On Learning From Inaccurate and Incomplete Traffic Flow Data

Francesco Alesiani, Member, IEEE, Luis Moreira-Matias, Member, IEEE, and Mahsa Faizrahnemoon

Abstract—Today, we live in an era where pervasive sensor networks both collect and broadcast rich digital footprints about the human mobility. However, most of this data often comes in an incomplete and/or inaccurate fashion. In this paper, we propose a knowledge discovery framework to handle such issues in the context of automatic incident detection systems fed with traffic flow data. This framework operates in three steps: 1) it clusters sensors with a novel multi-criteria distance metric tailored for this purpose, followed by a heuristic rule that labels the abnormal groups; 2) then, a spatial cross-correlation framework identifies seasonal and individual abnormal readings to perform a more fine-grained filtering; and 3) finally, we propose a novel fundamental diagram that discovers the critical density of a given road section/spot on a data-driven fashion that is resistant to both outliers and noise within the input data. Large-scale experiments were conducted over traffic flow data provided by a major Asian highway operator. The obtained results illustrate well the contributions of this framework: it drastically reduces the noise within the raw data, and it also allows determining reliable definitions of traffic states (congestion/no congestion) on a completely automated way.

Index Terms—Automatic incident detection, fundamental diagram, traffic flow, clustering, local principal curves, Gaussian process, ensemble learning.

I. INTRODUCTION

Traffic incidents [1] are one of the major challenges of sustainable Urban Mobility. A traffic incident occurs when traffic demand exceeds the existing road network capacity. It is caused either by excessive demand (e.g., during peak hours) or by suddenly reduced capacity provoked by an external event, e.g., car accident. When traffic incident occurs, the traffic flow suffers a disruption which usually results on an abnormal traffic behavior, e.g., large travel delays. The use of human mobility data has been considered by many researchers across different fields as one of the most promising approaches to solve to this problem [1]–[4].

Nowadays, there is a constant stream of mobility data (i.e., information that describes the mobility of persons, goods, or vehicles). Notable examples of possible sources include cellular communication logs, Global Positioning System (GPS) traces, traffic counting, Automated Vehicle Location (AVL) system records. During last decades, both companies and city councils joined efforts on putting up the required technological infrastructures to both capture and store all this mobility-related data (e.g., mass transit agencies [5]). However, today’s problem is how to take advantage of such data to extract information able to improve the mobility levels worldwide. The main idea is to minimize and/or anticipate these traffic incidents to reduce congestion levels in and between major urban areas. Consequently, automatic methods have attracted a large attention in recent years for their capacity of extracting relevant insights on different problem dimensions, from demand estimation [6] to short-term link travel time prediction [7].

The Machine Learning frameworks able to automatically flag an Incident on an early stage have turned increasingly popular in the recent years. Such early detection can help to restore a smooth traffic flow through Advanced Traveler Information Systems, e.g., variable message signs. These systems are commonly based on Traffic Flow data [1], [2]. Traffic Flow data comprises information on the vehicle movement on the road. Typically, there are five quantities related to this type of data: flow (vehicle per unit of time), density (vehicles per unit of distance), occupancy (percentage of time a specific part of the road is occupied by vehicles), speed (distance per unit of time), headway (time or distance between vehicles) and travel time (time between two points). This type of data contains peculiar characteristics that disallow many of standard inference methods to be put in place. Some of them are enumerated as follows: (i) it can be collected through a wide range of heterogeneous sources from static ones (e.g., fixed loop counters) [4] to dynamic ones (e.g., smartphones [8]); (ii) it presents a ratio between information and noise which may vary for different data sources and/or timespans (i.e., inaccurate) [9]; (iii) the collection periodicity is also irregular, resulting into relevant amounts of missing data [10] (i.e., incomplete); (iv) the resulting datasets are highly sparse while they lack on exact information (i.e., meta-data) about each source’s collection process (e.g., users identifier or sensor’s exact location).

In the last decades, a large investment was put in place to deploy large-scale sensor-networks and/or other types of hardware infrastructure in highways. Despite its limitations, modern Intelligent Transportation Systems (ITS) are expected to cope with such heterogeneous uncertain sources while still capturing relevant patterns from such raw data.
In this paper, we focus on learning from incomplete and inaccurate traffic flow data to build Automatic Incident Prediction (AIP) systems. From a high-level perspective, our goal is to provide a framework able to transform raw traffic flow data into an adequate and informative input to ML-based AIP frameworks, independently of its source(s) or degree of completeness. To do so, we propose three ML frameworks: (1) a clustering framework able to detect abnormal sensors (i.e., which are malfunctioning or producing data with a very low ratio information/noise); (2) a data quality indicator based on spatial correlations, which aim to identify anomalous changes in the sensor readings and (3) a semi-supervised ML method able to automatically determine scenario-oriented thresholds to define incidents from traffic flow data of a single road section, which are robust to outliers/noise within the input data. These frameworks may be used standalone (i.e., for preprocessing) or together as a full pipeline, depending on their final application. Their insights were validated through experiments conducted over data collected from a major Asian highway operator. The main contributions of this work are:

- the first ML-based framework (1-3) to be able to determine scenario-oriented definitions for incidents on a complete unsupervised way, independently on the noise levels within the input data;
- a novel multi-criteria statistical distance which is able to highlight data sources producing a high level of erroneous, invalid or missing values without any prior assumptions;
- a bagging-like ensemble method [11] able to identify the admissible maximum flow density in a section which can be up to 30% more robust to noise than the State-of-The-Art on this topic (i.e., Local Principal Curves [12]);
- a fuzzy traffic indicator component that detects anomalous sensor behaviors based on their raw measurements.

