Araujo et al. "Towards a crime hotspot detection framework for patrol planning." *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*. IEEE, 2018.

BibTeX:

```
@INPROCEEDINGS{araujo2018towards,
  author={A. {Araujo} and N. {Cacho} and L. {Bezerra} and C. {Vieira} and J. {Borges}},
  booktitle={IEEE 16th International Conference on Smart City},
  title={Towards a Crime Hotspot Detection Framework for Patrol Planning},
  year={2018},
  pages={1256-1263}
}
```

Towards a Crime Hotspot Detection Framework for Patrol Planning

Adelson Araújo Jr., Nélio Cacho,
Leonardo Bezerra, Carlos Vieira
Universidade Federal do Rio Grande do Norte,
Natal, RN Brazil
{adelsondias@ppgsc, carlosv@,
leobezerra@imd, neliocacho@dimap}.ufrn.br

Julio Borges
Karlsruhe Institute of Technology
Karlsruhe, Germany
borges@teco.edu

***Abstract*—By monitoring crime incidence with quantitative techniques, many studies have shown that it is possible to improve decision making through pattern recognition and prediction. In a smart city scenario, such approaches can be used to compose analytical background to improve resource allocation. This work presents a novel framework to improve patrol planning that precisely provides places and times that are likely to be more dangerous than short-term average using a portfolio of machine learning algorithms. Our approach follows an algorithm-as-a-service architecture (AaaS), providing insights to existing public safety systems and platforms. The service comprises the broader ROTA framework, a robust public safety platform devised for the ongoing smart cities initiative of Natal, Brazil. Results of an experimental evaluation provided insights about spatial granularity effects on the performance of the estimators adopted. Furthermore, an evaluation on algorithm selection demonstrates its outcomes on the hotspot detection task.**

***Keywords*—crime prediction, machine learning, patrol planning.**

I. Introduction

Data is the core material of artificial intelligence, and is there a social organism that produces more data than cities? In such context, smart cities arises from technological ubiquity through information gathering infrastructures to acquire knowledge of their own complex organization. As Caragliu et al. [1] state, a city becomes smart when it applies human and social capital together with Information and Communications Technology (ICT) sustainable growth to wisely manage urban and natural resources, through participatory governance. This culture makes the government think in an innovative way, opening itself to technological and scientific trends. Concerning public safety, previous studies have shown that criminal rates can be explained quantitatively by analytical efforts to measure aggregations and can increase the effectiveness of policing [2]–[4]. A strong body of research focuses on place-centric approaches to advise hotspot patrolling as a crucial pillar in the implementation of predictive policing [5]–[7].

In fact, some cities are applying predictive policing for smart city initiatives. For instance, the IEEE Smart City Initiative [1] of Natal (in Brazil) is deploying a full-fledged platform, named ROTA [8]–[10], that aims to improve public safety by integrating several information systems from different law enforcement agencies. ROTA provides a set of mobile applications that enhance information sharing aiming at providing actionable, timely intelligence, as well as to support police operations. Although the platform provides concrete benefits, ROTA does not support structured approaches to predict criminal activities and preferable places to patrol. Predicting hotspots of criminal activities allows police departments to deploy their task forces where and when they are most needed, patrolling certain geographical areas in certain times of the day where specific crime categories are most probable, proactively fighting crime in a *data-driven* manner.

In this paper we propose a predictive framework for hotspot detection that considers spatial, temporal and categorical configurations of crime to precisely provide places and times that are likely to be more dangerous than short-term average, using a portfolio of machine learning algorithms. We have implemented it for Natal and assessed the performance of the models using different spatial granularity levels. For practical purposes, this framework is implemented as a service, following the algorithm-as-a-service (AaaS) architecture. The framework is able to handle multiple data sources, offers multiple options of spatial granularity and feature engineering, and implements machine learning algorithm selection. The outcome of the framework provides an useful artifact for patrol planning, called program card (i.e. a list containing priority locations that police vehicles must patrol).

The main contributions of this work are summarized below:

1) A carefully designed crime prediction framework for smarter public safety planning that considers multiple data sources, spatial configuration, feature engineering and machine learning algorithm selection.
2) An improvement in the analytical module of ROTA platform to support hotspot detection for patrol planning scheduling tasks.
3) A discussion about the need of spatial configuration experiments to improve hotspot detection accuracy.

This paper is organized as follows. Section II describes the problem tackled in the smart city initiative of Natal. Section III provides a background on the predictive policing literature.

[1] https://smartcities.ieee.org/affiliated-cities.html

TABLE I
USE OF PREDICTIVE TECHNOLOGIES FOR CRIME ANALYSIS AND PREDICTION

Problems	**Predictive Analytics Approaches**
Identify areas at increased risk using historical crime data	Hotspot identification models [11], Crime Incidents Prediction [12], Spatiotemporal analysis methods [5], [13]
Using a range of additional data (911 records, economics...)	Classification, and clustering models [14]
Accounting for increased risk from a recent crime	Near-repeat modeling: increased risk in areas immediately surrounding a recent crime [15]
Determine when areas will be at most risk of crime	Spatiotemporal analysis methods [5], [13]
Identify geographic features that increase the risk of crime	Risk-terrain analysis [6], [16]

Section IV presents the proposed framework. Section V evaluates the performance of the prediction algorithms adopted. Finally, Section VI concludes and discusses future work.

