

RESEARCH

Open Access



# Multi-density crime predictor: an approach to forecast criminal activities in multi-density crime hotspots

Eugenio Cesario<sup>1\*†</sup>, Paolo Lindia<sup>1†</sup> and Andrea Vinci<sup>2†</sup>

<sup>†</sup>Eugenio Cesario, Paolo Lindia and Andrea Vinci have contributed equally to this work.

\*Correspondence:  
eugenio.cesario@unicl.it

<sup>1</sup> University of Calabria, Via P. Bucci 18B, 87036 Rende, Italy  
<sup>2</sup> Institute for High Performance Computing and Networking - National Research Council of Italy (ICAR-CNR), Via P. Bucci, 8/9c, 87036 Rende, Italy

## Abstract

The increasing pervasiveness of ICT technologies and sensor infrastructures is enabling police departments to gather and store increasing volumes of spatio-temporal crime data. This offers the opportunity to apply data analytics methodologies to extract useful crime predictive models, which can effectively detect spatial and temporal patterns of crime events, and can support police departments in implementing more effective strategies for crime prevention. The detection of crime hotspots from geo-referenced data is a crucial aspect of discovering effective predictive models and implementing efficient crime prevention decisions. In particular, since metropolitan cities are heavily characterized by variable spatial densities of crime events, multi-density clustering seems to be more effective than classic techniques for discovering crime hotspots. This paper presents the design and implementation of MD-CrimePredictor (Multi-Density Crime Predictor), an approach based on multi-density crime hotspots and regressive models to automatically detect high-risk crime areas in urban environments, and to reliably forecast crime trends in each area. The algorithm result is a spatio-temporal crime forecasting model, composed of a set of multi-density crime hotspots, their densities and a set of associated crime predictors, each one representing a predictive model to forecast the number of crimes that are estimated to happen in its specific hotspot. The experimental evaluation of the proposed approach has been performed by analyzing a large area of Chicago, involving more than two million crime events (over a period of 19 years). This evaluation shows that the proposed approach, based on multi-density clustering and regressive models, achieves good accuracy in spatial and temporal crime forecasting over rolling prediction horizons. It also presents a comparative analysis between SARIMA and LSTM models, showing higher accuracy of the first method with respect to the second one.

**Keywords:** Crime data mining, Crime forecasting, Crime hotspots, LSTM, Multi-density clustering, Urban crime data analysis, Smart cities

## Introduction

### Reference context

The increasing urbanization occurring during the last years is transforming every aspect of the urban society and affecting its sustainable development [1–4]. In fact, as

urbanization continues to grow, it is bringing significant social and economic benefits (i.e., additional urban services and employment opportunities), while also presenting challenges in city management issues, like resource planning (water, electricity), traffic, air and water quality, public policy and public safety services.

Among the main urban issues, criminal activities are one of the most important social problems in metropolitan areas, because they can severely affect public safety, harm the economy and sustainable development of a society, as well as reduce the quality of life and well-being of citizens. For such a reason, improving strategies to effectively manage and utilize limited public security resources has become a crucial issue for policymakers and urban management departments.

However, ICT technologies and sensor infrastructures are enabling public organizations and police departments to gather and store increasing volumes of crime-related data, with spatial and temporal information. This offers the opportunity to apply data analytics methodologies to extract useful knowledge models, which can effectively detect spatial and temporal patterns of crime events. By extracting useful predictive models and applying appropriate methods for data analysis, police departments are supported to better utilize their limited resources and implement more effective strategies for crime prevention.

### **Motivations and contributions**

Several criminal justice studies show that the incidence of criminal events is not uniformly distributed within a city [2, 3, 5, 6]. In fact, crime trends are strongly affected by the geographic location of the area (there are low-risk and high-risk areas). Also, they can vary with respect to the period of the year (there could be seasonal patterns, peaks, and dips). For this reason, an effective predictive model must be able to automatically determine which city neighborhoods are most affected by crime-related incidents, namely *crime hotspots*, as well as how the crime rate in each particular hotspot evolves over time. This knowledge can allow police departments to allocate their resources more efficiently over the urban territory, enabling the effective deployment of officers to high-risk areas, or moving officers from areas expecting a decline in crime activities, thus more efficiently preventing or promptly responding to crimes.

In literature, classic density-based clustering algorithms are largely exploited to discover spatial hotspots [7–11]. However, due to the adoption of global parameters, they fail to identify *multi-density hotspots* (i.e., different regions having various densities [12, 13]) unless the clusters (or hotspots) are clearly separated by sparse regions [14]. In particular, this is a key issue when analyzing crime data and thus correctly detecting the real crime hotspots. In fact, the density of population, traffic, or events in large cities can vary widely from one area to another area [5], which also makes the incidence of crime events extremely dissimilar in terms of density.

Such a spatial density variation in crime events challenges the discovery of proper hotspots when classic density-based algorithms perform the analysis. For example, the well-known DBSCAN [14] receives two global input parameters ( $\epsilon$  and *min - points*), which results in a minimum density threshold  $\delta_{min}$  that is exploited for clustering the whole dataset. The optimal value of  $\delta_{min}$  can affect the densities of the discovered hotspots and does not deal with large density variations in the urban data. Indeed, if the value of  $\delta_{min}$

is too small, the algorithm can discover several small non-significant hotspots that actually do not represent dense crime regions, while if  $\delta_{min}$  is too large, it can discover a few large regions having high intra-cluster density variations. Thus, classic density-based clustering algorithms fail to identify proper hotspots characterized by different density levels, and their application to discover crime hotspots can produce inaccurate results, particularly in urban environments. A recent study in Cesario et al. [5] shows that multi-density clustering achieves higher performance than classic approaches for discovering hotspots in multi-density urban environments.

This paper presents the design and implementation of MD-CrimePredictor (Multi-Density Crime Predictor), an approach based on multi-density crime hotspots and regressive models to automatically detect high-risk crime areas in urban environments, and to forecast crime trends in each area reliably. The algorithm is composed of three main steps. First, multi-density crime hotspots are detected by applying a multi-density clustering algorithm (i.e., CHD) proposed in Cesario et al. [5], where densities, shapes, and number of the detected regions are automatically computed by the algorithm without any pre-fixed division in areas. Then, a specific regressive model is discovered from each detected hotspot, analyzing the partitions discovered during the previous step. In this paper, this is done by exploiting both SARIMA [15] and LSTM [16] models, and a comparative experimental analysis is presented in terms of error measures. The final result of the algorithm is a spatio-temporal crime forecasting model, composed of a set of crime hotspots, their densities, and a set of associated crime predictors, each one representing a predictive model to forecast the number of crimes that are estimated to happen in its specific hotspot. The experimental evaluation of the proposed approach has been performed by analyzing a large area of Chicago, involving more than two million crime events (over a period of 19 years). The experimental evaluation, aimed at assessing the effectiveness of the approach over rolling prediction horizons, presents a comparative analysis between SARIMA and LSTM regression models, demonstrating higher accuracy of the first method with respect to the second one. We also provide a comparative assessment of the proposed approach with other studies proposed in literature, drawing a comparison in terms of hotspots detection and crime forecasting accuracy. Overall, the results show the effectiveness of the approach, by achieving good accuracy in spatial and temporal crime forecasting over rolling time horizons.

### Plan of the paper

The rest of the paper is organized as follows. Section "[Related work](#)" reports the most important approaches proposed in the literature for crime hotspot detection and crime forecasting. Section "[Problem Definition and Proposed Approach](#)" outlines the problem statement and describes the approach proposed in the paper and reports its steps in detail. Section "[Experimental Evaluation and Results](#)" provides the experimental evaluation of the proposed approach on a real-world scenario by showing a comparative analysis between SARIMA and LSTM performances. The section also shows a comparison between the results achieved with the presented approach and other methodologies proposed in the literature. Finally, Sect. "[Conclusion](#)" concludes the paper and plans future research works.

## Related work

Recently, crime hotspot detection and crime forecasting have been raised as hot topics within the research community. This section briefly reviews the most representative research works in both areas.

### Crime forecasting

One of the first frameworks proposed in the literature for crime data analysis is CrimeTracer [17], which is based on a probabilistic approach to model the spatial behavior of known offenders within areas they frequent, called *activity spaces*. This work is based on the assumption, based on crime pattern theories, that offenders frequently commit serial violent crimes in places they are most familiar with (namely, their activity space). Also, the authors claim that taxi flows can provide useful information to correlate activity spaces, even if they are not geographically connected. Experiments carried out on real-world crime data have shown that criminals frequently commit crimes within their activity spaces, rather than venture into unknown territories. CrimeTracer is indeed able to predict the location of the next crime committed by known offenders but it does not provide information about the time window for the next crime events. Also, it requires a dataset with information related to specific offenders, which could not be available in general.

The work in Catlett et al. [7] presented a predictive approach based on spatial analysis and auto-regressive models in order to detect high-risk regions in urban areas and to forecast crime trends in each region. The approach exploits the DBSCAN algorithm to detect high-risk regions and ARIMA models to fit crime predictors. The approach has been validated on two crime datasets (i.e., Chicago and New York City areas) comprising crime events spanning from 2001 to 2016. The study shows good performances on both datasets, considering a three-year ahead forecasting window, which is a long-term time horizon. The approach is capable of detecting crime-dense regions having any shapes, however the main drawback is that DBSCAN detects wide regions or a large number of outliers, as it cannot tackle the multi-density nature of urban datasets.

The study described in Zhu et al. [3] proposes a hierarchical crime prediction framework, which integrates a modified gated GCN (Graph Convolutional Networks) and VMD (variational mode decomposition), to holistically predict the short-term crime patterns in different communities and support proactive policing. The approach is composed of several steps. First, the temporal dependency is decomposed in the frequency domain, and a network is constructed to capture the spatial relationships within the sub-frequencies. Then, human mobility traces are exploited to characterize the dynamic relationships within the network. The experimental evaluation has been focused on the crime distribution evolution of crimes in Chicago, to predict the short-term criminal events in the different communities holistically. The study concludes that social interactions based on human activity data can characterize dynamic crime distribution relationships, as well as spatial crime distribution evolutions. The main strength of the research study proposed in Zhu et al. [3] leverages on the dynamic relationships between human mobility and crimes, which represents a relevant methodological difference with other approaches proposed in literature; in particular, the analysis of human mobility

allows to characterize also the dynamic distribution and evolution of crimes within and across areas, which is strongly affected by social interactions among individuals. However, while the approach exhibits reasonable effectiveness of taking a relationship-based perspective for crime forecasting, the theoretical description needs further verification (as also claimed by authors): in fact, as human activity data is multi-source, multi-granular, and multi-mode, and involves complex relationships, a more refined classification of human mobility trends is needed to understand their effects on different crime evolutions.