The remainder of the paper is structured as follows: Section 2 reviews the existing literature on related topics, pointing out its limitations as well as the problem statement addressed in this paper. Sections 3, 4 and 5 formally present the three steps of the methodology employed to solve such problem. The sixth section firstly describes how the dataset was preprocessed for this task, along with some descriptive statistics. Section 7 details how the methodology was tested in a real scenario, as well as the obtained results, followed by a brief discussion on their outcome. Finally, conclusions are drawn.

II. PROBLEM OVERVIEW

AIP algorithms can be folded into five categories [1], [2]: (1) traffic state/change detection; (2) data-driven event detection; (3) image-based processing, (4) traffic theory models and (5) traffic flow - occupancy prediction.

The first category of methods approaches the problem of detecting (typically, fast) changes on the flow/occupancy through some sort of signal processing filters (e.g., low-pass filter), as suggested by the well-known California algorithm [14]. However, to define a correct passband to apply in each situation may be tricky as incidents with different natures may also provoke distinct flow/occupancy rates. This issue is better addressed using (2) event detection methods, which use incident-based historical data to perform AIP. Naturally, the assumption that such data is available shrinks the applicability of these types of methodologies (as it requires that a plausible threshold definition(s) was already put in place by the operator). The two most common approaches on this are Fuzzy Logic (FL) [15] and Artificial Neural Networks (ANN) [2], [16]. FL is able to encode domain expert generated logical rules (typically expressed as combination of AND and OR operators) into continuous function, whereas ANNs are powerful machines able to establish complex (e.g., non-linear) relationships between a target variable (e.g., incident/no-incident label) and multiple explanatory variables directly from data. However, its effectiveness comes with a large set of drawbacks such as the need to have a consistently large dataset for achieving an near-optimal bias-variance tradeoff on training stage, a long and slow training stage, its inability to provide interpretability on their findings (as the target function is always unknown) or its high sensibility to noise. On the other hand, fuzzy logic methods are able to deal with inaccurate and small datasets. Yet, the definition of any fuzzy logic (independently of its type) is based on assumptions on the underlying model and requires potentially a large number of parameters, which may increase the problem’s complexity on an easy but undesired manner [17] (see more about fuzzy logic systems in Section IV).

Image-based processing techniques (3) take advantage on video-surveillance systems installed on the freeways to apply computer vision technology able to extract information about the traffic parameters to model their short-term status (by combining this data with other types AIP methods) and then, to verify the incident’s occurrence. Recently, the popularity of these methods have been increasing [3], [18]. However, just a few worldwide’s freeways are equipped with such type of systems. (4) Model-based models settle on complex theoretical formulations which aim to explain the traffic flow/occupancy. Hence, these methods rely on strong assumptions that may not hold when the system changes unexpectedly. Traffic flow have an infinite dimension which runs on nonlinear, stochastic and time-variant dynamic system. Usually, an traffic flow model is found under a certain finite set of assumptions which may be valid for a particular scenario (i.e. road segment) [1], [19]. Again, a scenario-oriented threshold to detect the presence of an incident is required to put any method of this type in place.

The Traffic flow/occupancy prediction (5) for AIP can be divided on two steps [1]: firstly, a prediction algorithm is used to infer the short-term future values of flow/occupancy based on historical data. The most commonly used methods to perform short-term traffic flow prediction include statistical filtering [20], time series analysis [4], ANNs [21] and Kalman Filters (KF) [22]. Secondly - and similarly to all the abovementioned AIP methods -, it compares the predicted values with a certain predefined threshold which defines the
occurrence of a traffic incident and/or with the real values. Additionally, most of the well-known AIP approaches are high sensitive to the presence of noise in the input data. Consequently, techniques to adequately prune the noisy sensors and/or filter such noisy samples are necessary to increase the applicability of such AIP algorithms on real-world solutions.

The problem we want to solve in this paper is to learn context-aware thresholds for AIP under a certain (apriori unknown) degree of uncertainty regarding the noise levels and the missing/erroneous values within the input data on a completely unsupervised fashion. Unlike typical problems and/or case studies presenting in traffic flow theory, engineering and related fields, we are not aim to address a specific problem on a particular road section (e.g. a junction that get always congested in rainy days), but rather to be able to sense and/or predict any possible traffic condition on any location of our road network. Hereby, we want to put forward the first step towards building an end-to-end system able to decide over our road network on a fully automated way.

To do it so, we propose a framework with three components - as illustrated by the diagram in Fig. 1. Firstly, (1) the system excludes faulty sensors following a two-stage process: (1-i) a top-down clustering framework which groups together sensors with similar behavior through a customized statistical distance metrics to then perform (1-ii) a selection based on dispersion statistics computed over a two-dimensional (2D) projection of the clusters samples through a Principle Coordinates Analysis (PCoA) [23]. Secondly, the framework computes a noise indicator for each road section, based on corridor-based comparisons with the nearest sections through a set of fuzzy systems, which removes typical within-day and intra-day variations. Then, when a sensor pruning stage takes place, a user-defined threshold is defined to filter admissible noise level sensors. The resulting set of sensors serves individually as input of three numerical predictors: two regression methods\(^2\) - (i) the State-of-The-Art Local Principal Curve and (ii) a robust domain-oriented gaussian process which determine an explanatory function for their density (flow vs. occupancy rate), as well as (iii) a robust maximum procedure for the critical density. The three correspondent outputs are passed through a bagging-like ensemble which computes an aggregated threshold (i.e. a minimum

\(^2\)where the resulting function maximum value represents their threshold.

admissible flow \(f_{Gi}^{\text{min}}\) and a maximum occupancy rate \(o_{Gi}^{\text{max}}\) to determine the transition point between free flow and an initial jam state (i.e. light incident/jam). These frameworks are formally described in the Sections III, IV and V, respectively.