II. THE SMART CITY INITIATIVE IN NATAL

Natal, a city of Northeastern Brazil, has joined the IEEE Smart City initiative as an affiliated city. This initiative aims to transform Natal into a smart city through the development of systems and applications to bolster the use of IT as means of contributing to improve the life quality of its citizens. Many actions have been implemented [8], [9], [17]–[19].

For instance, one of the solutions developed under the Natal Smart City initiative is ROTA [8], [9], a smart city platform aimed to improve public safety by collecting, integrating, analyzing, and sharing information about occurrences and patrol vehicles. ROTA provides some modules such as: *Patrol Supervisor Module* (ROTA-PSM) [8] and *Patrol Vehicle Module* (ROTA-PVM) [9]. ROTA-PSM is a mobile application used by patrol supervisors to display the real-time position of all patrol vehicles and occurrences. ROTA-PVM is an Android mobile application deployed in the patrol vehicles. The main purpose of ROTA-PVM is to support patrol in its operations, thus making occurrence handling easier and faster.

By adopting ROTA-PVM, the occurrence handling flow no longer requires using radio communication, except when strictly necessary. When an occurrence is registered, a notification appears on the screen of the patrol vehicle's tablet. Upon confirming that the received notification was seen, the policeman is provided with all relevant data regarding the occurrence, such as location, type, suspect's characteristics, and the transcription of the call made by the citizen. During the operation, the patrol officer can notify any change regarding the occurrence, e.g., his/her arrival on scene, the need of an additional displacement, finishing occurrence, and occurrence reports. Another functionalities of the ROTA-PVM is the program card. A program card is a practical policing methodology used by Police to distribute staff in a given area. It consists of a list containing predefined locations (haven, ambush, and crime-prone areas), patrol teams, route plan and staying time at which each police vehicle must patrol as shown in Figure 1. Hence, Patrol officers use patrol vehicles equipped with ROTA-PVM to patrol the assigned beats defined by the program card.

Fig. 1. Program Card in the ROTA-PVM mobile application.

Despite the existence of ROTA-PVM, program card elaboration still made manually, i.e., there is no tool to support the detection of spatial and temporal patterns of criminal activity. Patrol supervisors need to use their knowledge to define the program card for each patrol team that is under their supervision. Hence, detecting hotspots are essential for the definition of the program card. In a previous work [10], we exploit how a web application dashboard would help patrol supervisors in such task, by visualizing criminal incidence estimative to each police district.

Our proposed framework (see Section IV) builds upon this work adding several advantages in terms of prediction's architecture and performance improvement. First, we modeled a component to create and manage the spatial layer of the analysis. Then, a component to extract domain-specific features from the raw time series is included. Another upgrade is the algorithm selection component, that now consider a Portfolio of tuned algorithms. For instance, we tested different spatial discretization configuration reducing the size of the area for the predictions (improving in hotspots' spatial precision). Additionally, we encapsulate the prediction architecture in a web service manner to generalize implementation for other cities or application contexts. Finally, models performance were improved by tackling the classification problem of hotspot detection rather than crime incidence prediction.

III. Background

A. Predictive Policing and Hotspot Detection

As an established concept, predictive policing is well stated by Perry et al. as *"the application of analytical techniques (particularly quantitative) to identify likely targets for police intervention and prevent crime or solve past crimes by making statistical predictions"* [6]. From the point of view of smart city, the practical use of predictive policing can affect directly at least two and, indirectly, more two out of the 8 cluster factors defined by the Chourabi's framework [20] to rank smart cities. For instance, *Management and Organization* is changed in terms that resource allocation and decision making is beneficed with crime incidence estimative for different regions of the city (patrol planning, for instance). The *Technology* skill is directly improved because predictive policing provoke to analyst' research to become a software product and changes IT and statistical cultural issues for decision-making. Indirectly, *People and Communities* benefits with crime reduction [2]–[4] and *Governance* gains when such strategies provide data-exchange and service integration between tactical and operational agents [6]. Table I summarizes related work in this area.

A vast body of research implements predictive policing through hotspot analysis [4]. A hotspot is a spatial entity with high criminal incidence and can be physically represented in many ways. Some authors prefer to represent such zones analogically (for instance, through heatmaps), with Kernel Density Estimation techniques [11], [13] or through Risk Terrain Analysis, including geographical and demographical variables to represent dangerous areas [6], [16]. Others prefer to represent hotspots spatially as a discrete cell in a grid. Although several works uses rectangular grid, Ziehr [12] proposed a K-Means based convex hulls polygonal collections called KGrid [12], adjusted with occurrences coordinates, which is illustrated in Fig. 2 and will further be used in this work. Besides, it's possible to use the police districts or the boroughs as the spatial grid. The grids are relevant to evince hotspots for patrol planners. When implementing them, query the police department for their preference would determine the success of the hotspot detection tool. However, when it is asked for smaller units, as recommended by criminologists [4], artificial grids just as KGrid is more appropriate because K is a free parameter to manipulate lean and precise hotspots.

