Validation Study of Risky Event Classification using Driving Pattern Factors

German Castignani, Thierry Derrmann, Raphael Frank, Thomas Engel
University of Luxembourg - Interdisciplinary Centre for Security, Reliability and Trust (SnT)
4, rue Alphonse Weicker - L-2721 Luxembourg City, Luxembourg
Email: {firstname.lastname}@uni.lu

***Abstract*—In recent years, due to the increasing sensing capabilities of mobile devices, smartphones have become a suitable solution for telematics systems. By combining multiple sensors, GPS information and environmental data, smartphones can be used to detect abnormal driving events that are used to compute driving scores. In this paper we propose a Multivariate Normal model for abnormal driving events detection. This model takes as input smartphone motion sensors and GPS data and detects abnormal driving maneuvers that are classified as events of three different classes: acceleration, braking and cornering. Based on these events, a driving score is computed. In order to validate the reliability of the computed scores, we propose a correlation analysis of the driving score against multiple well-known driving pattern factors proposed in the literature.**

I. Introduction

Usage-based Insurance (UBI) and Fleet Management arise as the main applications for driver monitoring. In the past, telematics solutions focused on black-boxes, that remotely reported driving data to external entities without giving any feedback to the drivers. With the proliferation of smartphones with high sensing capacities, telematics systems are evolving in the direction of smartphone-based applications that can work completely standalone or in combination with a wireless interface to the vehicle (e.g. OBD-II dongles). Examples of such applications include Automatic [1], Dash [2] or Enerfy [3], which are publicly available for Android and iOS. The primary objective of these systems is to monitor the location and usage of the vehicles and motivate the driver to behave more efficiently (e.g., reduce the energy footprint, reduce travel time). In this paper, we focus on studying the capacity of smartphones to detect abnormal driving events using internal motion sensors and GPS data. However, the heterogeneity of smartphone hardware requires that detection algorithms adapt to different sensing capacities, i.e., sampling rate, resolution and signal quality. We introduce a novel adaptive driver profiling method that relies on a Multivariate Normal distribution (MVN) to build a statistical model of the user's driving characteristics. We first introduce a set of normal features obtained by fusing the output of different mobile phone sensors which can potentially be used to detect and classify lateral and longitudinal driving maneuvers. Then, we propose a risk function that is used for the generation of normalized driving scores.

In order to validate the reliability of the event classifier, we have studied the correlation between driving pattern factors defined in the literature [4] [5] and the computed driving scores over a big data set of driving traces that we have collected during five months, from more than three thousand different drivers over a distance of almost 3.5 million kilometers. This driving data, coming from drivers using iOS and Android smartphones, has been collected in a safe driving campaign done in collaboration with a local insurance company.

The paper is organized as follows, after introducing the related work in Section II, we introduce our method for abnormal driving event detection in Section III. In Section IV we introduce the driving pattern factors and we analyze the correlation to the driving score computed using our abnormal event classifier. Finally in Section V we conclude the paper.

II. Related Work

Eren et al. [6] propose a driver profiling method to differentiate between risky and safe drivers. Smartphone accelerometer, gyroscope and magnetometer are considered to detect driving events (e.g., sudden maneuvers, aggressive steering, braking or acceleration). The start and end time of each event are computed using a moving average algorithm and empirical thresholds. Then, the similarity of each detected event to previously collected data sets for different type of maneuvers using Dynamic Time Warping (DTW). Bayesian classification is used to decide whether a driver is risky or safe. The authors present an evaluation study including fifteen drivers carrying iPhones for fixed departure and arrival points. In the proposed evaluation a successful classification rate of $93.3\,\%$ is shown. However, using the proposed system to compute predefined event data sets appears to be unfeasible for an heterogeneous sensing environment, including different types of vehicles and smartphones.