III. CLUSTERING-BASED FAULT DETECTION

Let \(F_{Gi,t}\) and \(O_{Gi,t}\) be the averaged traffic flow and the lane occupancy rates on a given road section \(G_i\) aggregated by periods of \(n\)-minutes, respectively, measured until the time instant \(t\). They can be defined as follows:

\[
F_{Gi,t} = \{f_{i,1}, \ldots, f_{i,t}\} : f_{i,t} \in [0, +\infty), \quad \forall t \in \{1..t\} \quad (1)
\]

\[
O_{Gi,t} = \{o_{i,1}, \ldots, o_{i,t}\} : o_{i,t} \in [0, 1], \quad \forall t \in \{1..t\} \quad (2)
\]

Our goal is to detect sensors with an anomalous output based on their behavior (expressed by \(F_{Gi,t}\)). The proposed framework goes on two steps: firstly, we cluster the sensors with similar behavior given a distance matrix built upon a statistical distance metric specifically tailored for this purpose. Then, we select the clusters which contain anomalous sensors by proposing a heuristic rule that iterates over PCoA-based two-dimensional projections of the cluster’s relative positions. Projection into 2d space also allows an easier visualization of the results. These two steps are detailed below.

A. Clustering Using Customized Distance Matrix

In order to produce a group structure from a complex data set, a measure of proximity is required. A metric which aims to express a statistical distance is usually a value which is symmetric (e.g. the distance from the object \(i\) to the object \(j\) is the same than from \(j\) to \(i\)). The most commonly applied metric to do it so (e.g. for clustering) is the Euclidean one, which can be defined as

\[
d_{EUC}^2(i, j) = \|F_{Gi,t} - F_{Gj,t}\|^2 = \left( \sum_{t=1}^{t} (f_{i,t} - f_{j,t})^2 \right)^{\frac{1}{2}} \quad (3)
\]

However, this metric standalone does not serve our purpose, i.e. to group together sensors with similar behavior regarding their production of erroneous/missing values. To complement it, we propose a novel metric defined as a linear combination of three distinct metrics. Let \(A\) be a square matrix of size \(n \times n\) containing the distances between each possible pair of sections \(\gamma^l_{i,j}, \forall i, j \in \{1..n\}\). Let \(p^l_i(f)\) define the sample-based probability density function of the flows produced by a sensor \(G_i\) up to time \(t\). The distance between two sections, \(\gamma^l_{i,j}\), can be computed as follows:

\[
\gamma^l_{i,j} = \frac{1}{2}d_{JSD}(i, j) + \frac{1}{3}d_{EUC}(i, j) + \frac{1}{6}d_{MIS}(i, j),
\]

\[
d_{JSD}(i, j) = \frac{D_{KL}(p^l_i || p^l_j) + D_{KL}(p^l_j || p^l_i)}{2}, \quad (4)
\]

\[
D_{KL}(p^l_i || p^l_j) = \int_{-\infty}^{\infty} p^l_i(f) \ln \frac{p^l_i(f)}{p^l_j(f)} df, \quad (5)
\]

\[
d_{MIS}(i, j) = \begin{cases} 1 & \text{if } \theta^l(i) = \theta^l(j) \\ 0 & \text{otherwise.} \end{cases} \quad (7)
\]
\[
\theta^i(t) = \begin{cases} 
1 & \text{if } \exists \ l \in \{1, \ldots, I\} : f_{i,l} < 0 \lor f_{i,l} = \text{NaN} \\
0 & \text{otherwise.}
\end{cases}
\]

where \( p_i^j \) is given by \( p_i^j(f) = \frac{p_i^j(f)+p_i^j(f)}{2} \), \( d_{JSB}(l,j) \) is computed using the Jensen-Shannon distance - a popular method of measuring the similarity between two probability distributions [24] - while \( d_{MIS}(l,j) \) attests if the flow data produced by those two sensors have erroneous or missing values (symbolized by NaN). \( \Delta^l \) serves the similarity of sensors in amplitude, distribution and missing values patterns. Then, \( \Delta^l \) serves as an input to a clustering algorithm of interest, i.e. \textit{clus} which will produce results for a predefined range on the number of desired clusters, i.e. \( K \). Finally, the optimal number of clusters \( k^* \in K \) is the one which minimizes the Bayesian Information Criterion \([25]\).