In temporal domain of granularity, Maciejewski et al. [21] describes their procedure for daily aggregation and Malik et al. [13] used weekly data to conduct their analysis, showing that their model gives better overall prediction results when aggregated over longer time intervals. As for spatial configuration, time window is another parameter to be assessed in a practical way, consulting the police department to the best usage. However, these authors are advocating an average of at least ten crime records per time step in a sub-region in order to avoid having sparse regions with few data. While a spatial resolution level that is too fine (smaller spots) may lead to sparse data input in many regions, a scale that is too coarse can overgeneralize the data and reducing the value and specificity of the prediction results [13].

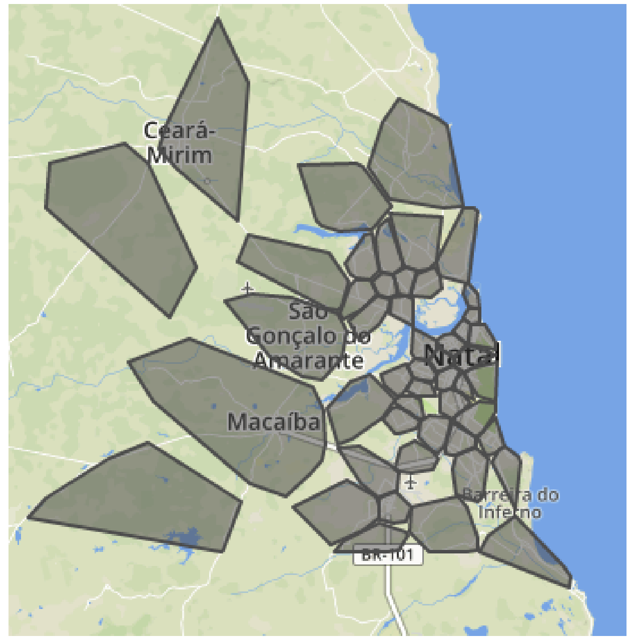

Fig. 2. KGrid of 50 cells

Aiming to estimate future hotspots using past sequential records, it needs to be modeled as a time series problem, which holds some challenging specificities. According to Langkvist et al. [22], time series data has a number of characteristics that make it different from other types of data. He explicits that it contains (i) high dimensionality and noise, (ii) uncertainty of data amount significance, (iii) explicit time dependency and (iv) non-stationarity (mean and variance changes over time). However, this difficulties may be suppressed in some cases with a structured modeling process of feature engineering. As time is an explicit dependency for such problems, features needs to be extracted from past observations, these may be called autoregressive terms. Classical time series analysis frameworks, such as Box and Jenkins approach [23], suggest to take autocorrelation function (ACF) results to track how many autoregressive features would be used. However, past observations are not the only features to be used in time series problems, and often moving averages are brought to the feature union to improve learning. Besides, another sources of features are welcome to improve model effectiveness when temporally variant and some researches include it in their crime hotspot prediction approach, using demographic [7], urban features [24] or even Twitter data collection [11].

In terms of machine learning, hotspot detection and prediction is applicable for both regression or classification tasks, depending on the target label setting provided. As crime incidence is a quantity, regression algorithms are more appropriate when it is preferable to predict how many occurrences will happen in the next time step. A body of researches tackle suck task [10], [12], [13], but it is also relatively hard to be

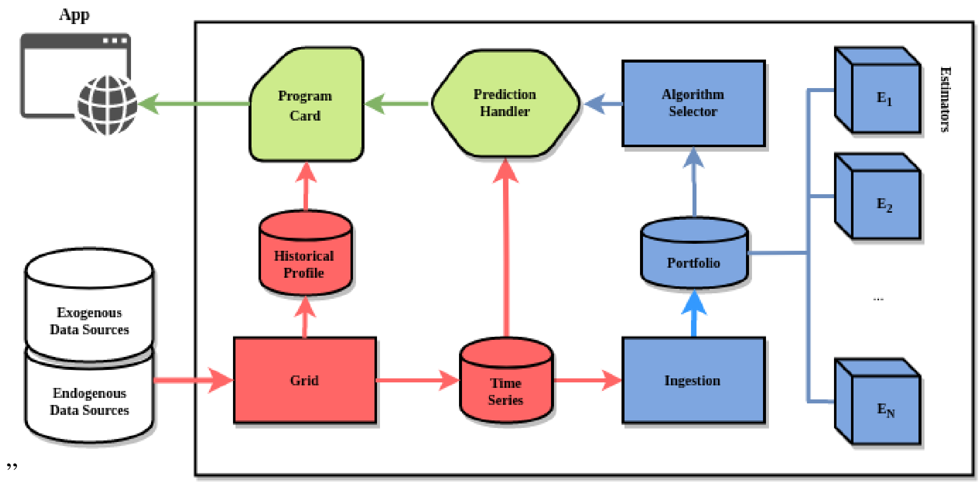

Fig. 3. Prediction framework architecture composed by processing components, data and models stored on disk and a interface for output. The components by red represent data preparation, by blue are related to the prediction and by green the interface between prediction and output.

modeled because time series ”peaks” are as rare as desirable to be predicted. Another strategy is to aggregate incidence ranges in classes to reduce the class unbalance, the cold and hotspot modeling suggested in Bogomolov et al. [7] and Yu et al. [5] were adapted to be used in the implementation described in Section V. As a consequence of that, less classes tend to simplify the problem in order to achieve better prediction, considering their distribution. The framework presented in the next section is not locked on a single machine learning task and it will be parametrized in its components depending on implementation issues.