Similarly, Paefgen et al. [7] present a number of experiments that compare the performance of smartphone-based driver profile algorithms and fixed Inertial Measurement Units (IMU). The authors collected driving samples from gyroscope, accelerometer and GPS from Android and iOS devices. As a precondition, before sample collection starts, the user manually sets the principal direction of the car in a so-called calibration process. Acceleration, braking and steering events were declared if the sensor data surpassed predefined thresholds (e.g., an accelerometer output of $0.1g$ for acceleration and braking events and $0.2g$ for steering). The authors present a measurement study to compare event detection using smartphone sensors to those from a fixed IMU. They observe that the obtained event count distribution matches different statistical distributions in smartphones and IMUs; this is due mainly to variations in smartphone-to-car fixing and positioning, which

978-1-4673-9907-4/15/$31.00 2015 IEEE

can dramatically affect the pre-established thresholds. However, the authors find weak correlations between smartphones and IMU-based events and describe some possible solutions to enhance smartphone event detection, including a dynamic calibration process [8].

Johnson et al. [9] propose MIROAD, a DTW-based driver profile algorithm, using smartphone sensors and GPS. The authors evaluate the performance of different sensor fusion sets to detect lateral and longitudinal movements. By evaluating over 200 driving events, the authors show that the sensor fusion set composed of the x-axis (i.e., gravity axis) rotation rate, y-axis (i.e., lateral movements) acceleration and pitch, provides the best classification performance using DTW.

Castignani et al. [10] describe a set of experiments to analyze the performance of smartphones to detect risky driving events. In particular, they introduce a sensor fusion algorithm and a Fuzzy System that detects aggressive steering, acceleration and braking events.

As previously introduced, the proliferation of inexpensive smartphone-to-car OBD-II interfaces using Bluetooth has motivated the development of smartphone driver profiling and eco-driving tools based on real-time vehicle data. Commercial platforms like Automatic [1], Dash [2] or Enerfy [3] are currently available in the market and focus on providing basic driving assistance tools with branded OBD-II adapters. In the same order, Meseguer et al. [11] describe the DrivingStyles platform (available for Android), which takes into account OBD-II acceleration, speed and engine RPM data to characterize driver style and infer the road type (e.g., urban, suburban, highway). In order to do this, they use an Artificial Neural Network approach that achieves road type classification with 98% accuracy and driver style recognition with 77% accuracy. Focusing primarily on eco-driving, Araujo et al. [12] developed a smartphone application that combines GPS and OBD-II data. The application is able to suggest switching off the engine, shifting gears earlier or decelerating. Average, minimum and maximum values for speed, acceleration and fuel consumption are considered as input variables of a proposed Fuzzy System. They evaluate and validate their algorithms using a mobile platform and several experiments driving a single car.

Various solutions in the literature consider the use of smartphone cameras to profile drivers. The CarSafe [13] project uses front and rear smartphone cameras to monitor face direction, eye states, following distance and lane trajectory. Based on this input, they developed a set of image recognition and machine learning algorithms to identify drowsy and inattentive driving, tailgating, lane weaving/drifting and careless lane changes. Veeraraghavan et al. [14] also explore the potential of supervised and unsupervised learning algorithms to detect risky activities while driving (e.g., operating a mobile phone, eating, or interacting with the dashboard).

III. MVN MODEL FOR DRIVER PROFILING

A. Model definition

In order to detect risky driving maneuvers and provide driver profiles, we use a Multivariate Normal (MVN) model [15]. Using MVN models allows handling the heterogeneity of mobile devices and vehicles, since the model

