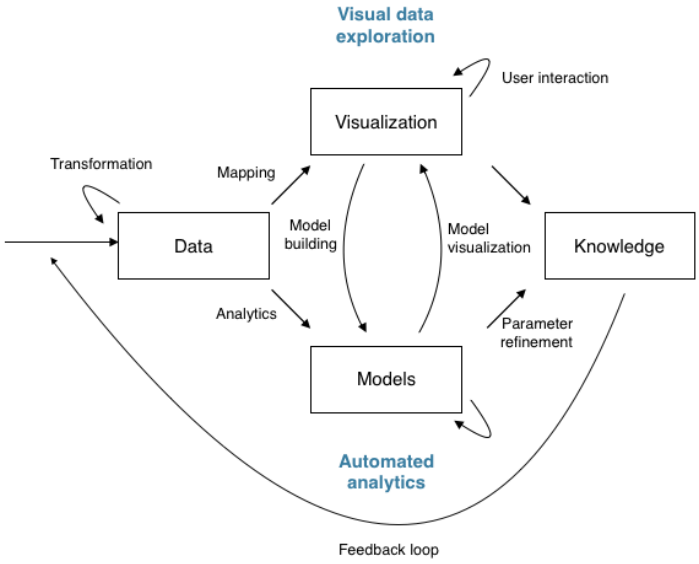
Source: Own elaboration (adapted from Plale and Kouper, 2017).

These models encompass the whole data lifecycle, but it does not require all activities to be represented in every project. Some ITS projects might use only parts of these lifecycles, although quality, assurance, description and preservation activities are crucial to any project.

8 Visual analytics

Ideally, it could be created systems that automatically discover knowledge from data using data mining algorithms that require no human input. However, the questions typically asked of data are often too exploratory for a completely automated solution and there may be trust issues (Steed, 2017). Visual Analytics (VA) is a new interdisciplinary field which stems from the field of information visualization. They are techniques that seek to support in the analysis and understanding of large datasets using statistical techniques and data mining, aided by visualization techniques and interaction techniques. The VA process adapted from Blytt (2013) is shown in Figure 9. As in the previous section, the KDD approach is one of the items that integrate the VA process.

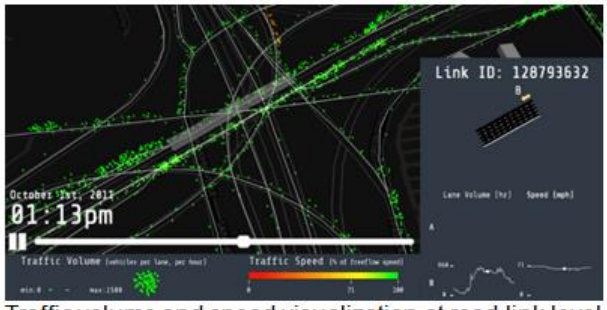
Figure 9. The visual analytic process.



Source: Own elaboration (adapted from Blytt, 2013).

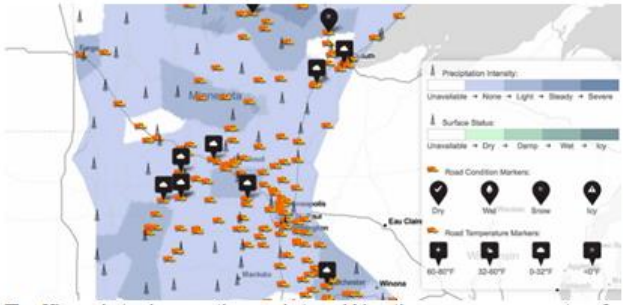
To select the right type of visual representation and the visual literacy that is required to extract value from data analytics are topics of research on their own within the field of information visualization (Yau, 2011). The US Federal Highway Administration's Research Data Environment (RDE) contains a vast amount of collected, measured, and simulated data about the highway systems of the United States. From this source, in Figure 10 it is shown six prototype visualization elements that explore the real-time interpretability of multidimensional data related to ITS and CV technologies.

Figure 10. Visualization of selected data in RDE (U.S. Department of Transportation).

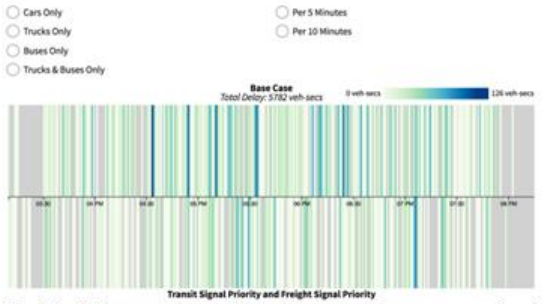


Traffic volume and speed visualization at road-link level.