IV. Prediction Framework

A. General Description

In this section we present the architecture of the proposed framework composing a plethora of functionalities related to hotspot detection. It is depicted in Fig. 3 and presented as a set of modular components connected by web service technology. This allows to maintain a clear separation of concerns between police department systems and the framework operation.

1) Data Sources: as input, the proposed framework receives not only crime endogenous data sources (data directly associated with crime) such as crime incidence levels, but exogenous sources, e.g. demographic [7], urban [24] and social media [11]. In fact, exogenous sources (data not directly related to crime) can be quite heterogeneous in their composition. For instance, demographics may be available to specific regions of the city or for the entire city, i.e. not indexed spatially. Similarly, it could be temporally indexed or not, for example districts' area and other spatial features [24]. Succinctly, it's possible to classify data sources concerning to these three main aspects.

- Nature: endogenous or exogenous.
- Spatial variability: variant or constant.
- Temporal variability: variant or constant.

2) Grid: regard the function of processing the city's spatial discrete distribution and aggregating data sources into *Time Series* and *Historical Profile*. It has to instantiate the spatial collection of objects (spots) following a discretization strategy that could be artificial (requiring a free parameter to adapt the size), such as rectangular grid or KGrid, or demographic as police's districts or boroughs.

3) Time Series: are one of the outputs of *Grid*, composing a series of temporal dependent variables, such as crime incidence level, for each spot. They should have a sample frequency of aggregation (time granularity level as daily, weekly, etc) and support filtering by a time window or an specific category (e.g. crime category for endogenous variables). In fact, exogenous sources that are temporally variable would introduce additional relevance when included to the predictions.

4) Historical Profile: is the other output of the *Grid*. It aggregates not only the other temporally constant sources, but data that provide extra detail would collaborate to decision-making, composing a profile for the spots. For instance, top 3 risky streets or 3 unsafe times of the day for each spot that historically aggregate higher criminal incidence should enrich the knowledge provided by the prediction.

5) Ingestion: traduces the domain problem (time series prediction) to a generic machine learning one. It is responsible to start the predictions by extracting and transforming features. As discussed in Section III, autoregressives, seasonals and trends are alternatives to extract features from *Time Series* and for this a important parameter is the number of lags (quantity of previous steps) considered. Empirical studies [12], [13] have shown that time series transformations, such as log or normalization, or even feature selection does impact on performance and should be made in this component and every feature engineering operations should be encapsulated here.

6) Estimators: are the wrappers for the machine learning algorithms. As every algorithm has its own parameters to be tuned for each instance of *Time Series* in order to improve performance, a tuning strategy must consider the computational budget available to train instances. Exhaustive searches in the parameter space sometimes are not scalable when the number of series becomes too large, and randomized search would be more flexible testing less combinations. Also, Cross-validation is recommended to reduce overfitting when tuning them.

7) Portfolio: stores a limited set of tuned machine learning *Estimators*. This component allows less algorithms to be tuned, considering the hypothesis that if an algorithm has a good performance in a problem, it may be also useful in another problem. Although crime behaves differently among regions, there is a trade-off between slightly variations on performance and computational budget to train it all. However, in case training all algorithms for all instances is practicable, a complete *Portfolio* would be fitted. Besides, there is a clear necessity to retrain the algorithms past some time, mainly because more data will be available and crime would change its behavior, even considering the fact that hotspot patrolling will directly disturb it. This retraining concern should be attached by the *Portfolio*, requiring to tune *Estimators* again.

8) Algorithm Selector: complies the function of selecting from *Portfolio* a performatic algorithm for a given instance, increasing the overall prediction performance by mapping a tuned algorithm to a problem instance. If all instances have all algorithms tuned for them (complete *Portfolio*), it's preferable to directly choose the best one. If it's impracticable, one should extract features from the instances and train a classifier to predict what is the best algorithm for it. Kotthoff [25] reviewed many strategies to do so.

9) Prediction Handler: works as the web service controller, requesting for the prediction given all the *Time Series* available using the map predefined by the *Algorithm Selector*. Also, it is responsible to update predictions after the time step referred to the prediction, keeping answering which spots will be classified as hot.

10) Program Card: organizes the output of the service by indexing detected hotspots with the *Historical Profile* of them. It operates as the view of the web service and should be structured in a GeoJSON format to handle not only the hotspots properties, but its geometry.