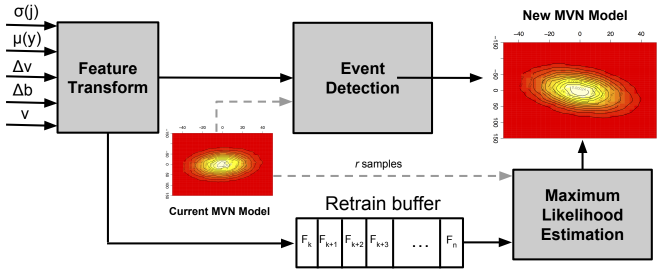


Fig. 1: Event detection and retrain phase

needs to adapt its parameters to each individual context. Traditional fixed threshold or supervised Machine Learning classification techniques (e.g., Neural Networks, Naive Bayes, SVM) require previous knowledge of the training labels, which is an important limitation if the system needs to be suitable for a diversity of devices and vehicles. An MVN-based model provides a high flexibility by dynamically adapting to different driving conditions with a periodic update of its parameters. Also, given that it does not require any preliminary knowledge on the driving events it has to detect, it can be applied to any type of vehicle and mobile device. The advantages of using MVN are the following: (i) There is no need to establish rule sets or thresholds to detect risky maneuvers. The probability of an observation enables the assessment of driver behavior in a continuous fashion rather than binary classification (i.e., distinguishing between a risky and a non-risky driving event). (ii) The model can be estimated efficiently on a mobile phone and can be continuously retrained to improve the parameter estimates. (iii) It is easy to interpret the model, since the sampling distribution probability density can easily be represented visually, e.g. by contour plots. (iv) Some of the input variables such as GPS acceleration and bearing, are normally distributed and thus lend themselves naturally to this kind of model.

In the proposed implementation, the model acts in two phases. In the first phase, the mobile device collects input variables that are then transformed into a two-dimensional feature space representing the axes of movement of the vehicle. These features are then stored in a training buffer that will finally serve to generate the first MVN model using the Maximum Likelihood Estimator (MLE). This first model will then be adjusted in sequential retraining phases, as shown in Fig. 1, at the same time as it is used for risky event detection.

The model's features consider common motion sensor variables from smartphones. The first component of the input vector $o = [\sigma(j), \mu(y), \Delta v, \Delta b, v]$ is the standard deviation of the jerk, $\sigma(j)$. The jerk is calculated as the time derivative of the norm of the acceleration vector using the device's accelerometer. This variable is able to provide a better fit to aggressive driving maneuvers than the raw acceleration, which can be greatly affected by vibrations due to speed. Then, we consider the average yaw rate, $\mu(y)$, the angular velocity measured with the device's magnetometer. Finally, we consider GPS input: the speed, v, the speed variation, Δv, and the bearing variation, Δb. In both the initial phase and the detection/retrain phases, the input vector refresh rate is fixed at a rate of 10 Hz in order to provide an accurate short-term view of potential events. The proposed MVN model considers a two-dimensional feature space $\mathbf{X}$. We define a feature $\mathbf{x} \in \mathbf{X}$ as a function of the input data o having the following properties: i) extreme values of the feature represent risky or abnormal driving maneuvers, ii) the feature's distribution can be approximated by a Gaussian distribution. For the proposed study, we have considered $\mathbf{x} = [\sigma(j) \cdot \Delta v \cdot v, \mu(y) \cdot \Delta b \cdot v]$. These variables have been chosen after several experimental trials [16] [17].

During the retraining phase (see Fig. 1), the model is built by considering n recent samples following the previous model and a number of samples r that are re-sampled from the previously computed model. In the retraining phase, there is a trade-off between the performance of the risky event detection and the adaptability of the model. In other words, even though a frequent retraining strategy (i.e., a low n and a low r) results in a rapid adaptation of the model to new conditions (e.g., a change in the driver's driving style or road conditions), it can greatly impact the quality of event detection by incorporating short-term data samples into the model. The objective is always to avoid updating the model with invariant input samples, i.e., samples that reflect a monotonous driving behavior, like driving on a straight line at constant speed. In order to verify that the model update is useful, we compute the determinant of the covariance matrix of the new n data samples. We propose a criterion based on the determinant of the covariance matrix of the newly-collected samples (n) and the current model resampled data (r): we simply compare the ratio n/r to a predefined limit. This predefined limit is empirically set after a sufficient number of validation tests. This rule guarantees that we do not degrade the sampling distribution by fitting data with very low variance.

B. Event detection

The proposed model outputs a metric for the severity (or risk factor) for a given input. To this end, we propose a cut-off quantile value, Q_{lim}, which marks the quantile limit between very common observations and those we consider to be of interest with regard to anomaly detection. Q_{lim} defines a cut-off probability value, p_{lim}, which can be used to classify the samples. Let p_{lim} be the model-dependent inverse CDF value of Q_{lim}, and $Q(\mathbf{x})$ the (approximate) quantile of the feature transform $\mathbf{x}$ of observation o, we can then propose the severity metric of Eq. 1.

$$s(\mathbf{x}) := \begin{cases} 0 & \text{if } p(\mathbf{x} \mid \theta) \geq p_{lim} \\ 1 - \dfrac{Q(\mathbf{x})}{Q_{lim}} & \text{if } p(\mathbf{x} \mid \theta) < p_{lim} \end{cases} \tag{1}$$

Eq. 1 defines a normalized metric of how anomalous an observation is. In this paper, we use $Q_{lim} := 0.01$. Note that the value of Q_{lim} (and correspondingly, p_{lim}) only shifts the limit quantile of observations of interest. If a lower value of Q_{lim} is chosen, more observations will be considered to be anomalies but with lower severities.