A general framework for crime data mining, exploited for some analysis tasks in collaboration with the Tucson and Phoenix Police departments, is presented in Chen et al. [18]. In particular, the paper describes three examples of its use in practice. First, entity extraction algorithms have been used to automatically identify persons, addresses, vehicles, and personal characteristics from police narrative reports (usually containing many typos, spelling errors, grammatical mistakes, etc.). Second, a text mining algorithm has been explored for deceptive identity detection, to discover the real identity of suspects that have given false names, faked birth dates, or false addresses. Third, a concept-based approach has been exploited to identify subgroups or key members in criminal networks, and to study interaction patterns among them. In our opinion, the main strength of this study is its innovativeness in providing investigators with a framework for automatically applying crime entity-extraction techniques on crime data, aiming to extract serial offenders' behavioral patterns. However, using only crime department data could limit the applicability and effectiveness of the framework; as also observed in Chen et al. [18], additional heterogeneous data (i.e., citizenship, secret services, immigration, web, social) could enable the development of more intuitive techniques for crime pattern and network visualization, and higher accuracy in criminal activity predictions.

Authors of Liang et al. [19] propose a framework, named CrimeTensor, to predict the number of crime incidents belonging to different categories within each target region. The framework, based on tensor learning with spatio-temporal consistency techniques, aims to offer fine-scale prediction results considering spatio-temporal categorical correlations in crime events. Crime data is modeled as a tensor, and an objective function is presented, which leverages spatial, temporal, and categorical information. The prediction task is done by applying CANDECOMP/PARAFAC decomposition to find an optimal solution for the defined objective function. The approach is validated by conducting experiments on two real-world crime datasets, collected in the Xiaogan (China) and New York City (USA), each one collecting one year of data. The approach can forecast crimes while distinguishing between different crime types, but it considers only a pre-defined set of regions. Furthermore, the experimental evaluation has been performed only on four months of data. Also, the resulting model requires several different information (i.e., crimes, regions, demographics, road networks) to be trained.

The work in Zhu et al. [2] presents an approach based on K-means clustering, signal decomposition techniques, and neural networks to identify crime distribution in urban areas and forecast crime trends in each area. The approach has been evaluated on a Chicago real-world dataset (collecting crime data from 2011 to 2018). As a main novelty of the approach, the authors exploited Bidirection Recurrent Neural Networks for the forecasting task. The results show good accuracy regarding one-day-ahead prediction

in terms of MAPE. The main strength of this study, as also reported by authors, consists in its experimental results showing that the crime time series in different areas exhibit a correlation in the long term, but this long-term effect cannot be reflected in the short period. This contradiction affects a different perception of public safety quality between police departments and individuals. On the other side, three main issues could be overtaken: (i) the application of k-means for cluster detection tends to detect globular-shaped crime hotspots, which could be not completely appropriate in dynamic environments like metropolitan cities; (ii) the number of clusters (six) detected in the whole City of Chicago can lead to have very large clusters, someone even larger than the pre-defined administrative police districts of the city; (iii) the crime types and social impacts of the crime are not considered in the approach, and could add an important value to the whole process.

The work in Wang et al. [20] studies crime inference between neighbor areas by exploiting crime data, POIs, and taxi flows analyzed by Linear Regression and Negative Binomial Regression models. The authors evaluated the approach on the Chicago crime data for five years (2010 and 2015) and considered the city's administrative boundaries to partition data. A wide set of experiments was performed to compare the results gathered with different feature combinations. Even if the approach was proven to be effective in crime inference, the findings show that, on the tested data, the taxi flow distribution is highly skewed, and this causes a significant forecasting error in some areas.

In Han et al. [21], the authors proposed an approach for predicting daily crimes by leveraging a combination of Long Short-Term Memory Network (LSTM) and Spatial-Temporal Graph Convolutional Network (ST-GCN). The algorithm involves topological maps, crime transitions detected by ST-GCN, and temporal trends extracted by LSTM. Finally, a Gradient Boost Decision Tree (GBDT) integrates the predicted values from both modules to create a spatial-temporal model for crime prediction. The experimental evaluation has been assessed on Chicago crime data. It provides an analysis of 0.32 million crimes over six years, considering only the communities with a large number of crime cases.

The approach presented in Li et al. [22], named ST-HSL, proposes a Spatial-Temporal Hypergraph Self-Supervised Learning framework. The approach focuses on the analysis of sparse crime data, with the aim of tackling the label scarcity issue in crime prediction. Specifically, the authors propose a method to perform spatial-temporal prediction via Graph Neural Networks, based on a cross-region hypergraph structure learning to encode region-wise crime dependency within the entire urban space. Additionally, a dual-stage self-supervised learning approach is designed, with the two goals of (i) capturing spatial-temporal crime patterns at both local and global levels, and (ii) enhancing the representation of sparse crime data by improving region-specific discrimination. The experimental evaluation has been carried out by integrating geographic grid-based regions and crime data on two real-world case studies, i.e., Chicago and New York City, by also performing a comparative analysis with several state-of-the-art baselines. In our opinion, the main strength of this approach consists in its capability of performing spatial-temporal representation with sparse crime data, and the ability of neural network-based models to differentiate spatial-temporal category crime patterns of different regions and time periods under data scarcity. On the other side, the predefined hotspot

boundaries in grid-cells could limit the effectiveness of the approach to detect spatial dynamic distributions of crimes in the area under investigation.

The study presented in Zhou et al. [23] takes inspiration from the fact that, due to municipal regulations and maintenance costs, it is not trivial for many cities to collect high-quality labeled crime data, whose availability is crucial for a further data analysis process. In such cases, authors propose to develop a crime prediction model for a target city without labeled crime data by learning knowledge from a source city with abundant data; the basic idea is to use common context data to train a model from the source city and then fine-tune this model to solve tasks in the target city. However, the authors highlight that the inconsistency of relevant context data between cities exacerbates the difficulty of this prediction task. To deal with this issue, the paper [23] proposes an unsupervised domain adaptation model (UDAC) for crime risk prediction across cities while addressing data scarcity and inconsistency issues. More specifically, the approach is composed of three main steps. First, given a target city affected by a scarcity of labeled crime data, several similar source city grids for each target city grid are identified. Then, based on these source city grids, auxiliary contexts for the target city are built, to make contexts consistent between the two cities. Finally, a dense convolutional network with unsupervised domain adaptation is designed to learn high-level representations for accurate crime risk prediction and simultaneously learn domain-invariant features for domain adaptation. The approach has been evaluated through experiments performed on three real-world datasets from New York City, Chicago, and Los Angeles. In our opinion, the topic investigated by this paper is very interesting, as data scarcity is a major challenge when training machine- and deep-learning models. However, as also noted by the authors, this technique could be applied to other fine-grained unsupervised crime risk prediction, such as predicting crime risk in roads, where the data sparsity problem is very high [23]. Also, the identification of equal-sized grids in the target and source cities could statically partition the territory, not considering the evolution of crimes during the time.

A comparison between several crime prediction and forecasting approaches is provided in Safat et al. [24]. The paper compares different machine learning algorithms, i.e., logistic regression, support vector machines, naive-bayes, k-nearest neighbors, decision trees, autoregressive integrated moving average models, and long-short term memory neural networks. The evaluation has been based on crime data gathered in Chicago (2004–2020) and Los Angeles (1990–2020) cities. The experimental evaluation provides forecasting results over a five-year window, considering the whole city and not specific areas within the city.

A systematic review of several research works about crime hotspot detection and crime prediction is presented in Butt et al. [1]. In particular, the paper analyzes the impact of clustering techniques on the discovery of crime hotspots, and how time series analysis and deep learning techniques can be exploited for crime trend prediction. The review shows that ARIMA and LSTM models are the most used techniques for predicting crime trends in urban environments. The review also highlights the need, for comparison purposes, to exploit publicly available data to assess crime prediction results, and that the most widely exploited measurements for evaluating the effectiveness of the different approaches are MAE, MAPE, and RMSE, and suggest the use of relative

performance indexes, such as MAPE, to simplify the comparison between different approaches.

Table 1 reports a more detailed and critical comparison of the solutions proposed in the literature, including also our proposed approach MD-CrimePredictor. The comparison takes into account several features, as detailed below.

*Goal of the approach.* This feature describes the topic under investigation and the goal of the proposal. MD-CrimePredictor and the approaches presented in refs. [2, 3, 7, 17, 21–23] are aimed at detecting crime hotspots (or crime locations) and crime forecasting models, while the approach proposed in Chen et al. [18] is more focused on deceptive identity detection and criminal-network analysis.

*Data.* This feature is related to the data the approaches have been tested on. All approaches have been evaluated on real-world crime datasets (mainly from Chicago, Vancouver, New York City, and Phoenix), in some cases integrated with human mobility data [3, 17, 20] and other contextual data [19, 20].

*Methods.* This feature differentiates the algorithms on the basis of the methodologies used for the faced crime analysis task. The approaches presented in refs. [2, 7] and MD-CrimePredictor exploit density-based clustering algorithms to detect interesting hotspots, and ARIMA-based and neural networks-based approaches to perform crime forecasting (with some differences among them). Another set of works exploits pre-defined area boundaries and Artificial Neural Networks based methodologies to predict crimes [21–23]. On the other side, the algorithms described in [3, 17, 18, 20] exploit other techniques, ranging from probabilistic approaches to variational mode decompositions, entity-detection, and text-mining approaches.