V. Evaluation

A. Design and Implementation

We have implemented the proposed framework with the Natal's crime dataset, provided by a partnership between SmartMetropolis research project and Public Safety Secretariat of the State of Rio Grande do Norte in Brazil. It aggregates a large set of criminal occurrences registered by police from 2006 to 2016, briefly represented by tuples of occurrence's ⟨*category, latitude, longitude, timestamp*⟩ which correspond to the endogenous source of data. The dataset used in the experiments contains sensitive information, and therefore only its metadata can be discussed. Also, we are using exogenous sources of city's features, such as streets geometries, schools and public squares points, all spatially variant and temporally constant. The data sources are the starting point to design the rest of the implementation and they were wrangled with the help of pandas python library [26]. In order to summarize all the parameters considered for each component, we refer to Table II.

TABLE II
PARAMETERS CONFIGURATION OF THE FRAMEWORK COMPONENTS

Component	**Aspect**	
Data Source	Endogenous	Crime records
	Exogenous	Streets, schools, public squares
Grid	Strategy	KGrid
	Size(s)	10, 30, 50, 70, 90, 110, 130
Time Series	Frequency	Week
	Filters	Time: 2013-01∼2016-06 Categories: all
Historical Profile	Days of the week	3
	Hour intervals	3
	Dangerous streets	4
	# Schools	-
	# Public squares	-
Ingestion	Autoregressives	5 lags
	Trends	5 lags
Estimators	Algorithms	Random Forest, MLP, KNN
	Task	Classification
	Tuning	Grid search (5-Fold CV)
Portfolio	Size	Complete *Portfolio*
	Retrain period	4 months
Algorithm Selector	Feature instances	None (complete *Portfolio*)

Concerning the *Grid*, we choose to fit it artificially with KGrid in order to evaluate prediction performance for smaller sizes of hotspot. The KGrid were built by clustering crime points (*latitude* and *longitude* properties), forming a collection of convex hull polygons as discussed previously [12]. To evaluate its variations on *Estimators* performance, we tested K with 10, 30, 50, 90, 110 and 130 cells (or spots). After *Grid* creation, *Time Series* were fitted to it using the endogenous features available, weekly aggregated and filtered for a time window of three years and a half of records. Using the exogenous sources of data mixed with the endogenous, the *Historical Profile* were built representing the top 3 threatening days of the week and hour intervals that helps to narrow patrol schedules temporally. In terms of spatial binds of the hotspots, it is calculated the most dangerous streets through spatial join operations between streets geometries and crime points. These risky streets were also considered to be indexed as a geometric property of the output of the service.

Besides mandatory autoregressive features in time series context, the *Ingestion* implementation also extracts trends considered for a lag window of 5 past observation. We used the statsmodels python library [27] to extract both seasonal and trend components. The trend was extracted by considering a moving average with the window corresponding to the period of the time series, in weekly time steps, it begins to

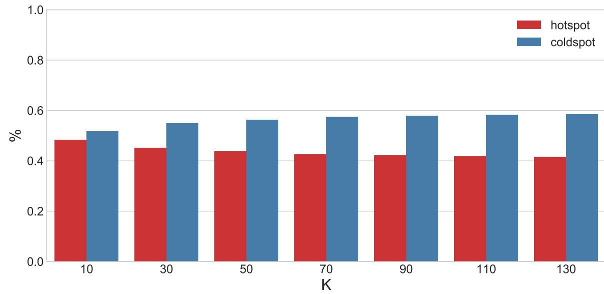

Fig. 4. Classes distribution along different KGrid of y_b.

repeat it self past a year, or 52 points, producing a smoothed version of the original time series. Considering 5 lags for autoregressives and 5 for trends, all features were normalized to avoid further problems in the estimation algorithms. The *Portfolio* of *Estimators* is composed by three implementation of scikit-learn [28] broadly used machine learning algorithms: (i) Random Forest [29] (ii) MultiLayer Perceptron (MLP) [30] and (iii) K Nearest Neighbors (KNN) [31] tuned specifically for each spot with 5-Fold Cross validation. As there are only three tuned estimators for each spot, we preferred to compose a complete *Portfolio* and let *Algorithm Selector* selects the best model considering its CV accuracy score (true positives plus true negatives rates). As discussed in Section III, regression and classification tasks are attainable for these series. However, we consider the *Estimators* objective to classify a *Grid* cell as cold or hotspot adapting Yu et al. [5] strategy through a binary classification task defined in Equation 1. Time series are composed by quantity of crimes per week and to traduce this to binary classes, we simply consider a hotspot ($y_b = 1$) if the current quantity of crimes is greater than the average of the last four observations, similar to a moving average threshold, else it is a coldspot ($y_b = 0$). Considering this definition, Fig. 4 shows class distribution for different grid configurations on the crime dataset.

$$y_b(t) = \begin{cases} 1 \quad if \quad y(t) > \frac{\sum_1^4 y(t-i)}{4} \\ 0 \quad else \end{cases} \tag{1}$$

After the *Algorithm Selector* component maps one of the *Estimators* from the complete *Portfolio* to an instance of *Grid* cell, the *Prediction Handler* is instantiated. It is the controller interface of the service to capture which are the ones considered hotspot for next time interval and passing their identifier keys to the *Program Card* to finally request *Historical Profile* for each region associated with such key. These two last components update the request every week and make it available to be pulled by an application, for instance, the ROTA-Analytics dashboard presented in a previous work [10].