C. Score computation based on severity

The severity is a metric of how abnormal an event is. In order to analyze driver riskiness, we need to explore how the observations corresponding to an event have evolved in the two-dimensional feature space. Using the severity metric $s(\mathbf{x})$ defined above, together with the Mahalanobis distance, we derive a risk function in Eq. 2.

$$R(\mathbf{x}) := \frac{s(\mathbf{x})}{2} \left(\frac{1 - \cos \alpha}{2} + \frac{2}{\pi} atan \left(D_m(\mathbf{x_i}, \mathbf{x_{i-1}}) \right) \right) \tag{2}$$

Eq. 2 is motivated by the idea that large variations in the feature space during an anomaly should be more strongly penalized. Regular anomalies manifest themselves as a temporal cluster of similar values, whereas more extreme maneuvers display more variability and cover larger distances within the feature space. Thus, we use the arctangent of the Mahalanobis distance and the relative angle α between consecutive samples $(\mathbf{x_{i-1}}, \mathbf{x_i})$ in the feature space. In particular for the relative angle, we evaluate its Haversine function. These functions were chosen to obtain a normalized risk metric. These values represent the similarity of two consecutive samples and are weighted using our severity metric ($s(\mathbf{x})$). Thus, the risk function reflects the trajectory of the anomaly within the feature space. The risk function is accumulated over a trip and can serve as a basis for driver scoring. Using this metric, risky events of a long duration are more comparable to short-term events (e.g., lane changes), as similar consecutive anomalies are filtered by the distance and angle coefficient.

In order to provide a driver score, we compute the risk density as the quotient sum of $R(\mathbf{x})$ for the whole trip and the trip distance. Then using a predefined linear function, we convert the proposed risk density to a score between 0 (for high risk densities) and 100 (for very low risk densities).

IV. Validating Scoring Using Driver Pattern Metrics

A. Data collection campaign

In March 2015, we have launched a mobile application that embeds the proposed MVN-based model for event detection and scoring. During the first six months, we have collected 195.000 trips from 5.350 distinct drivers, accumulating a total of 3.5 million kilometers of driving data. Using our MVN model, we detected more than 1 million events and computed around 0.5 million model updates. In the mobile application, an MVN model is built and retrained on each mobile device. Users record their driving and use the MVN model to classify driving events, as described in Section III. The output of the MVN classifier is sent to a remote server at the end of each trip. The server then calculates a drive score based on the density of MVN events, i.e., the ratio between the risk computed using Eq. 2 and the distance driven. Trips with negligible risk density (lower than a constant minimum threshold) are assigned a drive score of 100 points drive score. Drive score linearly decreases (down to zero) with increasing risk density. This drive score is then adjusted using a variety of contextual information, including historical weather and daylight information in order to more heavily penalize events occurring in bad environmental conditions. For every recorded trip, the driver gets a number of game points that are accumulated after each trip. The number of game points of each trip is proportional to the drive score, the distance driven and the frequency of usage of the application. As a result, after six months of usage of

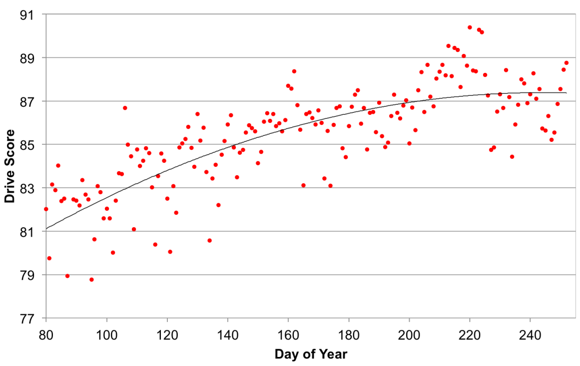


Fig. 2: Evolution of average score

the application, we were able to observe the evolution of the average score for the whole community of users. In Fig. 2, the average score calculated over the total number of users is shown for the first five months of the data collection campaign. We can observe that the average score improves over time (quadratic trend line with $R^2 = 0.63$).