*Main features.* In addition to the listed comparative categories, we report in Table 1 also a selection of the main features that characterize the revised approaches. The algorithms described in refs. [3, 7] have the good peculiarity of automatically detecting hotspots of any shape (e.g., circular, rectangular, irregular), while the approaches proposed in refs. [2, 7, 17] share the effective capability to perform predictions on rolling forecasting time-horizons. Also, some algorithms differentiate the predicted criminal activities on the basis of crime categories [18, 19, 22], which could be an added value knowledge aimed at supporting police prevention activities. Furthermore, some approaches [2, 17, 19–23] deal with only pre-given or specific crime hotspots (activity spaces, grid-cells, etc.): this may reduce the forecasting effectiveness of such techniques, because they could not detect dynamic changes in spatial criminal evolutions. Moreover, the approach described in Catlett et al. [7] detects multi-shape hotspots, but the results exhibit a significant number of noise points. Finally, the algorithms described in refs. [3, 17, 19, 20] rely on the availability and integration of multiple data (i.e., crimes, metro, taxi, demographic, land use, etc.): from one side the discovery of models correlating urban events and criminal activities is very interesting, from the other side this could be critical in cases where a part of such data are not available for the areas under investigation.

### **Crime hotspot detection**

The systematic review presented in Butt et al. [1] reports that for what concerns hotspot detection techniques, RandomForest and DBSCAN are the most popular approaches exploited. This specific aspect is also analyzed in Cesario et al. [25], which studies how other



**Table 1** Related work comparison table

Publication	Goals	Data	Methods	Features
Tayeb et al. [17]	Crime location prediction	Roads Map and Crimes (2001–2006) in Metro Vancouver	Probabilistic Framework based on Activity Spaces	Able to predict the location of the next crime committed by known offenders. A crime dataset containing information about offenders is required, which could not be available in general. It does not provide a time window for the next crime events.
Catlett et al. [7]	Crime hotspot detection and crime forecasting	Crimes events in Chicago (2001–2016) and New York City (2006–2016)	DBScan for Crime Hotspot detection, ARIMA models for forecasting.	It is able to predict the expected crimes number on a long-term time horizon for the detected crime-dense regions. It can detect crime-dense regions having any shapes. The exploited DBScan algorithm can detect wide crime-dense regions, or a large number of outliers.
Zhu et al. [3]	Crime pattern characterization and prediction	Chicago Crimes events (2004–2019) and taxi mobility (2013–2017)	Graph Convolutional Networks (GCN) and Variational Mode Decomposition (VMD)	Exploiting human mobility traces allows to dynamically model crime distributions and evolutions. Complex human relationships need more refined classification models of human mobility.
Chen et al. [18]	Deceptive identity detection, criminal-network analysis	Tucson and Phoenix Police Department's crime database	Entity extraction, text mining and concept-space approaches	Extracts behavioral patterns of serial offenders. Web and social data are not considered.
Wei et al. [19]	Crimes prediction distinguishing crimes categories	Crime events in Xiaogan and New York City, plus related demographic, POI, Road Network, and Land Use data.	Tensor representation, optimization through CANDECOMP/ PARAFAC decomposition, agglomeration nesting.	Differentiates crime categories, considers features other than crime data. Pre-defined set and shape of regions, custom-defined performance indexes.
Zhu et al. [2]	Crime hotspots detection and forecasting	Crimes events in Chicago (2001–2018)	k-means and neural networks	Correlation detected for long-term time series. Globular-shaped hotspots, crime types not considered.
Wang et al. [20]	Crime rate inference in communities	Crimes events in Chicago (2010–2014), related demographic, geographic, POI, and Taxi flows data	Linear Regression. Negative binomial regression.	Correlates information gathered by different dataset. Exploits a pre-defined set and shape of regions. Estimation error occurs in some regions, due to skewness of taxi flows data.

**Table 1** (continued)

Publication	Goals	Data	Methods	Features
Han et al. [21]	Crimes prediction with feature extraction	Crimes events in Chicago (2014-2020)	Long Short-Term Memory (LSTM) and Spatial-Temporal Graph Convolutional Network (ST-GCN)	Detection of transition tendencies of crimes between neighboring communities over time. Communities with low number of crimes not considered.
Li et al. [22]	Spatial-temporal prediction from sparse crime data	Crimes events in Chicago (2016-2017) and New York City (2014-2015)	Graph Neural Networks	Crime forecasts differentiated by categories. Predefined grid-cell hotspots could not detect dynamic crime distributions
Zhou et al. [23]	Detecting crime prediction models for a city without labeled crime data	POIs, Crimes in Chicago, New York City and Los Angeles (2015)	Dense Convolutional Networks	Unsupervised domain adaptation for crime risk prediction across cities. Partitioning target and source cities in equal-sized grids is a static solution, not considering the evolution of crimes during the time.
MD-Crime Predictor	Spatio-temporal crime forecasting in multi-density hotspots	Crimes events in Chicago (2001-2019)	CHD for Crime Hotspot detection, SARIMA and LSTM models for crime forecasting.	Detection of multi-density and multi-shape crime hotspots, specific crime forecasting model for each hotspot. Correlation with other urban events not considered.

clustering techniques, based on multi-density approaches, outperform classic approaches to discover urban hotspots. More specifically, the paper compares the DBSCAN, OPTICS-xi, HDBSCAN, and CHD algorithms against two artificial and one real datasets, by selecting the best fitting algorithm parameters through a parameter sweeping approach. The results of the experimental evaluation on the artificial datasets, made in Cesario et al. [25], are reported in Tables 2 and 3, where the clustering results are compared by several performance indexes (for each index, the best achieved result is reported in bold). The analysis shows that the HDBSCAN and CHD algorithms are the most effective in detecting clusters in multi-density dataset, and that CHD performs better than HDBSCAN on the second dataset (see Table 3). However, other approaches are presented in the literature, specifically tailored for clustering spatio-temporal data. The work in Nanni et al. [26] presents the TF-OPTICS algorithm, designed for time-focused clustering. The algorithm processes a set of spatio-temporal objects, each one represented by a trajectory of values, as a function of time. TF-OPTICS focuses on computing distances between trajectories by searching for the best possible time interval. This algorithm, as well as those ones tailored for clustering trajectories of moving objects, does not suit to the proposed use case, because we focus on crime events characterized both in time and space, that can not be aggregated in a set of well-defined trajectories. A more fitting algorithm for clustering spatio-temporal data is presented in Agrawal et al. [27]. The algorithm, called ST-OPTICS, is density-based, and exploits two different  $\epsilon$  parameters, one for clustering points in space and the other for clustering points in time. A comparison between the proposed approach, based on CHD, and an alternative one, based on the ST-OPTICS algorithm, is provided in the Sect. "[Comparative analysis with ST-OPTICS on hotspots detection and crime forecasting](#)".

#### **Main differences and novelty of MD-CrimePredictor**

With respect to the summarized works, this paper presents two main novelties. First, it introduces MD-CrimePredictor, where a multi-density clustering algorithm (i.e., CHD) is exploited for crime hotspot detection (to the best of our knowledge, this is the first research study in the crime data analysis domain, showing results on multi-density crime hotspots). The exploited approach CHD is able to automatically detect multi-density (and multi-shape) crime hotspots, which differentiates it w.r.t. all the other approaches reviewed here, thus showing important benefits in the urban data analysis. MD-CrimePredictor relies on the exploitation of both seasonal regressive (SARIMA) and deep-learning (LSTM) models for crime forecasting in each discovered hotspot, and, as a second contribution, the paper furnishes an extensive comparative evaluation between the results given by the two forecasting algorithms. Also, to assess the effectiveness of the CHD-based approach for hotspot detection, we show a comparative analysis of the proposed approach with other studies proposed in literature, drawing a comparison in terms of hotspots detection and crime forecasting accuracy

#### **Problem definition and proposed approach**

This section presents the problem formulation and the approach proposed in the paper to forecast crime events in multi-density crime hotspots. Specifically, Sect. "[Problem definition and goals](#)" depicts the problem under investigation and its goals, whereas Sect. "[The multi-crime-predictor approach](#)" details the algorithm proposed in the paper.

**Table 2** Performance comparison between different density-based clustering algorithm on dataset Zahn Compound [25]

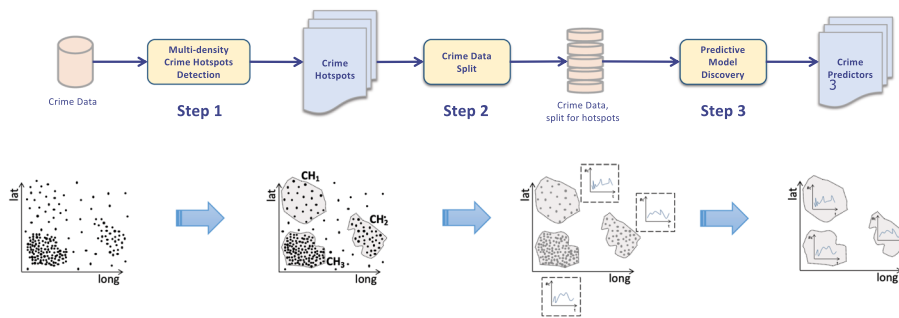
	Fowlkes	Adj. Rand	AMI	V-measure	Accuracy	F1 Score	Jaccard	Gamma	Rand	Homogeneity	Avg
DBSCAN	0.839	0.831	0.801	0.802	0.842	0.839	0.817	0.831	0.850	0.800	0.827
HDBSCAN	0.869	0.807	<b>0.862</b>	<b>0.864</b>	<b>0.904</b>	<b>0.861</b>	0.756	0.822	<b>0.920</b>	0.760	0.848
OPTICS-Xi	0.838	0.825	0.831	0.832	0.791	0.838	0.804	0.826	0.855	0.794	0.822
CHD	<b>0.891</b>	<b>0.873</b>	0.789	0.821	0.866	0.842	<b>0.840</b>	<b>0.873</b>	<b>0.920</b>	<b>0.856</b>	<b>0.862</b>