B. Estimators Performance

The previous described implementation is then evaluated experimenting the *Grid* changes in order to find better con-

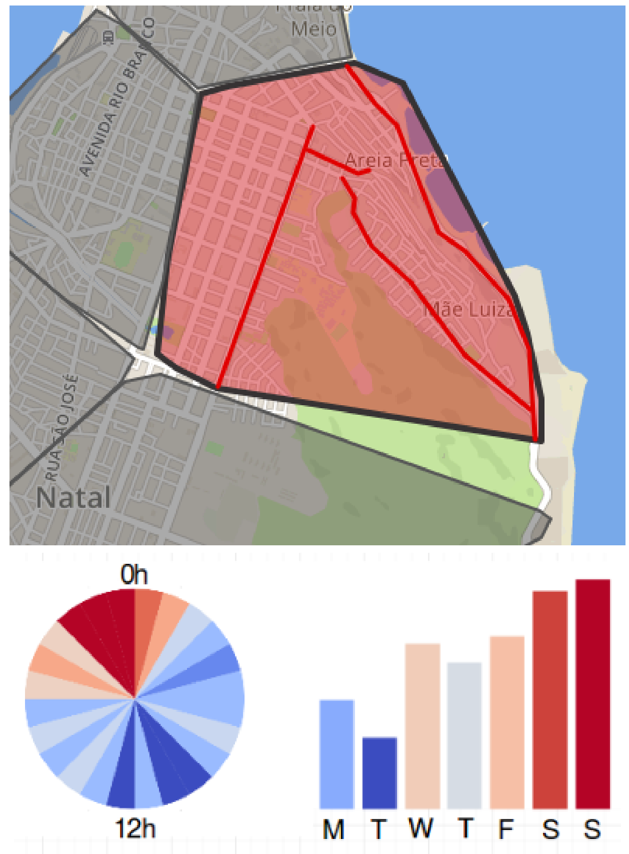

Fig. 5. A snippet of a single hotspot profile in Program-Card.

figurations of hotspots to be patrolled in terms of *Estimators* performance. Seven trials were conducted and Fig. 6 illustrates CV accuracy distribution on the hotspots detection task designed before. Note that accuracy is growing along with grid size, but it stabilizes. This is specially good for the police departments, because with more cells, the prediction will have more spatial precision and incisiveness for patrolling. A reason for the improvement would be explained by the micro-level variations in crime that this neighbors preserves. In a scenario of less and bigger hotspots, the overlap between different types of crime may impact noisily to the prediction. Ariel and Partridge have argued that unit of analysis should be as small as possible, considering what they call "*law of concentration of crime in place and time*": half of calls for service to police occurs in less than 5% of places and also are concentrated in certain times of the day, days of week and months of the year [4]. However, the overall performance starts to decrease in K=130, possible due to sparseness of data discussed in Section III.

In terms of algorithm performance, it is possible to infer that MLP and Random Forest are competing in overall performance. However, the eligible ones preferred by *Algorithm Selector* are illustrated in Fig. 7. It is clear that for all configurations on Natal's dataset the MLP classifier were preferred, but Random Forest does has a share of instances

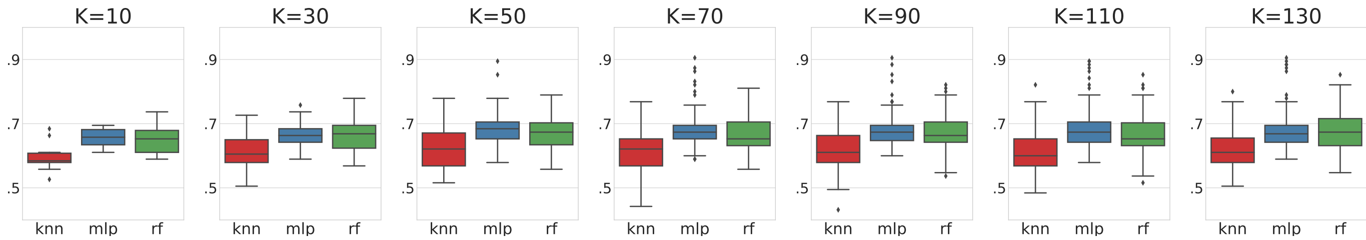

Fig. 6. 5-Fold CV accuracy performance for Random Forest (RF), MLP and KNN classifiers in seven configurations of KGrid.