**B. Anomalous Cluster Detection**

After obtain a set of \( k \) clusters, i.e. \( \Pi = \bigcup_{i=1, \ldots, k} \Pi_i \), it is necessary to decide which are the ones containing abnormal/faulty values (if any). The first step is to project the original samples into a 2D space using PCoA. The aim of this step is to normalize the distances into two independent and uncorrelated dimensions. Let \( X = \{x_1, \ldots, x_n\} \) denote a vector of coordinates projecting the original sensors into a 2D dimensional space. \( X \subset \mathbb{R}^2 \) can be computed as an optimization problem:

\[
\arg \min_X \sum_{i,j=1}^n \left( \|x_i - x_j\| - d_{ij} \right)^2, \quad \forall x_i, x_j \in X
\]

where \( d_{ij} = \gamma_{ij} \). After finding the \( X \) correspondent to the local minima of the Eq.9, we compute inter and intra-cluster distances to attest their dispersion. Let \( x_{\Pi_j} \) be the projected coordinates of a sensor included in the cluster \( \Pi_j \subset \Pi \), where \( \bar{x}_j \) defines the coordinates of their centroids. The intra-cluster distance, i.e. \( d_{INT}^{\Pi_j} \) can be computed as

\[
d_{INT}^{\Pi_j} = \frac{1}{|\Pi_j|} \sum_{x_i \in \Pi_j} ||x_i - \bar{x}_j||, \quad \forall x_i \in \Pi_j
\]

Similarly, the inter-cluster distance of \( \bar{x}_j \) can be computed as

\[
d_{EXT}^{\Pi_j} = \frac{1}{k} \sum_{i=1, \ldots, k, i \neq j} ||\bar{x}_j - \bar{x}_i||
\]

Finally, the binary selection of faulty clusters is computed as follows

\[
F_{\Pi_j} = \begin{cases} 
1 & \text{if } d_{EXT}^{\Pi_j} > \sigma_{EXT} \land (d_{INT}^{\Pi_j} > \sigma_{INT} \lor |\Pi_j| < \eta) \\
0 & \text{otherwise.}
\end{cases}
\]

where \( \sigma_{EXT}, \sigma_{INT} \) denote the standard deviation of all the internal/external distances, respectively, and \( \eta \) is an user-defined parameter to denote the concept of small-sized clusters (i.e. typically, \( \eta < 10 \)).

By doing so, we exclude the sensors whose values’ statistics strongly diverge from the remaining ones. To perform a more fire-grained pruning of the sensors based on their noise levels, we rely on mining the expected coherency given by the spatial correlations within the road topology. Such indicator computation is described in the next section.

**IV. SENSOR PRUNING USING SPATIAL CORRELATIONS**

There are two typical types of changes on the traffic behavior along consecutive road sections: (1) punctual ones, which tend to be limited in time or (2) regular ones, which will repeat themselves with a certain periodicity. Consequently, we can empirically derive that an abrupt drift may not be related with typical events (i.e. which originate recurrent drifts as in \([26]\)), but rather to some malfunction in our road sensor network (e.g. communicational issues, misconfiguration, inaccurate measurements, etc.). Recurrent drifts reflect behavioral changes that tend to normalize after a certain period of time. Consequently, any reliability indicator should be monitored within a given sliding window. The size of such time window will define the method’s sensibility - low values may raise a high number of false alarms, while high values may skip some relevant alarms as well. Hereby, we propose to consider the spatial correlation of traffic flow signals (e.g. flow, occupancy) between to consecutive sections. We did it so by identifying changes in the correlation function output. Consequently, the indicator computation elaborates on this idea. The correlation between two consecutive sections \( G_j, G_{j+1} \) can be computed as follows:

\[
C_{G_j,G_{j+1}}(t, \zeta) = \frac{\sum_{t=1}^{T}(\varepsilon_t - \overline{\varepsilon})(\eta_{t,\zeta} - \overline{\eta})}{\sqrt{\sum_{t=1}^{T}(\varepsilon_t - \overline{\varepsilon})^2\sum_{t=1}^{T}(\eta_{t,\zeta} - \overline{\eta})^2}} \quad \forall \varepsilon, \eta
\]

where \( \varepsilon = F_{G_j}(t + \omega + T/2 - 1, \zeta) \) and \( \eta = F_{G_{j+1}}(t + \omega + T/2 - 1, \zeta) \) stand for data value on the two sections \( G_j, G_{j+1} \) respectively, of day \( \zeta \) at time instant \( t \) inside a sliding window whose size is given by \( T \) and is centered at \( \omega \), while \( \overline{\varepsilon} = \frac{1}{T} \sum_{t=1}^{T} \varepsilon_t \) and \( \overline{\eta} = \frac{1}{T} \sum_{t=1}^{T} \eta_t \).

To compute the indicator values, we rely on a Fuzzy Logic system \([27]\), which is defined by a set of fuzzy membership functions for the inputs/outputs and a set of rules to establish the correspondent dependencies. It is able to express logical rules into a non linear function. The rule sets are adaptable to every evaluation scenario.

The basic idea behind the use of such system is that consecutive sections should present high cross-correlations of their flow counts that changes in case of sensor malfunction. Consequently, a low level of noise would result in a high cross-correlation value on a given lag. Such fuzzy framework is fed by a set of parameters whose values can be derived from the cross-correlation structure between two consecutive sections. Fig. 2 illustrates a possible decomposition of the cross-correlation structure between two consecutive sections on such parameter set. Those parameters can be enumerated as follows: (1) \( C_{\text{Max}} \) represents the maximum correlation value between two consecutive road sections while (2) \( \tau_{\text{Max}} \) denotes the correspondent time lag. (3) The area beneath such maximum and (4) its change rate can be denoted by \( A_{\text{Max}} \) and \( \alpha_{\text{Max}} \), respectively. where the latter one is denoted by the angle formed by the two correlation function lines that intersect on such maximum.
This set of data quality indicator is extracted from the intermediate outputs of three fuzzy systems. Each indicator will correspond to a value ∈ [0, 1], where the data quality concept would be reversely proportional to the amount of noise within.