B. Driver pattern metrics

In order to validate the representativeness of the computed driver score based on the MVN model, we have studied well-defined driver pattern metrics over the set of collected data. These metrics have been previously defined in the literature [4] [5] and are summarized in this section. In order to analyze the correlations, for each trip collected in the data set, we obtain the computed drive score, the distance driven and the speed time series. The proposed driver pattern metrics are obtained through calculation over the speed time series and trip distance. Then, we compute linear models for each metric and drive score.

1) Positive Kinetic Energy (PKE): Positive Kinetic Energy (PKE) represents the driver behavior during the acceleration process. As shown in Eq. 3, PKE is computed as the density over distance of the sum of the squared speed variation during acceleration phases. v_i indicates the speed as the beginning of the acceleration phase while v_f indicates the speed at the end of the acceleration phase. As stated in [4], a high PKE is associated with nervous driving, while a low PKE denotes smooth driving.

$$PKE = \frac{\sum v_f^2 - v_i^2}{d} \quad when \ \frac{dv}{dt} > 0 \tag{3}$$

2) Relative Positive Acceleration (RPA): Relative Positive Acceleration (RPA) describes the power demand of the driver. RPA shows high values when the driving pattern considers a high number of high power-demand accelerations, increasing aggressiveness and decreasing fuel efficiency.

$$RPA = \frac{1}{d} \int (v \cdot a^+) dt \tag{4}$$

It is defined as the quotient between the integral of the product of speed and positive acceleration (a^+) over the total trip distance.

3) Extreme Acceleration Factor (EAF): A trip with high Extreme Acceleration Factor (EAF) occurs when a driver surpasses a certain acceleration threshold (i.e., $2.5\,m/s^2$) with a high frequency. EAF is defined as in Eq. 5, where function $C(x)$ denotes the number of occurrences of event x.

$$EAF = \frac{1}{d} \cdot C\left(\frac{dv}{dt} > 2.5\,m/s^2\right) \tag{5}$$

4) Speed Oscillation Factor (SOF): Speed Oscillation Factor (SOF) represents the number of speed variation phases during a trip. It is assumed that trips done with a very low speed variation (i.e., almost constant speed) denote a lower risk. Eq. 6 models speed oscillation, which is computed as the number of occurrences of non-zero second speed derivative.

$$SOF = \frac{1}{d} \cdot C\left(\frac{d^2v}{dt^2} \neq 0\right) \tag{6}$$

5) Deceleration Factor (DF): The Deceleration Factor (DF) is simply computed as the average deceleration value during a trip as shown in Eq. 7

$$DF = \frac{1}{n} \sum a^- \tag{7}$$

6) Factor for Speed (FS): The Factor for Speed (FS), as shown in Eq. 8 indicates the number of occurrences where speed is grater than the maximum allowed speed (130 km/h).

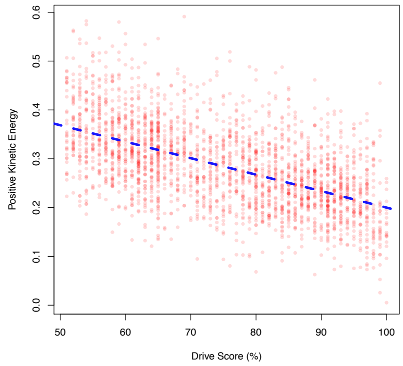
$$FS = \frac{1}{d} \cdot C\left(v > 130\,km/h\right) \tag{8}$$