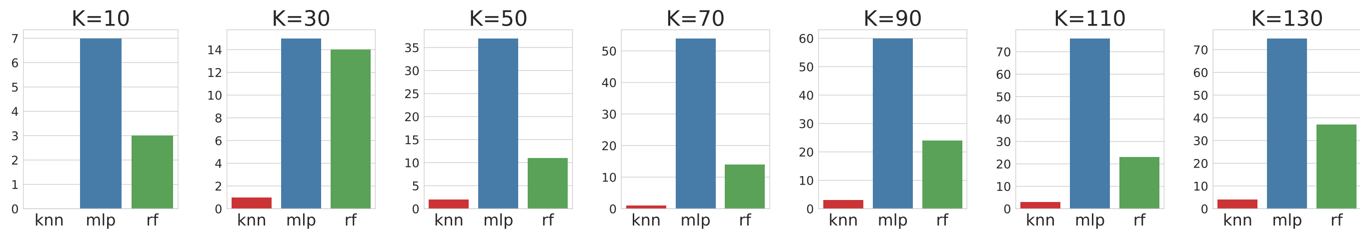

Fig. 7. Algorithm Selector histograms of estimators preferences in seven configurations of KGrid.

to be the predictor, mainly for K=30, followed by KNN with lower participation. Considering the *Algorithm Selector* effects, Fig. 8 illustrates a visual intuition perspective that the selection effects median was improved approximately 10%. This emphasizes that algorithm selection is crucial to achieve improvements in performance and should be indexed in the predictive policing framework. For the best configuration (K=70), the median performance reaches almost 80% of accuracy and for some instances reaches 90%.

VI. Conclusion

In this work we formulate a novel framework towards crime prevention and helps smart cities to support automated patrol planning. To ensure practical use, we have adopted a web service architecture as a way to integrate with public safety systems and platforms, as ROTA one. While it is a conceptual description of the framework, we have implemented an instance as guideline to practitioners. It was possible to describe examples of the component's importance and what kind of parameters should be considered to evaluate the quality of the results. One highlighted parameter in the evaluation is spatial granularity. Our implementation shows that for fine grained grid, i.e. higher K, machine learning models are more suitable. As a consequence, police departments would use them as a precise tool to distribute patrolling in this micro region with the help of descriptive statistics as a profile with its hourly and daily patterns. Additionally, other advantage noted by our framework is that hotspots are generated for each new time window step (weekly in our implementation) and new areas prone to crime are detected over time, considering short-term crime behavior, avoiding vicious patrol in certain places. The framework is attached to ROTA architecture in order to improve Natal, a promising smart city initiative.

VII. Acknowledgement

This work is supported by the SmartMetropolis Project[2]. Nelio Cacho is supported in part by CAPES - Brazil (88881.119424/2016-01).

References

[1] A. Caragliu, C. Del Bo, and P. Nijkamp, "Smart cities in europe," *Journal of urban technology*, vol. 18, no. 2, pp. 65–82, 2011.

[2] A. A. Braga, A. V. Papachristos, and D. M. Hureau, "The effects of hot spots policing on crime: An updated systematic review and meta-analysis," *Justice quarterly*, vol. 31, no. 4, pp. 633–663, 2014.

[3] B. L. Benson, D. W. Rasmussen, and I. Kim, "Deterrence and public policy: Trade-offs in the allocation of police resources," *International Review of Law and Economics*, vol. 18, no. 1, pp. 77–100, 1998.

[4] B. Ariel and H. Partridge, "Predictable policing: Measuring the crime control benefits of hotspots policing at bus stops," *Journal of Quantitative Criminology*, vol. 33, no. 4, pp. 809–833, 2017.

[5] C.-H. Yu, M. W. Ward, M. Morabito, and W. Ding, "Crime forecasting using data mining techniques," in *11th International Conference on Data Mining Workshops (ICDMW), 2011*. IEEE, 2011, pp. 779–786.

[6] W. L. Perry, B. McInnis, C. C. Price, S. C. Smith, and J. S. Hollywood, "Predictive policing: The role of crime forecasting in law enforcement operations," 2013.

[7] A. Bogomolov, B. Lepri, J. Staiano, N. Oliver, F. Pianesi, and A. Pentland, "Once upon a crime: towards crime prediction from demographics and mobile data," in *Proceedings of the 16th international conference on multimodal interaction*. ACM, 2014, pp. 427–434.

[8] J. Coelho, N. Cacho, F. Lopes, E. Loiola, T. Tayrony, T. Andrade, M. Mendonça, M. Oliveira, D. Estaregue, and B. Moura, "Rota: A smart city platform to improve public safety," in *New Advances in Information Systems and Technologies*. Springer, 2016, pp. 787–796.

[9] M. Mendonça, B. Moreira, J. Coelho, N. Cacho, F. Lopes, E. Cavalcante, A. Araujo Jr, J. L. Ribeiro, E. Loiola, D. Estaregue *et al.*, "Improving public safety at fingertips: A smart city experience," in *International Smart Cities Conference (ISC2), 2016*. IEEE, 2016.

[10] A. Araujo Jr, N. Cacho, A. C. Thome, A. Medeiros, and J. Borges, "A predictive policing application to support patrol planning in smart cities," in *International Smart Cities Conference (ISC2), 2017*. IEEE, 2017.

[11] M. S. Gerber, "Predicting crime using twitter and kernel density estimation," *Decision Support Systems*, vol. 61, pp. 115–125, 2014.