The best results are reported in bold

**Table 3** Performance comparison between different density-based clustering algorithm on Ordered Chess dataset [25]

	Fowlkes	Adj. Rand	AMI	V-measure	Accuracy	F1 Score	Jaccard	Gamma	Rand	Homogeneity	Avg
DBSCAN	0.508	0.384	0.436	0.453	0.471	0.479	0.343	0.427	0.625	0.386	0.467
HDBSCAN	0.391	0.313	0.346	0.362	0.419	0.379	0.274	0.330	0.506	0.329	0.380
OPTICS-Xi	0.137	0.051	0.114	0.127	0.172	0.120	0.074	0.067	0.184	0.103	0.131
CHD	<b>0.644</b>	<b>0.606</b>	<b>0.594</b>	<b>0.615</b>	<b>0.690</b>	<b>0.690</b>	<b>0.525</b>	<b>0.607</b>	<b>0.768</b>	<b>0.638</b>	<b>0.653</b>

The best results are reported in bold



**Fig. 1** The multi-crime-predictor algorithm workflow

**Problem definition and goals**

We begin by fixing a proper notation to be used throughout the paper. Let  $T = \langle t_1, t_2, \dots, t_H \rangle$  be an ordered timestamp list, such that  $t_h < t_{h+1}, \forall_{0 \leq h < H}$ , and where all  $t_h$  are at equal time intervals (e.g., every hour, day, week). Let  $\mathcal{CD}$  be a crime dataset collecting crime events,  $\mathcal{CD} = \langle CD_1, CD_2, \dots, CD_N \rangle$ , where each  $CD_i$  is a data instance described by  $\langle latitude, longitude, t \rangle$ , i.e., the coordinates of the place and the time (with  $t \in T$ ) the event occurs at. Now, let us consider a future temporal horizon,  $S = \langle t_s, t_{s+1}, \dots \rangle$ , with  $s > H$ . The goal of the analysis is to discover a set of crime hotspots in the city (which can have multi-density distribution of the events) and predictive models for reliably forecasting the number of crimes in each hotspots at a given timestamp  $t_s \in S$ . More specifically, the goal of the proposed approach aims at achieving the following goals:

1. Discover a set  $\mathcal{CH}$  of *crime hotspots*,  $\mathcal{CH} = \{CH_1, \dots, CH_K\}$ , where a *crime hotspot*  $CH_k$  is a spatial area which criminal events occur in with an higher density than other areas in the city;
2. Compute a set  $\Sigma$  of *crime hotspot densities*,  $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_H\}$ , where each  $\sigma_h$  is the spatial density of events occurred in the hotspot  $CH_h$ .
3. Extract a set  $\mathcal{F}_{crimes}$  of *crime predictors*,  $\mathcal{F}_{crimes} = \{\mathcal{F}_{crimes}^1, \dots, \mathcal{F}_{crimes}^K\}$ , where each function  $F_{crime}^k : \mathcal{S} \rightarrow \mathcal{R}$ , given a timestamp  $t_s \in \mathcal{S}$  states the number of crimes  $N \in \mathcal{R}$  that are predicted to happen in the *crime hotspot*  $CH_k \in \mathcal{CH}$  at the timestamp  $t_s$ .

**The multi-crime-predictor approach**

The approach proposed in this paper is sketched in Fig. 1, and its meta-code is reported in Algorithm 1. The algorithm is composed of three main steps, as described in the following.

*Step 1. Multi-density Crime Hotspots detection.* The first step consists in the *detection of multi-density crime hotspots* from the original dataset, that is, areas where crime events occur with greater density than other adjacent areas. The goal of this step is to detect spatial areas of interest for crime forecasting, in order to conduct the further analysis over areas rather than single points. This step is performed by

the DISCOVERCRIMEHOTSPOTS( $\mathcal{D}$ ) method (line 1 of Algorithm 1), which returns the set  $\mathcal{CH} = \{CH_1, \dots, CH_H\}$  of *crime hotspots* and their corresponding densities  $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_H\}$ . This task has been modeled as a geo-spatial clustering instance and has been performed, as described in Sect. "Detection of multi-density crime hotspots", using the City Hotspot Detector (CHD) multi-density clustering algorithm [5]. The number of detected hotspots is automatically detected by the algorithm, and their shapes are traced without any pre-fixed division in areas. The parameter setting for CHD is chosen by adopting a parameter-sweeping methodology, that is, by running several instances of the CHD algorithm by varying their input parameters, and choosing the parameter settings that maximizes a set of internal indexes which comprises Silhouette [28], DBCV [29], CDBW [30], Calinsky-Harabaz [31], Davies-Bouldin [32].

*Step 2. Crime Time Series Extraction.* The second step consists in the *spatial data splitting* of the original crime data, based on the clustering model discovered at the previous step. In other words, the points of the original crime data events assigned to the  $i^{th}$  hotspot are transformed in a time series and gathered in the  $i^{th}$  output dataset, for  $i = 1, \dots, K$ . At the end of this step,  $K$  different time series data sets are available, each one containing the time series of crimes occurred in its associated dense region, aggregated on a weekly basis.

*Step 3. Predictive Crime Models extraction.* The third step is aimed at extracting a specific crime prediction model  $F_{crime}^i$  for each  $i^{th}$  crime hotspot, analyzing the crime data split during the previous step. This task can be done by applying different regression techniques. In particular, in our approach this task has been implemented by exploiting both SARIMA and LSTM techniques (which have been resulted the most effective approaches to this purpose), as described in Sect. "Extraction of crime predictors".

**Algorithm 1** MultiCrimePredictor

---

```

Require:
     $\mathcal{D}$ : crime dataset;
Ensure:
     $\mathcal{CH} = \{CH_1, \dots, CH_K\}$ : a set of  $K$  multi-density crime hotspots;
     $\Sigma = \{\sigma_1, \dots, \sigma_K\}$ : a set of  $K$  crime hotspot densities;
     $\mathcal{F} = \{F_{crimes}^1, \dots, F_{crimes}^K\}$ : a set of  $K$  crime predictors;
1:  $\{(CH_1, \sigma_1), \dots, (CH_K, \sigma_K)\} \leftarrow \text{DISCOVERCRIMEHOTSPOTS}(\mathcal{D})$ 
2:  $\{D_1^+, \dots, D_K^+\} \leftarrow \text{BUILDCRIMETSADATA}(\mathcal{D}, CH_1, \dots, CH_K)$ 
3: for each  $k = 1, \dots, K$  do
4:    $F_k \leftarrow \text{DISCOVERLOCALCRIMEPREDICTOR}(D_k^+)$ 
5:    $\mathcal{F} \leftarrow \mathcal{F} \cup F_k$ 
6: end for
7: return  $\mathcal{CDR}, \Sigma, \mathcal{F}$ 

```

---

**Detection of multi-density crime hotspots**

The detection of crime hotspots has been done by exploiting the CHD algorithm [5], a multi density-based clustering algorithm that has been purposely designed for processing urban spatial data and discover multi-density hotspots. The algorithm is composed of several steps, as reported in Algorithm 2. First, given a fixed  $k$  variable, the  $k$ -nearest neighbors distance for each point is computed and exploited as an estimator of the density of each data point (line 1). Then, the points are sorted with respect to their estimated density, and the density variation between each consecutive couple of points in the

ordered list is computed (line 2). The obtained density variation list can show very frequent fluctuations between subsequent values (in particular, in the analysis of real-world urban data), thus a moving average filtering over windows of size  $s$  is applied to smooth out such fluctuations and highlight main trends (line 3). The data points are then partitioned into several *density level sets* (each one characterized by homogeneous density distributions), on the basis of the smoothed density variations (line 4). Then, a different  $\epsilon$  value is estimated for each density level set (line 5). Finally, each set is analyzed by the DBSCAN algorithm (lines 7–12). Specifically, each instance takes as input the specific  $\epsilon$  value computed for the analyzed density level set. The set of clusters detected for each partition constitutes the final result of the CHD algorithm. More details about CHD can be found in [5]. Moreover, in Cesario et al. [25] CHD has been proven to be effective in detecting clusters characterized by different densities in urban spatial datasets.

**Algorithm 2** The CityHotspotDetector algorithm

---

**Require:**

$\mathcal{D}$ : urban event dataset;  
 $k$ : an integer value to compute the  $k$ -neighborhood density;  
 $\omega$ : a coefficient for the density variation threshold;  
 $s$ : smoothing window size;

**Ensure:**

$UH = \{uh_1, uh_2, \dots, uh_H\}$ : a set of  $H$  city hotspots;  
 $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_H\}$ : a set of  $H$  hotspot densities;  
1:  $kDistList \leftarrow COMPUTEKDIST(\mathcal{D}, K)$   
2:  $denVarList \leftarrow COMPUTEDENSITYVARIATION(kDistList)$   
3:  $smoothDenVarList \leftarrow MOVINGAVERAGEFILTERING(denVarList, s)$   
4:  $\tau \leftarrow \mu(smoothDenVarList) + \omega \cdot \sigma(smoothDenVarList)$ ;  
5:  $densityLevelSetsList \leftarrow PARTITIONDENSITYVARIATION(\mathcal{D}, smoothDenVarList, \tau)$ ;  
6:  $epsList \leftarrow COMPUTE EPSVALUES(densityLevelSetsList)$ ;  
7: **for** each  $\epsilon_i$  in  $epsList$  **do**  
8:    $DLS_i \leftarrow densityLevelSet[i]$   
9:    $(uh_{\epsilon_i}, \sigma_{\epsilon_i}) \leftarrow DBSCAN(DLS_i, \epsilon_i, k)$   
10:    $UH \leftarrow UH \cup uh_{\epsilon_i}$   
11:    $\Sigma \leftarrow \Sigma \cup \sigma_{\epsilon_i}$   
12: **end for**  
13: **return**  $(UH, \Sigma)$

---

### Extraction of crime predictors

Given a specific crime hotspot, the DISCOVERLOCALCRIMEPREDICTOR() method (line 4 in Algorithm 1) extracts a regressive model to forecast the number of crimes that will happen in its specific area. In this paper, this has been performed by exploiting SARIMA (Seasonal AutoRegressive Integrated Moving Average) and LSTM (Long Short-Term Memory) models. Such models and their principles are briefly summarized in the following.

#### SARIMA models

Multiple regression models have been defined with the goal of forecasting a variable of interest using a linear combination of predictors [33]. In particular, in an *auto-regression model*, the variable of interest is forecasted using a linear combination of its past values (the term auto-regression indicates that it is a regression of the variable against itself), while a *moving average model* uses past forecast errors in a regression-like model. Sometimes, as a preliminary step to the regressive analysis, time series need a *differencing*



transformation to stabilize the mean of a time series and so eliminating (or reducing) trend and seasonality. A combination of differencing, auto-regression and moving average methods is known as *AutoRegressive Integrated Moving Average* model (more frequently referred by its acronym *ARIMA*) [33], formally defined in the following.