C. Correlation analysis

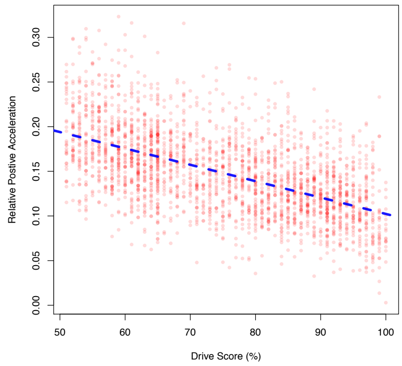
The previous driver pattern metrics have been used in related works to characterize the aggressiveness or smoothness of drivers. In this section, we present a correlation study of those metrics against the computed driver score calculated using our previously introduced model. The objective of this analysis is to conclude on the representativeness of such scores, which are computed using an MVN-based event detector, differently from the driver pattern metrics, which are purely based on speed time series analysis. Fig. 3 illustrates, by the means of scatter plots, the relation between drive score and driver pattern metrics for each individual trip, i.e., each point on the scatter plot represents the driver score of the trip (S) and the driver pattern metric (M) computed over the speed time series of that trip. A linear model is also computed for each correlation. Table I shows the correlation index between the drive score and each driver pattern metric. The correlation index $\rho(S, M)$ between driver score S and driver pattern metric M is calculated as in Eq. 9

$$\rho(S, M) = \frac{E\left[\left(S - E\left[S\right]\right)\left(M - E\left[M\right]\right)\right]}{\sigma_S \cdot \sigma_M} \tag{9}$$

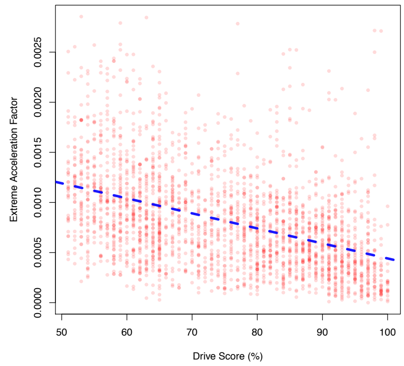
We observe that, among the six proposed metrics, four of them present a good correlation with driver score. Both PKE and RPA, modeling the intensity of positive acceleration processes, shows the best correlation with driver score, with a correlation index of 0.55. This was actually expected, since high positive speed variation implies a strong acceleration