Hereby, we propose to measure the data quality (i.e. the amount of noise) of a given sensor by using a two step approach (depicted in Fig. 4): (i) on a first layer, we compute the correlation level of each consecutive sensor pair with approach (depicted in Fig. 4): (i) on a first layer, we compute the correlation level of each consecutive sensor pair with respect to three different fuzzy systems (A, A’ and B) that take as input the above mentioned parameters $C_{M_{\text{Max}}}$, $\cos(\alpha_{\text{Max}})$, $A_{\text{Max}}$ and/or some processing over their values. Then, (ii) the outputs of them are combined using a fourth fuzzy system $C$. Fuzzy system $A$ considers the current day correlation and it outputs an indicator regarding the behavior of a single day, $I_D$. The second, i.e., $I_M$, is based in the system A’. It considers the average correlation over some days, e.g. a week. The third system $B$ considers the differences of the correlation parameters and outputs the indicator $I_E$. The last fuzzy system, $C$, is used to combine the intermediate indicators: 1) the indicator associated with the historical data, 2) the current data or 3) the difference between the previous ones. The system $C$ outputs $I_G$, which is the ultimate indicator value to be considered.

We use Mamdani fuzzy system with triangular membership $m_i(x)$ functions for both input and output, as proposed in [28]. The membership function are illustrated in Fig. 3 (which depicts the three internal discrete states necessary for the fuzzyfication/defuzzyfication as Low/Normal/High along with the definitions of symbols $a, b, c, d, e, f$). It is possible to observe that each membership function overlaps its neighbor functions, being defined by both its start and end points. They can be defined as follows:

$$m_0(x) = \begin{cases} \frac{(b - a) - x}{b - a} & \text{if } a \leq x \leq b \\ 0 & \text{otherwise.} \end{cases}$$

$$m_1(x) = \begin{cases} \frac{(x - c)}{(d - c)/2} & \text{if } c \leq x \leq c + (d - c)/2 \\ \frac{(c + (d - c)/2 - x)}{(d - c)/2} & \text{if } c + (d - c)/2 \leq x \leq d \\ 0 & \text{otherwise.} \end{cases}$$

$$m_2(x) = \begin{cases} \frac{x - e}{f - e} & \text{if } e \leq x \leq f \\ 0 & \text{otherwise.} \end{cases}$$

where $m_0(x)$, $m_1(x)$ and $m_2(x)$ are ∈ [0, 1].

Its rules are described in Table I, where each one defines the input and output fuzzy classes. The input sets are combined with logical operator $\text{AND}$. Consequently, the inference of the output member uses the min operator. The final output value is generated by the intermediate weighting of the single input membership values using Centroid defuzzification, as proposed in [29].

After pruning out malfunctioning/abnormal sensors in the last two steps, it is necessary to find out which are the definitions of traffic state in each sensor location/segment. A framework to do it in a automated data-driven way is proposed in the next Section.

V. LEARNING CONTEXT-AWARE THRESHOLDS FOR AIP

One of the most commonly used techniques to understand road congestion in Traffic Engineering is the Fundamental Diagram (FD) [13], [30]. It aims to represent the relationships between the fundamental quantities in traffic theory: vehicle flow (vehicles/hour), speed (km/hour) and density (vehicle/km). Figure 5a illustrates an example of
Flow phase, (ii) the Synchronized Flow) is of particular interest to address congestion. Naturally, this threshold is of particular relevance on predicting congested phases due to the particular traffic movements inherent to those (i.e. irregular interaction among vehicles). The transition between this two types of phases (Free Flow and Synchronized Flow) is of particular interest to address congestion. Early alarms (and the consequent control measures) will affect negatively the traffic, increasing the section traversing time unnecessarily. Per opposition, late alarms may reduce the effectiveness of the control measures, which can be translated into additional delays to the affected road section’s cruising times. Traditionally, the critical parameters are theoretically documented in tabular form in The Highway Operation Guide [31], where the road topology is the input given to determine their values. This method may not reflect the actual current road characteristics (e.g. due to temporary roadworks, weather phenomenons or infrastructural failures caused by a deficient maintenance plan). Consequently, congestion may be detected in very later stages. Some modern attempts to use flow-based data measurements to build FD are detailed in [32] - where the approaches rely in some helpful heuristics that allow to fit the parameter values to the input samples somehow. Although they typically do not settle upon many assumptions about the road topology, they end up approximating the ground truth when fed with quality input samples.

The major drawbacks of the data driven approaches are the implicit loss of interoperability with respect to the remaining Traffic Management System, as well as their reliability when fed with low-quality data, i.e. noise. A typical example on this issue is the common use of the maximum flow to estimate the critical density. The maximum is typically very prone to outliers (e.g. an error on a single measurement), forcing the dataset to be preprocessed beforehand. At the best of our knowledge, this aspect have been neglected in the State-of-The-Art of this topic. In this Section, we introduce a data driven method to estimate the Critical Parameters of a FD which is robust to noise.