[2]http://smartmetropolis.imd.ufrn.br

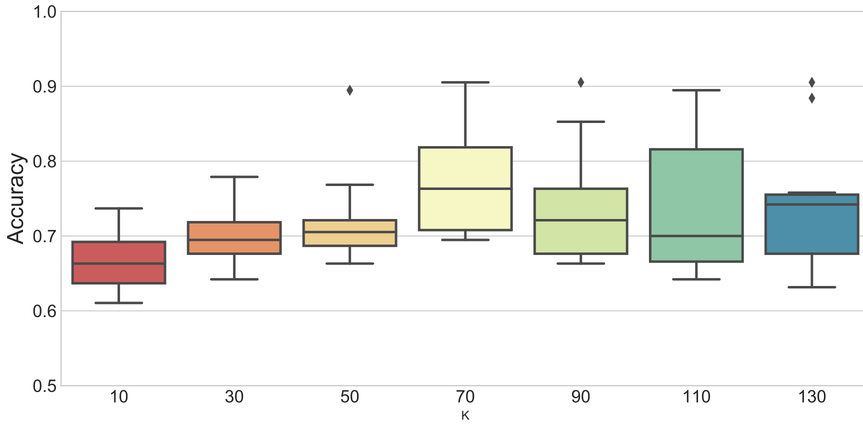

Fig. 8. 5-Fold CV accuracy for the estimators provided by *Algorithm Selector*

[12] D. Ziehr, "Leveraging Spatio-Temporal Features for Improving Predictive Policing," Master's thesis, Karlsruhe Intitute of Technology, Germany, 2017.

[13] A. Malik, R. Maciejewski, S. Towers, S. McCullough, and D. S. Ebert, "Proactive spatiotemporal resource allocation and predictive visual analytics for community policing and law enforcement," *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 1863–1872, 2014.

[14] S. V. Nath, "Crime pattern detection using data mining," in *2006 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology Workshops*, Dec 2006, pp. 41–44.

[15] G. O. Mohler, M. B. Short, P. J. Brantingham, F. P. Schoenberg, and G. E. Tita, "Self-exciting point process modeling of crime," *Journal of the American Statistical Association*, vol. 106, no. 493, pp. 100–108, 2011.

[16] L. W. Kennedy, J. M. Caplan, and E. Piza, "Risk clusters, hotspots, and spatial intelligence: risk terrain modeling as an algorithm for police resource allocation strategies," *Journal of Quantitative Criminology*, vol. 27, no. 3, pp. 339–362, 2011.

[17] N. Cacho, F. Lopes, E. Cavalcante, and I. Santos, "A smart city initiative: The case of natal," in *International Smart Cities Conference (ISC2), 2016*. IEEE, 2016, pp. 1–7.

[18] A. Cacho, M. Figueredo, A. Cassio, M. V. Araujo, L. Mendes, J. Lucas, H. Farias, J. Coelho, N. Cacho, and C. Prolo, "Social smart destination: a platform to analyze user generated content in smart tourism destinations," in *New Advances in Information Systems and Technologies*. Springer, 2016, pp. 817–826.

[19] A. Souza, J. Pereira, J. Oliveira, C. Trindade, E. Cavalcante, N. Cacho, T. Batista, and F. Lopes, "A data integration approach for smart cities: The case of natal," in *International Smart Cities Conference (ISC2), 2017*. IEEE, 2017.

[20] H. Chourabi, T. Nam, S. Walker, J. R. Gil-Garcia, S. Mellouli, K. Nahon, T. A. Pardo, and H. J. Scholl, "Understanding smart cities: An integrative framework," in *45th Hawaii International Conference on System Science (HICSS), 2012*. IEEE, 2012, pp. 2289–2297.

[21] R. Maciejewski, R. Hafen, S. Rudolph, S. G. Larew, M. A. Mitchell, W. S. Cleveland, and D. S. Ebert, "Forecasting hotspots—a predictive analytics approach," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 4, pp. 440–453, 2011.

[22] M. Längkvist, L. Karlsson, and A. Loutfi, "A review of unsupervised feature learning and deep learning for time-series modeling," *Pattern Recognition Letters*, vol. 42, pp. 11–24, 2014.

[23] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[24] J. Borges, D. Ziehr, M. Beigl, N. Cacho, A. Martins, S. Sudrich, S. Abt, P. Frey, T. Knapp, M. Etter, and J. Popp, "Feature engineering for crime hotspot detection," in *Conference on Smart City Innovations (SCI), 2017 International*. IEEE, 2017.

[25] L. Kotthoff, "Algorithm selection for combinatorial search problems: A survey," in *Data Mining and Constraint Programming*. Springer, 2016, pp. 149–190.

[26] W. McKinney, "pandas: a foundational python library for data analysis and statistics."

[27] S. Seabold and J. Perktold, "Statsmodels: Econometric and statistical modeling with python," in *9th Python in Science Conference*, 2010.

[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[29] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[30] S. Haykin, "A comprehensive foundation," *Neural networks*, vol. 2, no. 2004, p. 41, 2004.

[31] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.