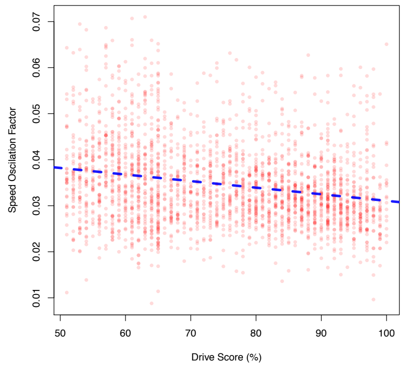
(a) PKE



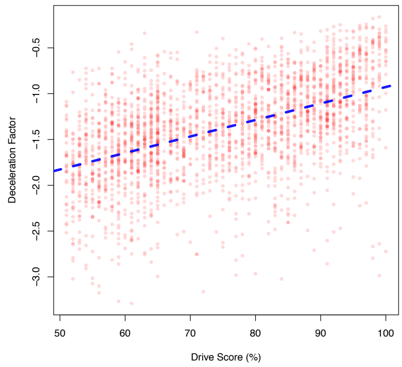
(b) RPA



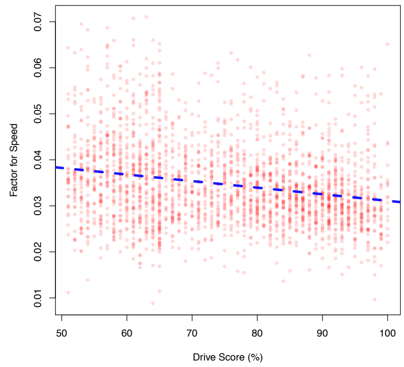
(c) EAF



(d) SOF



(e) DF



(f) FS

Fig. 3: Correlation analysis of MVN-based score against driver pattern metrics

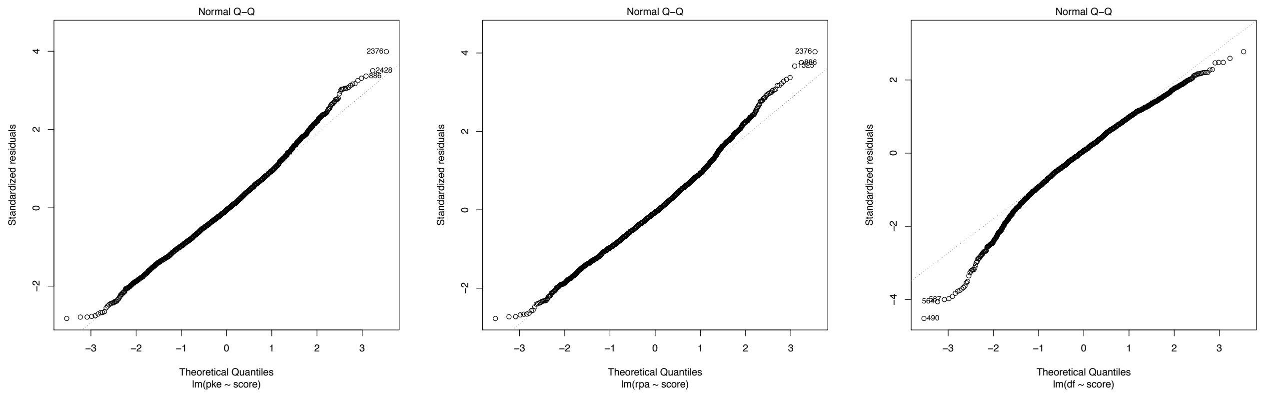
pedal pressure which is labeled by the MVN-based event classifier as a risky event. On a second level, DF and EAF also shows a medium correlation, in particular, average deceleration during trips (DF). Finally, SOF and FS show low correlation. Note that as shown in Fig. 5, SOF shows higher values during peak hours (6 to 9 AM, 4 to 7 PM). If we had observed a high correlation between driver score and SOF, that would have meant that the MVN-based event detection was sensible to traffic conditions, i.e., driving in highly congested roads may negatively impact the drive score, which was not desired. However, this is not observed in our analysis since the correlation index between driver score and SOF is low. In the case of FS metric, a low correlation is observed since fixing a single speed threshold, which is the case in existing driver profiling platforms [1] [2] [3], is not enough to characterize aggressive driving. As illustrated in Fig. 4, for both PKE and RPA metrics, the distribution of the residuals of the linear model fits a normal distribution. In the case of DF, the third best correlated metric shows a slight left skewed distribution.

V. Conclusion and perspectives

We introduced in this paper an MVN model to detect risky driving events using smartphone sensors and GPS data. The proposed system dynamically builds and maintains an MVN model for each device and driver in order to adapt to changing driving conditions and environments. Using the density of abnormal event's severity, we compute a risk metric that is used to provide users with a driver score between 0 and 100. We have implemented our system for Android and iOS and collected driving traces of more than 5000 drivers over a total driving distance of more than 3.5 million kilometers. Using the collected traces during the first five months of the campaign, we have studied the representativeness of the drive score by analyzing the correlation against well-known driving pattern metrics. We have observed in our study that metrics based on speed time-series (PKE, RPA or DF) are highly correlated to the drive score computed with the MVN-models. Also, we

TABLE I: Correlation index

M	$\rho(S, M)$
RPA	0.553
PKE	0.552
DF	0.491
EAF	0.397
SOF	0.234
FS	0.233



(a) PKE (b) RPA (c) DF

Fig. 4: Normal Q-Q plots for PKE, RPA and DF

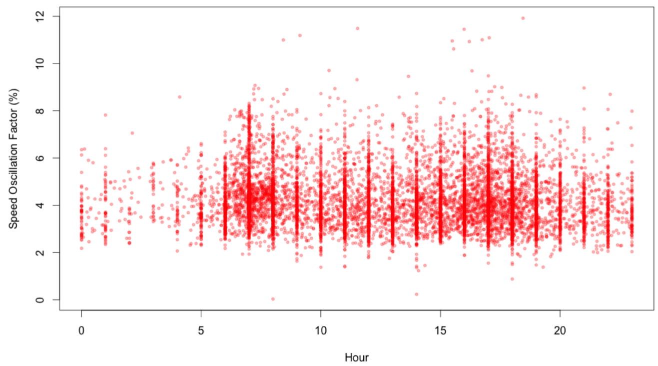


Fig. 5: SOF during time of day

observed very low correlation of driver score against traffic-jam metrics (like SOF), which confirms that the MVN-model is able to adapt to different environmental conditions.