### A. Data Driven Fundamental Diagram Estimation

Two successfully applied methods are the Local Principal Curve (LPC) [12], [33] and the Gaussian Process (GP) [34]. Like many other Supervised Machine Learning methods (e.g. Support Vector Machines [35]), these methods tend to overfit when fed with a low number of samples, i.e. they tend to follow sudden changes/variances which have no evident explanation.

GP is an algorithm typically used for regression [34]. It is based on the concept that any two values of the target function \( f(x) \) evaluated in two data points \( x_i, x_j : i \neq j \) follow a Gaussian distribution whose covariance is given by a kernel function, i.e. \( k(x, y) \). An example of kernel function is the
squared exponential with noisy term, defined as follows
\[ k_e(x_i, x_j) = \sigma^2_{ke} \exp \left( -\frac{||x_i - x_j||^2}{2\sigma^2_{ke}} \right) + \sigma^2_\delta \delta(x_i, x_j) \]  
(17)
where \(\delta(x_i, x_j)\) is a function which evaluate zero except when the inputs are equal, \(\sigma^2_\delta\) and \(\sigma^2_{ke}\) are the weighting factors of the kernel function, while \(\sigma^2_i\) define the size of the influence between two features \(x_i, x_j\). A proper choice of \(\sigma^2_i\), also dependent on the value of the occupancy, can be used to counteract the absence of data in some data ranges.

LPC [12] is also a regression method. It is an extension of Principal Curve (PC) [36]. It can be considered a non linear version of Principle Component Analysis (PCA) [37]. The method seeks for a local one-dimensional approximation of the data variance. A normal unconstrained version of this procedure may overfit high-variance components of the data in some locals of the search space, turning the resulting curves to be impractical or rather useless.

B. Robust Gaussian Process

GP as local method is affected by outliers - whose relevance is inversely proportional to the number of input samples. Moreover, The correlation distance of the kernel function in Eq.17 is proportional to \(\sigma_j\). Varying this parameter can reduce the impact of outliers. However, it may also affect the quality of the approximation.

To mitigate such noise sensitivity issue for the present application, we propose a robust version of the Gaussian process by pruning the data input to the one included on the first three quantiles. Consequently, we ignore the values within the latter one. This process enforces a biasing of the target function curve towards a lower flow level than the admissible one, but hence ignoring upwards outliers.

The direct consequence of doing so is to obtain a curve which describes most of the data on a more robust way by exposing less the target function estimation process to the introduction of singular outliers in the learning stage.

This pruning of the fourth quantile to turn the learner more robust to outliers is similar (in concept) to the soft margin introduced by Cortes and Vapnik [36] in his Support Vector Machines algorithm. By assuming that the regressor will never approximate values within the fourth quantile (which is not that relevant for this application), we are also setting up our available budget, i.e. \(C\), to the admissible errors on our output values - even if the absolute value of \(C\) would depend here on the number of input data samples, as well on the data distribution within such quantile.

C. Ensemble Learning

In order to reduce the overfitting effects introduced by sparse locals and/or insufficient input samples, we also proposed to employ an ensemble. The Ensembles are a combination of learning methods that typically aim to reduce the variance of their outputs, thus decreasing their generalization error. In this context, we propose to do it so to by combining their outputs through majority voting schemas (either simple or weighted average). The weights considered are inverse proportional to the distance of the flow to the theoretical flow, i.e. \(q_\tau\). Given a series of \(j\) baseline predictors producing the following outputs \(M = M_1, \ldots, M_j\) where \(M_i = \rho_i C, \forall i \in \{1, \ldots, j\}\) we can compute a generic ensemble of them using a linear combination of them computed as follows
\[ \rho_C = \sum_{i=1}^j w_i \rho_i \]  
(18)
where the weights \(w_i\) can be alternatively defined by one of the following two equations:
\[ w_i = \frac{|q_i - q_\tau|}{\Phi}, \Phi = \sum_{i=1}^j w_i \]  
(19)
\[ w_i = \frac{1}{j} \]  
(20)
where the first represents the weighted and the second stands for the mean ensemble, respectively.

These methods were then tested using real-world flow data. The case study used to do it so is described in the following Section.

VI. Case Study

This study was conducted using data collected through a traffic monitoring system of a major expressway connecting two major cities in Japan. The whole process took place in January 2014. Its layout is composed by 3 lanes in both directions during roughly 90% of its total extension and two lanes in the remaining segments.

This system both collects and broadcasts traffic-based measurements in real-time with distinct temporal granularities (depending on the type of sensor’s installed on each lane). Each sensor measures traffic flow, lane occupancy rate and instantaneous vehicle’s speed. Yet, just the data of the first two was used for this study. The largest time granularity of this data collection system (\(p = 5\) minutes) was used to normalize all the collected time series. Samples containing missing data suffered a simple data imputation process where the missing values inferred by interpolating the values of its Euclidean neighborhood. This step aims to establish a common comparative testbed for different sections, independently on its lane number or sampling frequency. Hence, it disables the possibility to distinguish main lanes from input/output ramps flows.

This dataset was formed by collecting data from 106 sensors which includes both freeway’s transit directions. The total length of the analyzing sections is roughly 20km while its sensors are deployed each 500m. This data was collected through 3 non-consecutive weeks. Fig. 6 illustrates five sample-based probability density functions obtained using a (gaussian) kernel density estimator over all the flow measurements available - one global and four specific for each of the considered timespans (divided by Periods I-IV, identified by the same display order as Fig. 6 legend). Table II includes descriptive statistics on our dataset. The top 10 sensors regarding the number of observed incidents were highlighted. As it is observable, the occupancy rate is higher in these sensors. Not surprisingly, the most critical period is the morning peak, comprised between 6:40 and 13:20.
Fig. 6. Flow-Based Probability Density function estimated from the all available samples aggregated by 15m-periods.