Communication between CVs (basic security message) and road equipment up to a 500m radius.



Traffic-related weather data. Weather, movement of snowploughs, and road conditions are shown during a snowstorm.



Traffic interventions. Traffic signals communicate with vehicles and possible behave differently for different types of vehicles.

Geographical locations

Interactive visual overview of the Basic Safety Message, that logs vehicle's attempt to communicate with roadside equipment.

Source: Own elaborations (adapted from https://www.its-rde.net/RDE-Visualizations).

Other visualization techniques applied to CVs and other referenced material available can be found in Steed (2017).

9 Conclusions

This study aims to characterize data stream from CVs from an analytical perspective and examines the analytical lifecycle of streaming data generated in CVs by means of a selected literature analysis. But also, to detect the issues that need to be resolved for the correct implementation of DS in CV technology.

Data streams pose new challenges for machine learning and data mining as the traditional algorithmic methods have been designed for static datasets and are not capable of efficiently analysing fast growing amount of data. The next are the main limitations and characteristics that should be taken in consideration to achieve maximum potential in the application of DS to CVs:

— Data generated by a CV are characterized by a multitude of formats and data types and have therefore, a pronounced volume and variety dimensions. It must be addressed how to improve the quality of data in real time (filtering), how to summarize and sketch them, how to unify data (fusion methodologies) and processing models, how to implement knowledge creation and reasoning, and how to conduct short-term and long-term storage.

— The application of data mining techniques to streaming data generated by sensors in CV may deliver approximate results. It must be understood the assessment, representation and propagation of uncertainty, but also, the developing robust-optimization methods and design of optimal sequential decision-making.

— Algorithms that process data streams must face limited computational resources as memory and time, as well as constraints to make predictions in reasonable time.

— Since the nature of data in a CV environment is changing and evolving continuously over time, advancements on adaptive algorithms should take into account the next two key aspects:

  - The *stability-plasticity dilemma*, that asks how a learning system can be designed to remain stable and unchanged to irrelevant events, while plastic to new, important data in the CV environment.
  - The phenomenon called *concept drift* is related to changes in the distribution of data, which occur in the streams over time. This concept might deteriorate the performance of built models.

— Features previously considered irrelevant might become relevant, and vice-versa, to reflect the dynamic of the process generating data.

— Although there are an increasing number of streaming learning algorithms, the metrics and the design of experiments for assessing the quality of learning models is still an open issue. Discussions on best practices for performance assessment and differences in performance when learning dynamic models that evolve over time should be addressed.

Mining pervasive data streams requires new and efficient algorithms executed in dynamic and changing environments under time and memory constraints. Also, the selection of the data pre-processing methods, data reduction and data fusion operations becomes a challenge due to the variant complexity of the algorithms involved. The faster training and improved generalization capabilities in the learning algorithms are among the issues that need to be resolved for the correct implementation of DS in CV technology.

References

Aggarwal, C.C., *Data mining: The Textbook*, Springer, New York, 2015.

Akerkar, R., Sajja, P.S., *Intelligent techniques for Data Science,* Springer, New York, 2016.

Anand, S., Padmanabham, P., Govardhan, A., Kulkarni, R.H.,`An extensive review of data mining methods and clustering models for Intelligent Transportation Systems´, *Journal of Intelligent Systems*, 2017, pp. 1-11.

Andrade, H.C.M., Gedik, B., Turaga, D.S., *Fundamentals of Stream Processing: Application Design, Systems and Analytics*, Cambridge University Press, Cambridge, 2014.

Appice, A., Ciampi, A., Fumarola, F., Malerba, D., *Data Mining Techniques in Sensor Networks: Summarization, Interpolation and Surveillance*, Springer, London, 2014.

ASEE (Automotive sensors and Electronic Expo 2017) website, http://www.automotivesensors2017.com, last accessed 26 September 2017.

Beringer, J., Hüllermeier, E., `Online clustering of parallel data streams´, *Data & Knowledge Engineering*, Vol. 58, No 2, 2006, pp. 180-204.

Bifet, A., Kirkby, R., *Data stream mining. A practical approach*, Centre for Open Software Innovation. The University of Waikato, Hamilton, New Zealand, 2009.

Blytt, M., 2013, *Big challenges for visual analytics. Assisting sensemaking of big data with visual analytics*, retrieved on 26 September 2017 from https://www.ntnu.no/documents/10401/1264433962/MikkelArtikkel.pdf/.