VI. Acknowledgment

This project is supported by the Luxembourg National Research Fund (FNR POC/13/07).

References

[1] Automatic Labs, Inc., URL: https://www.automatic.com/, 2014.

[2] Dash Website, URL: https://dash.by/, 2014.

[3] Enerfy Driving, URL: http://www.enerfydriving.com/, 2014.

[4] C. Andrieu and G. Saint Pierre, "Evaluation of ecodriving performances and teaching method: comparing training and simple advice," EJTIR, vol. 14, no. 3, pp. 201–213, 2014.

[5] E. Ericsson, "Independent driving pattern factors and their influence on fuel-use and exhaust emission factors," Transportation Research Part D: Transport and Environment, vol. 6, no. 5, pp. 325–345, 2001.

[6] H. Eren, S. Makinist, E. Akin, and A. Yilmaz, "Estimating driving behavior by a smartphone," in 2012 IEEE Intelligent Vehicles Symposium (IV), Jun. 2012, pp. 234 –239.

[7] J. Paefgen, F. Kehr, Y. Zhai, and F. Michahelles, "Driving behavior analysis with smartphones: Insights from a controlled field study," in Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia, ser. MUM '12, 2012.

[8] J. Almazan, L. Bergasa, J. Yebes, R. Barea, and R. Arroyo, "Full auto-calibration of a smartphone on board a vehicle using imu and gps embedded sensors," in Intelligent Vehicles Symposium (IV), 2013 IEEE, June 2013, pp. 1374–1380.

[9] D. Johnson and M. Trivedi, "Driving style recognition using a smartphone as a sensor platform," in Intelligent Transportation Systems (ITSC), 2011 14th Int. IEEE Conference on, 2011, pp. 1609–1615.

[10] G. Castignani, R. Frank, and T. Engel, "Driver Behavior Profiling using Smartphones," in Intelligent Transportation Systems (ITSC), 2013 16th International IEEE Conference on, 2013.

[11] J. Meseguer, C. Calafate, J. Cano, and P. Manzoni, "Drivingstyles: A smartphone application to assess driver behavior," in Computers and Communications (ISCC), 2013 IEEE Symposium on, July 2013.

[12] R. Araujo, A. Igreja, R. de Castro, and R. Araujo, "Driving coach: A smartphone application to evaluate driving efficient patterns," in 2012 IEEE Intelligent Vehicles Symposium (IV), Jun. 2012, pp. 1005 –1010.

[13] C.-W. You, N. D. Lane, F. Chen, R. Wang, Z. Chen, T. J. Bao, M. Montes-de Oca, Y. Cheng, M. Lin, L. Torresani, and A. T. Campbell, "CarSafe App: Alerting Drowsy and Distracted Drivers Using Dual Cameras on Smartphones," in Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services, ser. MobiSys '13, 2013.

[14] H. Veeraraghavan, S. Atev, N. Bird, P. Schrater, and N. Papanikolopoulos, "Driver activity monitoring through supervised and unsupervised learning," in Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE, Sept 2005, pp. 580–585.

[15] Andrew Ng, "Multivariate Gaussian Distribution (tutorial)," URL: http://cs229.stanford.edu/section/gaussians.pdf, 2014, Last Accessed in July 2015.

[16] G. Castignani, R. Frank, and T. Engel, "An Evaluation Study of Driver Profiling Fuzzy Algorithms using Smartphones," in Workshop on Vehicular Communications and Applications (VCA'13), collocated with IEEE ICNP'13, 2013.

[17] G. Castignani, T. Derrmann, R. Frank, and T. Engel, "Driver behavior profiling using smartphones: A low-cost platform for driver monitoring," Intelligent Transportation Systems Magazine, IEEE, vol. to appear, Fall 2014.