Fig. 7. Illustration of the fuzzy system modus operandi for detecting anomalies in our sensors.

VII. EXPERIMENTS

This section starts by introducing the Experimental Setup followed by our experiments, pointing out the obtained results as well as a brief discussion on their insights.

A. Experimental Setup

We divided our experimental testbed into two distinct stages: (i) Sensor Pruning and (ii) Context Learning. On the first stage, we try to exclude sensors which are unreliable by detecting some sort of malfunction based on their measurements.

The (i) first stage was performed by applying the methodology proposed in Sections III (for sensors exhibiting a very high degree of abnormality and which require to be replaced) and IV (for sensors which exhibit seasonal and/or lower degree of abnormalities which may require some maintenance), respectively.

To approach the (ii) Context Learning problem, we evaluated seven methods - two baselines from State-of-the-Art, (1) Max Flow (MF) and (2) Robust Max Flow (RMF); two methods from the State-of-the-Art in this topic, (3) GP and (4) LPC; and three methods proposed hereby: Robust GP (RGP), Weighted and Mean Ensemble (WE and ME, respectively) of RMF, RGP and LPC. The last five methods are already presented in Sections V-B and V-C. MF operates by dividing the samples in two clusters/states - one for free flow and another for the congested states. The threshold between the two states correspond on the point where the first order derivatives of the flow/density have inverted inflexion points (i.e. maximum/minimum, respectively). RMF operates in a similar way by considering the same sample filtering technique introduced to RGP in Section V-B. These 7 methods were evaluated with respect to their robustness to noise distributions and unreliable measurements. The intuition behind such idea is that a reliable Context Learning method should provide the same threshold (i.e. Critical Density value) when fed by different sample set that were, however, collected from the same scenario/section/sensor. To do that, we suggest to compute the following Coefficient of Critical Density Variation (CCDV):

\[
CCDV = \frac{|\rho_{C}^{T} - \rho_{C}^{S}|}{\rho_{C}^{T} + \rho_{C}^{S}}
\]

where \(\rho_{C}^{T}\) and \(\rho_{C}^{S}\) denote the critical density in training and test sets, respectively. Two different testbeds were used to evaluate these methods: 5-fold Cross Validation with 100 repetitions (SCV-100) and Noisy Samples (NOI). The first testbed is a traditional cross validation with repetition process which envisions to use all the samples in both roles of training and test sets. The second one operates in an holdout fashion, by taking into account the entire dataset as training once and then testing each method in the same samples artificially perturbed by Gaussian noise.

All the experiments were conducted using Matlab and R. The algorithm selected to implement the clus function was the Gaussian Mixture Models, using the R package MClust. The parameter setting employed is described in Table III. The hyperparameters of GP were automatically computed by the functions within the Matlab package, i.e. GPML. LPC used the implementation from the R package LPCM.

B. Results

The (i) Sensor Pruning stage allowed us to disregard 10% of the total sensors data in our experiments by excluding all the sensors that were in total malfunction or that exhibited an anomaly for, at least, 1h. Fig. 8 presents the results of our cluster-based fault detection framework. Fig. 7. illustrates the modus operandi of the fuzzy system on detecting anomalies in our sensors.

Table IV contains the main results of the two experiment testbeds with respect to the CCDV.

To better understand the reasons behind such results - as well the methods behavior - we depicted two additional Figures: Fig. 9 illustrates a comparison between the critical...
This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

### Table IV

<table>
<thead>
<tr>
<th>Method</th>
<th>5CV-100</th>
<th>NOI</th>
</tr>
</thead>
<tbody>
<tr>
<td>MF</td>
<td>09.72% ± 05.28%</td>
<td>40.84% ± 15.72%</td>
</tr>
<tr>
<td>RGP</td>
<td>11.96% ± 12.53%</td>
<td>56.99% ± 10.46%</td>
</tr>
<tr>
<td>GP</td>
<td>23.05% ± 11.81%</td>
<td>56.62% ± 10.41%</td>
</tr>
<tr>
<td>LPC</td>
<td>17.65% ± 07.34%</td>
<td>04.77% ± 08.83%</td>
</tr>
<tr>
<td>WE</td>
<td>16.35% ± 09.81%</td>
<td>06.91% ± 08.73%</td>
</tr>
<tr>
<td>MF</td>
<td>12.96% ± 08.48%</td>
<td>05.73% ± 08.73%</td>
</tr>
</tbody>
</table>

---

**Discussion**

1) **On Detecting Malfunctioning Sensors:** The spectral clustering depicted in Fig. 8 illustrates well the relationships between Faulty and Normal sensors, which resulted on the exclusion of roughly 10% of them from our experiments. The malfunction of these sensors was confirmed - by a margin of 90%+ - through a direct inspection of the sensor output by a domain expert. This step is critical in any traffic system, since a malfunctioning sensor may tamper the functioning of the entire system.