Let us consider the time series  $\{y_t : t = 1 \dots n\}$ , where  $y_t$  is the value of the time series at the timestamp  $t$ . Then, an *ARIMA*( $p, d, q$ ) model is written in the form

$$y_t^{(d)} = c + \phi_1 y_{t-1}^{(d)} + \dots + \phi_p y_{t-p}^{(d)} + \theta_1 e_{t-1} + \dots + \theta_q e_{t-q} + e_t$$

where:

- $y_t^{(d)}$  is the  $d^{th}$ -differenced series of  $y_t$ , that is:  $y_t^{(d)} = y_t^{(d-1)} - y_{t-1}^{(d-1)}, \dots, y_{t-p}^{(d)} = y_{t-p}^{(d-1)} - y_{t-p-1}^{(d-1)}$   
 $y_t^{(d)} = y_t^{(d-1)} - y_{t-1}^{(d-1)}, \dots, y_{t-p}^{(d)} = y_{t-p}^{(d-1)} - y_{t-p-1}^{(d-1)}$
- $\phi_1, \dots, \phi_p$  are the regression coefficients of the auto-regressive part;
- $\theta_1, \dots, \theta_q$  are the regression coefficient of the moving average part;
- $e_{t-1}, \dots, e_{t-q}$  are lagged errors;
- $e_t$  is white noise and takes into account the forecast error;
- $c$  is a correcting factor.

The regression model above described is referred as *ARIMA*( $p, d, q$ ), where the order of the model is stated by three parameters:  $p$  (order of the auto-regressive part),  $d$  (degree of first differencing involved) and  $q$  (order of the moving average part). A useful notation commonly adopted when treating this kind of models is the 'backshift notation' [34–36], that is based on the  $B$  operator. The  $B$  ( $B^d$ ) operator on  $y_t$  has the effect of shifting the data back one period ( $d$  periods). This is very useful when combining differences, as the operator can be treated using ordinary algebraic rules. By using the 'backshift' operator, the full model can be written as:

$$(1 - \phi_1 B - \dots - \phi_p B^p)(1 - B)^d y_t = (1 - \theta_1 B - \dots - \theta_q B^q) e_t$$

whose details are out of the scope of this work and a formal demonstration can be found in [33–35].

In order to deal with seasonality, the classical *ARIMA* processes have been generalized and extended by the *SARIMA* (i.e., *Seasonal ARIMA*) models. A *SARIMA* model is formed by including additional seasonal terms (modeling a seasonal component that repeats with a given periodicity) in the classic *ARIMA* models previously introduced. The seasonal part of the model consists of terms that are very similar to the non-seasonal components of the model. In the final formula, the additional seasonal terms are simply multiplied with the non-seasonal terms. A seasonal *ARIMA* model is referred as *SARIMA*( $p, d, q$ )( $P, D, Q$ ) $_m$ , where  $m$  is a periodicity factor.

The *SARIMA* model can be written as [15]:

$$\phi_p(B) \Phi_P(B^m) \prod_{d=1}^d \prod_{m=1}^D y_t = \theta_q(B) \Theta_Q(B^m) e_t$$

where  $p$  and  $q$  represent non-seasonal ARIMA order,  $P$  and  $Q$  represent seasonal ARIMA order,  $d$  is the number of time differences and  $D$  is the number of seasonal difference.  $B$  is the backshift operator and is defined such that  $y_t B^s = y_{t-s}$ .  $\phi_p(B) = (1 - \phi_1 B - \dots - \phi_p B^p)$  is the AR operator and  $\theta_q(B) = (1 - \theta_1 B - \dots - \theta_q B^q)$  is the MA operator.  $\Phi_P(B^m) = (1 - \Phi_m B^m - \dots - \Phi_{Pm} B^{Pm})$  is the seasonal AR operator and  $\Theta_Q(B^m) = (1 - \Theta_m B^m - \dots - \Theta_{Qm} B^{Qm})$  is the seasonal MA operator.  $y_t$ , which has both seasonal and non-seasonal components, is differenced  $d$  times (length one) and  $D$  times (length  $m$ ).  $\nabla^d = (1 - B)^d$  is the non-seasonal differencing operator and  $\nabla_m^D = (1 - B^m)^D$  is the seasonal differencing operator.  $e_t$  is the random shocks that are not autocorrelated.

Once the differencing order has been chosen i.e.  $d$  and  $D$  values, the estimation of the best model order and the regression coefficient values is performed by applying the Hyndman-Khandakar’s algorithm. Briefly, the algorithm performs a step-wise search to traverse the model space and discover the optimal combination of  $p, q, P$  and  $Q$  values, which is based on the minimization of the AIC (Akaike’s Information Criterion) [33]. Then, the estimation of the regression parameters of both seasonal (i.e.,  $\phi_1, \dots, \phi_p$  and  $\theta_1, \dots, \theta_q$ ) and non-seasonal part ( $\Phi_1, \dots, \Phi_p$  and  $\Theta_1, \dots, \Theta_q$ ) is obtained by maximizing the MLE (Maximum Likelihood Estimation) [33], i.e., the probability of fitting the data that have been observed.

**LSTM**

The LSTM model is a recurrent neural system designed to overcome the exploding/vanishing gradient problems that typically arise when learning long-term dependencies, even when the minimal time lags are very long [16]. The LSTM architecture consists of a set of recurrently connected sub-networks, known as memory blocks. The idea behind the memory block is to maintain its state over time and regulate the information flow through non-linear gating units [37]. The output of the block is recurrently connected back to the block input and to all of the gates. As shown in Fig. 2 LSTM has an internal state variable, which is passed from one cell to the subsequent, and modified by the following Operation Gates [37]:

- *Forget gate*: it is a sigmoid layer that takes the output at  $t - 1$  and the current input at time  $t$ , concatenates them and applies a linear transformation followed by a sigmoid:

$$f^{(t)} = \sigma(W_f[h^{(t-1)}, x^t] + b_f)$$

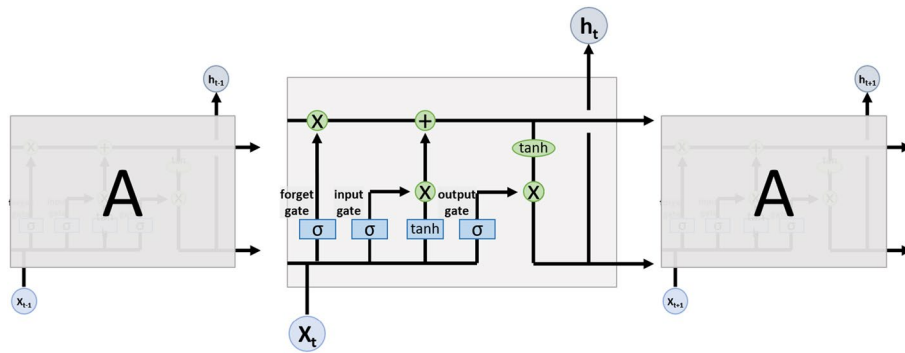
- *Input gate*: it takes the previous output and the new input and passes them through another sigmoid layer, so this gate returns a value between 0 and 1.

$$i^{(t)} = \sigma(W_i[h^{(t-1)}, x^t] + b_i)$$

This value is multiplied with the output of the candidate layer:

$$C^{(t)} = \tanh(W_c[h^{(t-1)}, x^t] + b_c)$$

The candidate layer applies a hyperbolic tangent returning a candidate vector to be added to the internal state, which is updated as follows:



**Fig. 2** LSTM architecture

$$C^{(t)} = f^{(t)} C^{(t)} + i^{(t)} C^{(t)}$$

The previous state is multiplied by the forget gate and then added to the fraction of the new candidate allowed by the output gate.

- *Output gate*: it controls how much of the internal state is passed to the output and it works in a similar way to the other gates:

$$o^{(t)} = \sigma(W_o[h^{(t-1)}, x^t] + b_o)$$

$$h^{(t)} = o^{(t)} \tanh(C^{(t)})$$

Once the number of layers, the number of nodes/units and the activation function per layer have been chosen, the estimation of the best model weights is performed by applying the backpropagation algorithm, i.e. one of the most popular neural network algorithms exploited to compute the necessary correction of weights that have been set randomly at first. Briefly, the algorithm can be decomposed in the following steps [38]:

- *Feed-forward computation*: given an input for the network, the output is computed by evaluating the network layer by layer, from the input to the output layers.
- *Back propagation*: the error (loss) of the output layer is computed by comparing it with the reference. Once the layer error has been identified, it is exploited to compute the error for the previous layer, thus propagating it backward. This is repeated for all the layers back to the input one.
- *Weight updates*: as the errors in all the network layers have been computed, the weights are changed in order to reduce the error, by exploiting the gradient descent algorithm.

The algorithm is stopped when the changes in the value of the chosen loss function become lower than a given threshold value.

## Experimental evaluation and results

To assess the performance and usefulness of the algorithm described above, we conducted an extensive experimental analysis by running several experiments in a real-world case study represented by a large area of Chicago. Our analysis aims to identify the most significant multi-density crime hotspots and build efficient prediction models that can forecast the number of future crimes likely to occur in each hotspot. We also present a comparative analysis between SARIMA and LSTM forecasting models. The rest of this section is organized as follows. Section "Data description" describes the area selected for the analysis and the gathered data, Sect. "Crime hotspots: results and discussion" reports the results in terms of multi-density crime hotspots, and Sect. "Crime forecasting models: results and discussion" describes the evaluation of the regressive models, i.e., SARIMA and LSTM, comparing the achieved accuracy to predict crimes in the detected hotspots. Sect. "Comparative analysis with ST-OPTICS on hotspots detection and crime forecasting" furnishes a comparative evaluation of CHD and ST-OPTICS, establishing a contrast in crime prediction accuracy between hotspots based on CHD and those based on ST-OPTICS. Finally, Sect. "Comparison with other crime forecasting approaches on the Chicago Crimes dataset" reports a comparison of the performances between MD-CrimePredictor with other crime forecasting approaches [21–23] proposed in literature.