Carbone, A., Jensen, M., Sato, A.H., `Challenges in data science: a complex systems perspective´, *Chaos, Solitions & Fractals*, Vol. 90, 2016, pp. 1-7.

Castanedo, F., `A review of data fusion techniques´, *The Scientific World Journal*, Vol. 2013, 2013.

Cavanillas, J.M., Curry, E., Wahlster, W., *New Horizons for a data-driven economy. A roadmap for usage and exploitation of Big Data in Europe*, Springer, New York, 2016.

DataOne (Data Onemodel), https://www.dataone.org/, last accessed 29 September 2017.

Davies, N., Clinch, S., `Pervasive data science´, *IEEE Pervasive Computing*, Vol. 16, No 3, 2017, pp. 50-58.

DCC Curation Lifecycle Model, http://www.dcc.ac.uk/resources/curation-lifecycle-model, last accessed 29 September, 2017.

Domingos, P., Hulten, G., 2000, `Mining high-speed data streams´, *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 71-80.

Domingos, P., Hulten, G., 2001, `Catching up with the data: research issues in mining data streams´, *Proceedings of Workshop on Research Issues in Data Mining and Knowledge Discovery*.

Durrant-Whyte, H.F., `Sensor models and multisensor integration´, *International Journal of Robotics Research*, Vol. 7, No.6, 1988, pp. 97–113.

Elnahrawy, E., Research directions in sensor data streams: solutions and challenges, In: DCIS Technical Report DCIS-TR-527, Rutgers University, New Jersey, 2003.

European Union, 2010, *Directive 2010/40/EU of the European Parliament and of the Council of 7 July 2010 on the framework for the deployment of Intelligent Transportation Systems in the field of road transport and for interfaces with other modes of transport*, Official Journal of the European Union.

European Commission, 2016, A European strategy for low-emission mobility, COM(2016) 501 final.

Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., 1996, `Knowledge discovery and data mining: towards a unifying framework´, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pp. 82-88.

Gama, J., Gaber, M.M., *Learning from Data Streams: Processing techniques in sensor networks*, Springer, 2007.

Gama, J., *Knowledge discovery from data streams*. Chapman and Hall/CRC Press, New York, 2010.

Gama, J., `Data stream mining: the bounded rationality´, *Informatica* Vol. 37, No 1, 2013, pp. 21-25.

Ganguly, A.R., Gama, J., Omitaomu, O.A., Gaber, M., Vatsavai, R.R., *Knowledge discovery from sensor data*, CRC Press, Boca Raton, 2009.

Google, *Designing a connected vehicle platform on Cloud IoT Core*, https://cloud.google.com/solutions/designing-connected-vehicle-platform, last accessed 30 September, 2017.

Haroun, A., Mostefaoui, A. , Dessables, F, `Improving Data Fusion in Big Data Stream Computing for Automotive Applications´, *2016 International IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress*, 2017.

Higgings, S., `The DCC Curation lifecycle model´, *International Journal of Digit Curation*, Vol. 3 , No 2, 2008, pp. 135-140.

Hoens, T.R., Polikar, R., Chawla, N.V., `Learning from streaming data with concept drift and imbalance: an overview´, *Progress in Artificial Intelligence, Vol.* 2012, No 1, 2012, pp. 89-101.

Kargupta, H., 2017, *Data analytics for connected cars*, retrieved on 26 September 2017 from https://data-analytics.cioreview.com/cxoinsight/data-analytics-for-connected-cars-nid-6142-cid-156.html.

Khan, S.M., Rahman, M., Apon, A., Chowdury, M. Chowdury, M., Characteristics of intelligent transportation systems and its relationship with data analytics, in: Chowdury, M., Apon, A., Dey, K., (Eds.) *Data analytics for Intelligent Transportation Systems*. Elsevier, The Netherlands, pp. 1-30, 2017.

Krawcyz, B., Minku, L.L., Gama, J., Stefanowski, J., Wozniak, M. 2017. `Ensemble learning for data stream analysis: a survey´, *Information Fusion, Vol.* 37, 2017, pp. 132-156.

MDOT (Michigan Department of Transportation), CAR (Centre for Automotive Research), Leidos, 30 Sept 2014, *Connected vs. automated vehicles as generators of useful data*, retrieved on 26 September 2017 from http://www.cargroup.org/publication/connected-vs-automated-vehicles-as-generators-of-useful-data/.