To confirm if the results on Table IV are statistically significant, we employed a Nemenyi test with a significance level of $\alpha = 0.05$. The Nemenyi test is performed to compare if the results of a given procedure differ significantly of all the remaining ones. The obtained results are depicted in Fig. 11 resorting to the Critical Diagram.

---

Fig. 7. Illustrative Example of the outputs produced Fuzzy Quality Indicators (y-axis) throughout 24h for a particular sensor. Note the behavior of $I_q$ on the rush hours.

Fig. 8. 2D-Projection of the clusters obtained with our framework. Three clusters were flagged as having faulty values.

Density threshold found by three methods (our own, RGP, and the SoA ones, GP and LPC) on noisy/non-noisy data, while Fig. 10 performs the same comparison with respect with the learning function approximation, where we note that the RGP provides a more reliable approximation of the FD, without introducing artifacts visible for example using the GP or LPC methods with noisy data.

To confirm if the results on Table IV are statistically significant, we employed a Nemenyi test with a significance level of $\alpha = 0.05$ on both experimental setups: NOI and 5-CV 100. The Nemenyi test is performed to compare if the results of a given procedure differ significantly of all the remaining ones. The obtained results are depicted in Fig. 11 resorting to the Critical Diagram.

Fig. 9. Critical Density (annotated with circle markers) learning with noisy data versus the not noisy data using GP, RGP and LPC on a particular road section. The change of the critical density point in the case of LPC is evident.

Fig. 10. Learning function approximation with noisy data (N) versus the not noisy data (NF) using GP, RGP and LPC on a particular road section.
ME ranks first being closely related to RGP and LPC (i.e. not having results differing significantly from those). Here, more basic methods - such as MF and RMF - are heavily penalized, whereas MF is listed as the worst one.

Figs. 9 and 10 aim to explain why these results were obtained. In Fig. 9, it is visible how a critical density value computed based on a LPC curve can change with noise while compared with RGP. Fig. 10 confirms the same trend: the critical density estimated by the LPC and RGP on non-noisy/noisy data has a variation of 25.9% and 2.5%, respectively.

In the authors opinion, these results clearly highlight the contribution of the proposed ensemble on guaranteeing accurate results in this particular traffic engineering task.

D. Potential Deployment

The proposed learning framework for AIP (see Fig.1) can be deployed in any Traffic Management System where an automated incident detection system is already in place with high expectations in terms of performance. A typical case is when some parts of the operational control are also automated (e.g. speed control with Variable Message Sign (VMS)). It is desirable that such system has as less failure points as possible. The benefit of having such system in place is that a malfunction in a reasonably small part of the sensors of the network - which is common - will not necessarily compromise the functioning of the entire system.

Moreover, the data quality indicators computed for each sensor by this framework can be directly used in traffic control or in advanced mitigation strategies that are able to exploit the dynamic variation of the reliability of the sensor measurements.

VIII. Conclusions

This paper presented a framework able to transform raw traffic flow data into an informative input to ML-based AIP. We do that in two distinct stages: (i) filtering data from fully/partially malfunctioning sensors and (ii) learning context-aware thresholds to define congestion/free flow states. To do it so, we proposed a semi-supervised ML pipeline that aims to reduce the variance of the critical density (i.e. threshold) computation through a basic committee machine up to 30%. These results were confirmed using experiments run over a real world case study in Asia.

As future work, the authors propose to explore the following three research lines: (2) the usage of supervised learning by having information about anomalous sensors/congestion events; (2) the usage of active learning, by including the possibility of request and learn human labeling of events/sensors; (3) the injection of artificial anomalies on the sensor measurements (instead of simple Gaussian noise) - commonly to what is State-of-Pratice in many common testbeds in outlier detection [39].

REFERENCES


Francesco Alesiani received the Ph.D. degree in electrical and telecommunication engineering in 2004. He is currently a Senior Researcher with the Intelligent Transportation System Group, NEC Laboratories Europe. He is involved in several European and national projects comprising GNSS technology, wireless sensor network, and intelligent transport system. His research interests includes linear, not linear and statistical analysis and optimization, network optimization, and control theory applied to intelligent transportation systems.

Luis Moreira-Matias received the M.Sc. degree in informatics engineering and the Ph.D. degree in computer science (majoring in machine learning) from the University of Porto in 2009 and 2015, respectively. He is currently a Senior Researcher with NEC Laboratories Europe, Heidelberg, Germany, where he is fortunate to lead AI-based research lines. His interests include (automated) machine learning, data mining, and predictive analytics in general. Academically, he has authored 40+ high-impact peer-reviewed publications on related topics. He won the International Data Mining Competition held at the Research Summer School, TU Dortmund, in 2012. He served for the Program Committee and/or as an Invited Reviewer for multiple high-impact research venues such as KDD, AAAI, IEEE TKDE, ESWA, ECML/PKDD, and KAIS. He was fortunate to be invited to give 10+ keynote talks worldwide, ranging locations from Brisbane, Australia, to Las Palmas, Spain.

Mahsa Faizrahnemoon received the Ph.D. degree in applied mathematics from the University of Gothenburg/Chalmers University of Technology, Gothenburg, Sweden. She is currently pursuing the Ph.D. degree with the Hamilton Institute, National University of Ireland. She was an Intern with NEC Laboratories Europe, for six months. She is currently an Industrial Post-Doctoral Researcher with Simon Fraser University, Vancouver, Canada. She was involved in several European projects. She received the Best Scientific Paper Award from the ITS Congress, Dublin, in 2013.