### Data description

The data that we used to train the models and perform the experimental evaluation has been gathered from the *Chicago Data Portal*, a publicly available data search and exploration platform designed and currently managed by the City of Chicago.<sup>1</sup> In particular, crime data have been gathered from the 'Crimes—2001 to present' dataset, a real-life collection of instances describing criminal events that occurred in Chicago from 2001 to the present. Each crime is described by several attributes (e.g., type of crime, location, date, community area).<sup>2</sup>

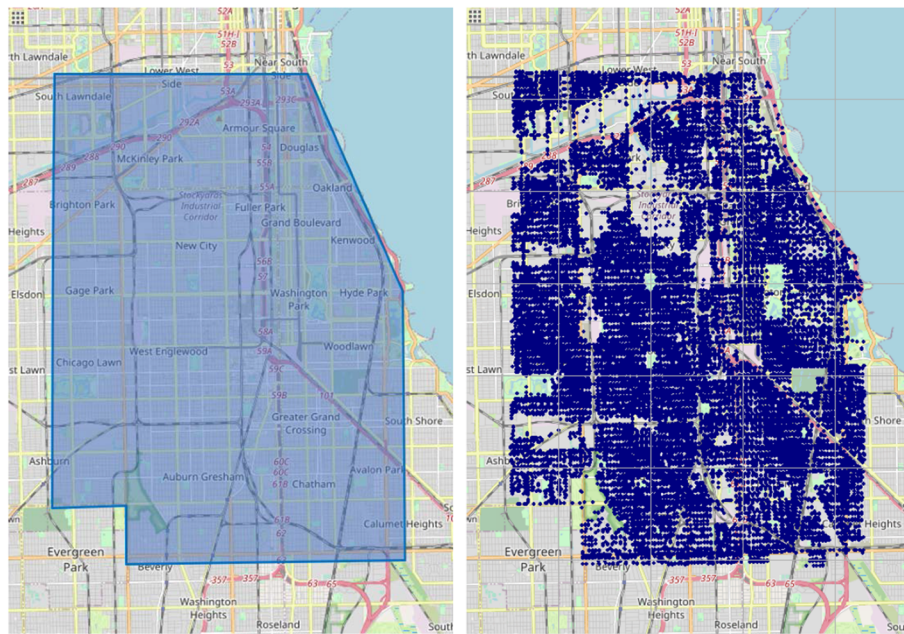
In this work, we focus our experiments on a large area of Chicago, whose boundaries and collected geo-localized crime events are shown in Fig. 3a and b, respectively. The chosen region encompasses several city neighborhoods, each one experiencing different population and commercial activity growths, with different crime densities over their territory (so making it interesting for multi-density crime analysis). Its perimeter is about 50 KM and its area is approximately 157 KM<sup>2</sup>. Starting from the 'Crimes—2001 to present' dataset, we collected all crime events within the bounded area over 19 years, from January 2001 to December 2019. The total number of collected crimes is 2,306,670, while the average number of crimes per week is 2328. The total size of the whole dataset is 167 MB.

Figure 4a and b show a preliminary view of the collected crime data, which provides some insights about data trends and distribution. In particular, Fig. 4a plots the number of collected crimes versus the time of observation. The plot immediately reveals some interesting insights. First, it is evident that the number of crimes is decreasing over the time period, showing a general clear *decreasing trend* from 2001 to 2015 in

---

<sup>1</sup> <https://data.cityofchicago.org/>.

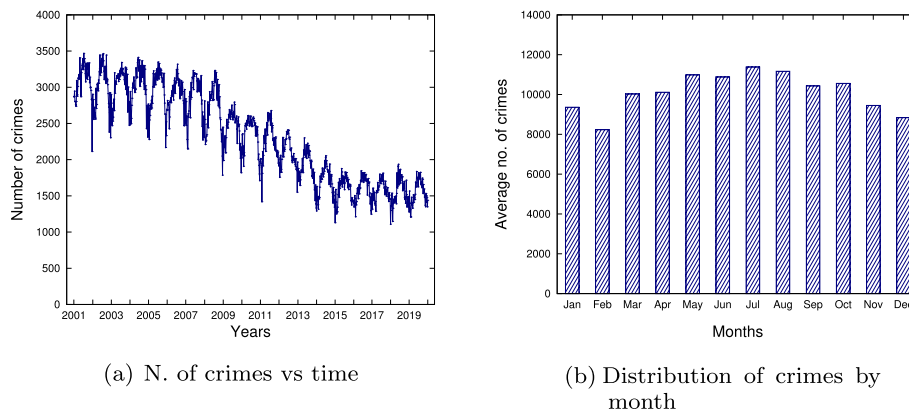
<sup>2</sup> <https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present-Dashboard/5cd6-ry5g>.



(a) Polygon of the area

(b) Geo-localized crimes

**Fig. 3** Selected area of Chicago and geolocalized crime events (2001–2019)



(a) N. of crimes vs time

(b) Distribution of crimes by month

**Fig. 4** Number of crimes vs time and their distribution by month

the data, and a stable trend from 2015 to 2019. Second, a recurring *seasonal pattern* within each year is easily discernible, whose magnitude appears to get smaller as the total number of crimes in the series decreases. By observing the plot, we can see that the number of crimes tends to rise in the late Spring, achieves their peak in the Summer, decreases in the Autumn, and generally declines in the Winter. Figure 4b plots the distribution of the average number of crimes by month, thus providing a clearer picture of the seasonality pattern hidden in the data. The histogram shows significant seasonal variations in the number of crimes during the year. In particular, the number of criminal events is highest in July (with 11,380 crimes on average), and lowest in February (with 8234 crimes, on average).

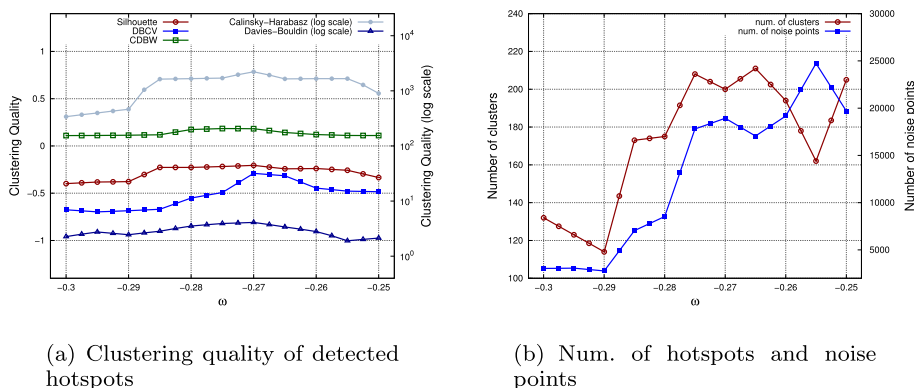
To perform the regression task and its validation, we split the original dataset into two partitions: the training set and the test set. The first is used to discover the relationships inside data, while the second is used for evaluating whether the discovered relationships hold generally. In our case, the overall crime data set has been split with respect to the number of years: the training set contains the crime data of the first 15 years (2001–2016), while the test set holds the crime data of the last 3 years (2017–2019). As described in the following sub-sections, we trained the knowledge model (i.e., crime-dense regions and crime predictors) on the training set, and we used the trained model to forecast the crime events on the test set, so to assess the quality of the predictions in each hotspot.

### Crime hotspots: results and discussion

As described in Sect. "The multi-crime-predictor approach", crime hotspots are detected by applying the CHD algorithm. However, in order to detect high-quality crime-dense regions, it is necessary to tune the key parameters of the algorithm so as to improve the results' performance. Specifically, the CHD algorithm requires setting  $k$ ,  $\omega$ , and  $s$ . In particular, the values of  $\omega$  and  $k$  have a direct influence on the quality of the results, and thus it is critical to choose their value to achieve the right balance among separability, compactness, and significance of detected hotspots. To show the best results achievable by the algorithm, we adopted a parameter-sweeping methodology, that is, we run several instances of the algorithms by varying their input parameters. Then, we select the best result, in terms of clustering quality achieved by the algorithm, which best suits our application scenario and the considered dataset. In particular, in our case, the clustering quality can be computed by *internal validation measures* [39], which evaluate the goodness of a clustering structure without respect to external labels. To do so, the following set of *internal indexes* are here adopted: Silhouette [28], DBCV [29], CDBW [30], Calinsky-Harabaz [31], Davies-Bouldin [32], which are used in literature to evaluate the clustering quality in terms of compactness, separation, number of clusters and density when no external information is available [39].

The first set of experimental results is reported in Fig. 5, which shows the performance achieved by the CHD algorithm with  $\omega$  varying from  $-0.3$  to  $-0.25$ . In particular, Figure 5a shows how the aforementioned internal indexes, evaluating the clustering quality, vary with respect to  $\omega$  values. We can observe that the quality of detected hotspots is very sensitive to  $\omega$ , whose best value, in this case, can be clearly estimated as equal to  $\omega^* = -0.27$ . On the other side, Figure 5b shows how the number of noise points (blue curve) and the number of detected hotspots (red curve) vary with respect to  $\omega$  values. Noise points are data instances that do not meet the criteria for falling into any of the detected clusters (and are considered outliers by the algorithm), while the number of detected hotspots depends on the algorithm's ability to find a balanced trade-off between separability and compactness properties. We can observe that for  $\omega^* = -0.27$ , the number of detected noise points is 18,929, while the number of detected clusters is 200.