Muthukrishnan, S., `Data streams: algorithms and applications´. Foundations and Trends, *Theoretical Computer Science*, Vol 1, No. 2, 2005, pp. 117-236.

Myers, J., Hedstrom, M., Akmon, D., Payette, S. et al. `Towards sustainable curation and preservation: The SEAD project's data service approach´, 2015 IEEE 11th International Conference on e-Science (e-Science). Munich, Germany, 2015.

Nguyen, H.L., Woon, Y.K., Ng, W.K., `A survey on data stream clustering and classification´, *Knowledge and Information Systems*, Vol., 45, No 3, 2015, pp. 535-569.

Plale, B., Kouper, I., The Centrality of Data: Data lifecycle and Data Pipelines, in: Chowdury, M., Apon, A., Dey, K., (Eds.) *Data analytics for Intelligent Transportation Systems*. Elsevier, The Netherlands, pp. 91-111, 2017.

Ramírez-Gallego, S., Krawczyk, B.,, García, S., Wozniak, M., Herrera, F., `A survey on data preprocessing for data stream mining: current status and future directions´, *Neurocomputing*, Vol. 239, 2017, pp. 39-57.

RDE (The US Federal Highway Administration's Research Data Environment), https://www.its-rde.net/RDE-Visualizations/, last accessed 29 September 2017.

Rehman, M.H., Liew, C.S., Wah, T.Y., Khan, M.K., `Towards next-generation heterogeneous mobile data stream mining applications: Opportunities, challenges, and future research direction´, *Journal of Network and Computer Applications*, Vol. 79, No 2017, 2017, pp. 1-24.

SEAD, http://sead-data.net/, last accessed 10 December 2017.

Steed, C.A., Interactive data visualization, in: Chowdury, M., Apon, A., Dey, K., (Eds.) *Data analytics for Intelligent Transportation Systems*. Elsevier, The Netherlands, pp. 167-190, 2017.

USGS (United States Geological Survey), 2013, Faundeen, J.L., Burley, T.E., Carlino, J.A., Govoni, D.L., Henkel, H.S., Holl, S.L., et al., The United States Geological Survey Science Data Lifecycle Model, retrieved on 22 September 2017 from https://pubs.usgs.gov/of/2013/1265/pdf/of2013-1265.pdf.

Yau, N., *Visualize this: The flowing data guide to design, visualization, and statistics*, Wiley, 2011.

Wang, Y., Tian, D., Sheng, Z., Wang, J., *Connected Vehicle Systems. Communications, Data, and Control*, CRC Press, Florida, 2017.

Zang, W., Zhang, P., Zhou, C., Guo, L., `Comparative study between incremental and ensemble learning on data stream: case study´, *Journal of Big Data*, Vol. 5, 2014, pp. 1-16.

Zhang, P., Zhu, X., Shi, Y. Categorizing and mining concept drifting data streams, in: KDD '08, 2008, *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp 812-820.

List of abbreviations

ART	Autonomous Road Transport
CV	Connected Vehicle
DS	Data Science
kbps	Kilobytes per second
KDD	Knowledge Discovery in Databases
Mbps	Megabytes per second
i.i.d	independent and identically distributed (data)
IoV	Internet of Vehicles
ITS	Intelligent Transportation Systems
VA	Visual Analytics

List of figures

List of tables

GETTING IN TOUCH WITH THE EU

In person

All over the European Union there are hundreds of Europe Direct information centres. You can find the address of the centre nearest you at: http://europea.eu/contact

On the phone or by email

Europe Direct is a service that answers your questions about the European Union. You can contact this service:

- by freephone: 00 800 6 7 8 9 10 11 (certain operators may charge for these calls),

- at the following standard number: +32 22999696, or

- by electronic mail via: http://europa.eu/contact

FINDING INFORMATION ABOUT THE EU

Online

Information about the European Union in all the official languages of the EU is available on the Europa website at: http://europa.eu

EU publications
You can download or order free and priced EU publications from EU Bookshop at: http://bookshop.europa.eu. Multiple copies of free publications may be obtained by contacting Europe Direct or your local information centre (see http://europa.eu/contact).

KJ-NA-28923-EN-N

JRC Mission

As the science and knowledge service of the European Commission, the Joint Research Centre's mission is to support EU policies with independent evidence throughout the whole policy cycle.

EU Science Hub
ec.europa.eu/jrc

@EU_ScienceHub

EU Science Hub - Joint Research Centre

Joint Research Centre

EU Science Hub

Publications Office

doi:10.2760/822136

ISBN 978-92-79-77041-8