As reported above, we have run several experimental tests to find the parameter settings capable of detecting the highest-quality city hotspots. For such a reason, in the following, we present the results achieved by fixing  $\omega = -0.27$ ,  $k = 64$ ,  $s = 5000$ , which

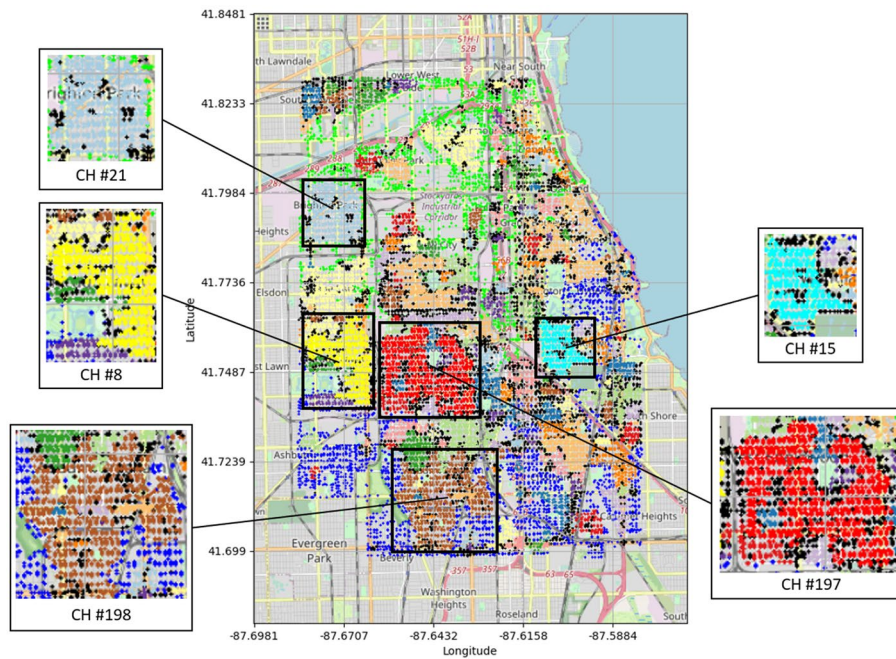


**Fig. 5** CHD clustering quality, num. of hotspots and num. of noise points vs  $\omega$ , with  $k = 64$  and  $s = 5000$

have been assessed to best suit our application scenario and the considered dataset by the previous analysis.

Now, let us analyze more in detail the crime hotspots detected in the considered scenario. As reported in Sect. "The multi-crime-predictor approach", the clustering algorithm exploited in this work first partitions the original data in several density level sets (each one characterized by homogeneous density distributions on the basis of density variations), then analyzes each density level set through a specific density-based clustering algorithm to detect proper clusters in each partition. The final hotspots (i.e. totally 200) discovered by the algorithm are shown in Fig. 6, where a different color represents each region. Interestingly, this image shows how crime events are clustered on the basis of a density criterion; for example, the algorithm detects several significant crime regions clearly recognizable through different colors: a large crime region (in red) in the central part of the area along with seven smaller areas (in green, blue and light-blue) on the left and right side, corresponding to zones with the highest concentration of crimes. The five most relevant crime hotspots (*CH#197*, *CH#198*, *CH#8*, *CH#21*, and *CH#15*) are zoomed-in on the left and right sides of Fig. 6. Many other hotspots are detected, representing areas having minor crime-densities w.r.t. the highlighted ones, or local high-density crime zones surrounded by low-density ones. Table 4 shows several statistics about the whole area and the five most numerous crime hotspots. Overall, these regions cover about 22% of the whole area extension and about 55% of the crime events detected in the whole area between 2001 and 2019.

Finally, in order to make a comparative analysis among classic density-based algorithms and multi-density approaches for hotspots detection, we report here a comparative table (Table 5) showing the results of four algorithms (two classic approaches: DBSCAN and OPTICS-Xi, and two multi-density approaches: CHD and HDBSCAN). Table 5 shows, for each algorithm, the selected input parameters and some statistics related to the achieved results (i.e., number of detected hotspots, percentage of noise points, Silhouette evaluation measure) on the Chicago crime dataset exploited in this paper and described in Sect. "Data description". By observing the results in Table 5, we can observe that HDBSCAN and CHD achieve higher clustering qualities than DBSCAN and OPTICS-Xi; in fact, HDBSCAN and CHD (multi-density algorithms)



**Fig. 6** Detected crime hotspots in the selected area of Chicago, whose the top-5 largest ones are zoomed-out on the left and right

**Table 4** Descriptive statistics—whole area and crime hotspots

Cluster	Extention (km <sup>2</sup> )	Extention (%)	Perimeter	Crimes (#)	Crimes (%)	Crime density (n. of crimes/km <sup>2</sup> )
Whole Area	156.77	100.00	49.57	2,306,670	100.00	14,713.72
CH #197	12.57	8.02	5.34	565,790	24.53	45,011.14
CH #198	10.24	6.53	4.27	352,674	15.29	34,440.82
CH #8	5.45	3.48	2.59	170,740	7.40	31,328.44
CH #21	4.83	3.08	2.22	102,763	4.46	21,275.98
CH #15	1.53	0.98	1.53	84,039	3.64	54,927.45

**Table 5** Comparative results achieved by DBSCAN, OPTICS-Xi, CHD and HDBSCAN to detect crime hotspots, on the Chicago crime dataset [25]

	Input parameters	# Hotspots	# Noise points (%)	Silhouette index
DBSCAN	$\epsilon = 500, \text{minPoints} = 60$	78	12.6	-0.28
OPTICS-Xi	$\xi = 0.05, \text{minPoints} = 60$	279	71.9	-0.46
CHD	$\omega = -0.27, k = 64, s = 5000$	181	5.7	-0.23
HDBSCAN	$\text{min\_cluster\_size} = 200, \text{minPoints} = 60$	61	34.6	-0.19

assess on silhouette values equal to -0.19 and -0.23, respectively, which are better than DBSCAN and OPTICS-xi’s results, whose clustering qualities assess on -0.28 and -0.46. Such results show that multidensity clustering (i.e., HDBSCAN and CHD) is able to distinguish and identify proper hotspots in urban environments better than



classic density-based techniques. Moreover, focusing on the two multi-density algorithms CHD and HDBSCAN results, we can observe that CHD achieves a slightly lower silhouette than HDBSCAN, but it labels a very lower percentage of noise points (5.7%) with respect to HDBSCAN (34.6%). For such a reason, CHD resulted the best algorithm to be exploited in our crime data analysis case study. A more detailed analysis about the comparison among such algorithms is reported in [25].

### Crime forecasting models: results and discussion

As described in Sect. "The multi-crime-predictor approach", the next steps of the algorithm consist of (i) transforming the original crime data set in several time series, and (ii) training local crime predictors for each crime hotspot. In particular, as described in Sect. "Extraction of crime predictors", the extraction of crime regressors has been performed by applying SARIMA and LSTM models on each hotspot. Specifically, we present here the details of the regressive models obtained by both algorithms for the whole area and the three largest crime hotspots, i.e., CH#197, CH#198, and CH#8. Then, we will show the predictive performance of the models on the test set for all hotspots.

The regressive models extracted by SARIMA are reported in Table 6. For each area, the table shows the order of the models, the final autoregressive formulas (in back-shift notation), and the final coefficient values. It is worth noting that the predictive crime models differ among the hotspots, showing that each area presents specific crime trends and patterns, thus making the discovery of different predictive models reasonable.

The models extracted by LSTM are reported in Table 7. For each area, neural networks are trained with 4 layers, *ReLu* [40] activation function, a number of epochs equal to 50, and a customised batch size and number of units/nodes per layer. In each of the models presented, the mean absolute error (*mae*) loss function is considered. One of the most important factors in neural network training is the learning rate, a customized hyperparameter with a small positive value between 0.0 and 1.0 [41]. The rate at which weights are changed during the training is known as the step size or learning rate. A learning rate of 0.01 produced superior results in the NN models reported here than other learning rates. Even in the case of LSTM models, each hotspot has specific crime trends and patterns.

In order to assess the effectiveness and accuracy of the regressive functions, we performed an evaluation analysis on the test set consisting of the last three years of data (i.e., years 2017–2019). In particular, for each crime hotspot and for the whole area, their associated SARIMA and LSTM models have been exploited to predict the number of crimes that are likely to happen in that hotspot, week by week. Figures 7 and 8 show observed, SARIMA-forecasted and LSTM-forecasted data (plotted in blue, orange and green, respectively), for the whole area and the crime hotspot CH#197 (the largest one), respectively. We consider here four prediction horizons on the test set, from one to four-week ahead. We note that forecasts generally adhere very well to the observed data over the whole test set period. However, the forecasting accuracy clearly decreases (in particular for LSTM) with the increase of the prediction horizon.

**Table 6** Details of the SARIMA models trained for the whole area and the top 3 largest crime hotspots in Chicago

Whole area	
Model:	SARIMA(3, 1, 3)(1, 1, 2) <sub>52</sub> , MLE = -5903.48, AIC = 11826.96
Backshift notation:	$y_t = \frac{(1 - \theta_1 B - \theta_2 B^2 - \theta_3 B^3)(1 - \Theta_1 B^{52} - \Theta_2 B^{104}) e_t}{(1 - \phi_1 B - \phi_2 B^2 - \phi_3 B^3)(1 - \Phi_1 B^{52})(1 - B)(1 - B^{52})}$
Coefficient values:	$\phi_1 = -0.0286, \phi_2 = 0.7987, \phi_3 = -0.0589, \theta_1 = -0.6755, \theta_2 = -0.8928, \theta_3 = 0.5949, \Phi_1 = 0.9862, \Theta_1 = -1.7097, \Theta_2 = 0.7109$
Crime Hotspot 197	
Model:	SARIMA(1, 1, 3)(1, 1, 1) <sub>52</sub> , MLE = -4861.32, AIC = 9736.64
Backshift notation:	$y_t = \frac{(1 - \theta_1 B - \theta_2 B^2 - \theta_3 B^3)(1 - \Theta_1 B^{52}) e_t}{(1 - \phi_1 B)(1 - \Phi_1 B^{52})(1 - B)(1 - B^{52})}$
Coefficient values:	$\phi_1 = 0.7560, \theta_1 = -1.5868, \theta_2 = 0.6273, \theta_3 = -0.0215, \Phi_1 = 0.0657, \Theta_1 = -0.8027,$
Crime hotspot 198	
Model:	SARIMA(1, 1, 2)(2, 1, 1) <sub>52</sub> , MLE = -4503.01, AIC = 9020.01
Backshift notation:	$y_t = \frac{(1 - \theta_1 B - \theta_2 B^2)(1 - \Theta_1 B^{52}) e_t}{(1 - \phi_1 B)(1 - \Phi_1 B^{52} - \Phi_2 B^{104})(1 - B)(1 - B^{52})}$
Coefficient values:	$\phi_1 = -0.8186, \theta_1 = 0.0241, \theta_2 = -0.7394, \Phi_1 = 0.1181, \Phi_2 = 0.0340, \Theta_1 = -0.9027$
Crime hotspot 8	
Model:	SARIMA(2, 1, 0)(3, 1, 3) <sub>52</sub> , MLE = -4216.93, AIC = 8451.87
Backshift notation:	$y_t = \frac{(1 - \Theta_1 B^{52} - \Theta_2 B^{104} - \Theta_3 B^{156}) e_t}{(1 - \phi_1 B - \phi_2 B^2)(1 - \Phi_1 B^{52} - \Phi_2 B^{104} - \Phi_3 B^{156})(1 - B)(1 - B^{52})}$
Coefficient values:	$\phi_1 = -0.5384, \phi_2 = -0.2661, \Phi_1 = -1.5079, \Phi_2 = -0.8666, \Phi_3 = 0.0336, \Theta_1 = 0.5728, \Theta_2 = -0.5925, \Theta_3 = -0.9798$

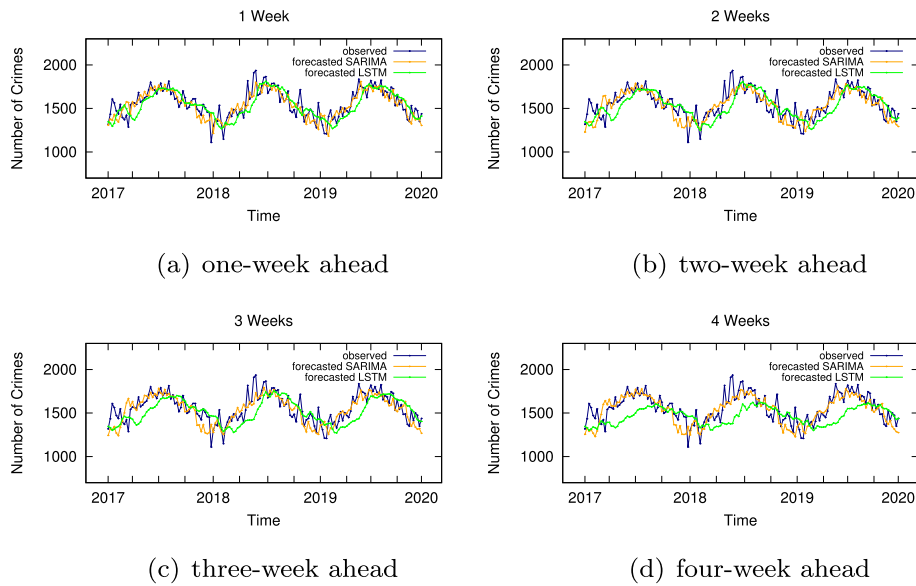
**Table 7** Details of the LSTM models trained for the whole area and the top 3 largest crime-dense regions in Chicago

Cluster	Layers	Units	Activation	Epochs	Batch size	LR	Loss	Loss value
Whole area	4	25	relu	50	26	0.01	mae	91.77
CH #197	4	50	relu	50	13	0.01	mae	26.59
CH #198	4	100	relu	50	13	0.01	mae	21.78
CH #8	4	100	relu	50	52	0.01	mae	12.99

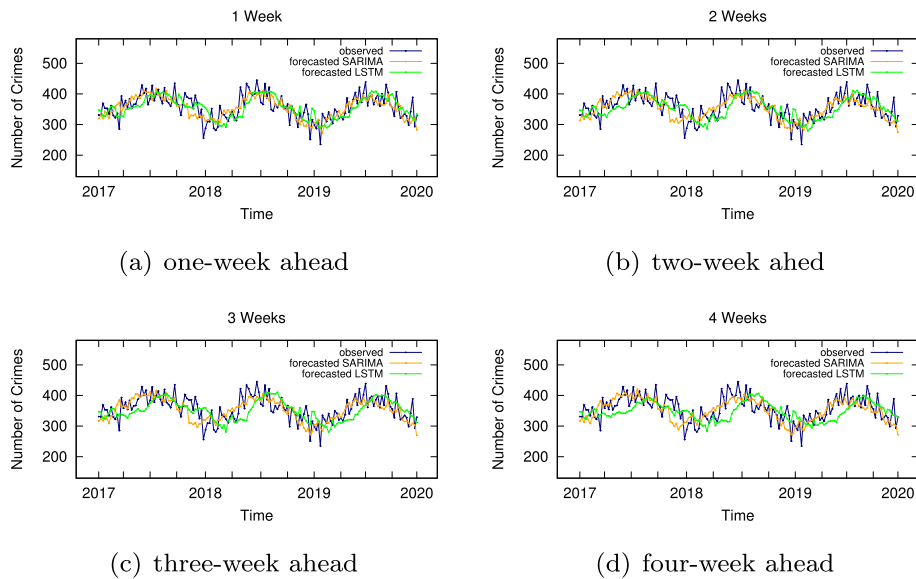
Now, let us give a quantitative evaluation of the performance of the regressive models and their effectiveness in making predictions on the corresponding test sets. To this end, we computed six error measures (MAE, MAPE, MSE, RMSE, MaxError, MeanError), which are commonly used in regressive analysis literature to quantify forecast performance [12].

Table 8 reports the values of the error measures described above achieved by SARIMA and LSTM models for the whole area and the three largest detected crime hotspots. Looking at the values reported in the table, we can make the following observations.

*The smaller hotspot, the lower MAE.* Looking at the values in the table, we can observe that MAE values decrease when hotspot areas are smaller and smaller. In



**Fig. 7** Observed vs forecasted crimes, on the whole area. Number of crimes observed, SARIMA-forecasted and LSTM-forecasted (blue, orange and green lines) on the Chicago test set, for the whole area and several prediction horizons

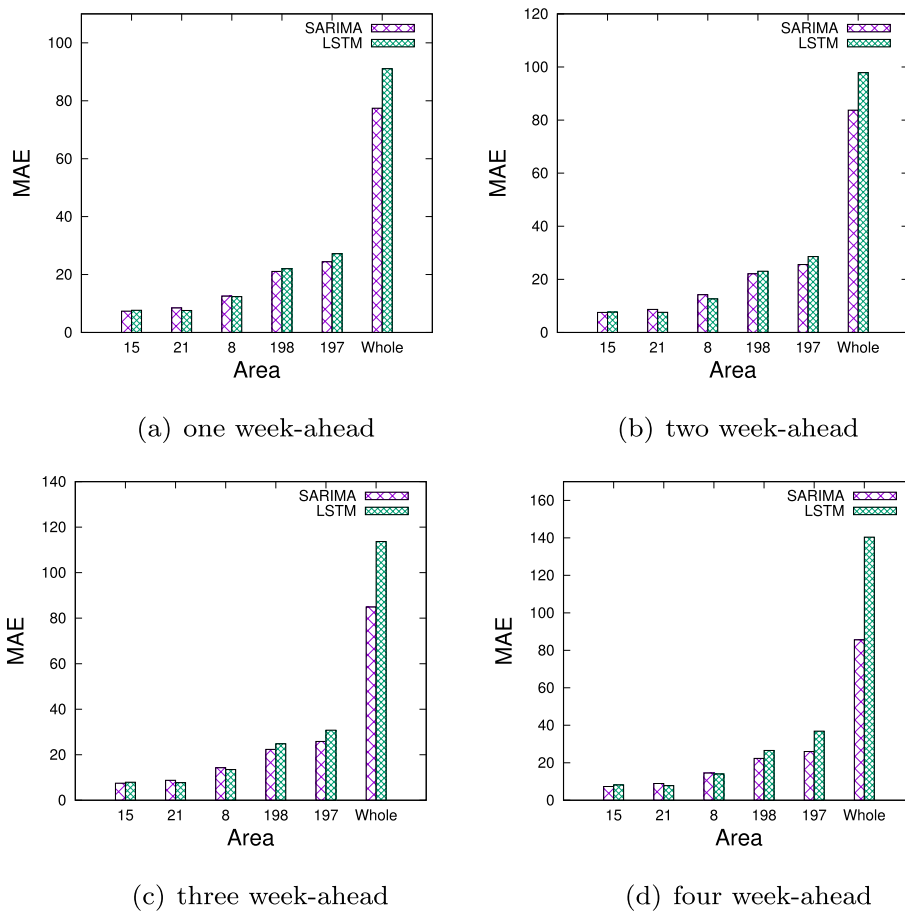


**Fig. 8** Observed vs forecasted crimes, on the largest hotspot. Number of crimes observed, SARIMA-forecasted and LSTM-forecasted (blue, orange and green lines) on the Chicago test set, for the hotspot 197 and several prediction horizons

fact, considering one-week-ahead forecasting, the MAE achieved by SARIMA models monotonously decreases from 77.44 (whole area) to 24.42, 21.09, and 12.59 (three largest crime hotspots, ordered by decreasing size), and similarly for all other forecasting horizons. LSTM forecasts show decreasing MAE values as well. The trend is clearly recognizable in Fig. 9, which plots the MAE achieved by both SARIMA and

**Table 8** MAE, MAPE, MSE, RMSE, Max Error and Mean Error vs several weekly prediction horizons, for the whole area and the top three largest crime hotspots in Chicago City

Time	MAE																				
	Cluster197			Cluster198			Cluster198			Cluster197			Cluster198			Cluster197			Cluster198		
	Area	SARIMA	LSTM	Area	SARIMA	LSTM	Area	SARIMA	LSTM	Area	SARIMA	LSTM	Area	SARIMA	LSTM	Area	SARIMA	LSTM	Area	SARIMA	LSTM
1 Week	77.45	91.06	24.43	27.19	21.09	22.00	12.55	12.36	5.09	5.93	6.93	7.74	8.09	8.36	11.44	8.36	11.44	8.36	11.44	8.36	11.44
2 Weeks	83.73	97.86	25.58	28.59	22.10	23.06	14.21	12.67	5.52	6.36	7.25	8.12	8.48	8.74	13.07	8.74	13.07	8.74	13.07	8.74	13.07
3 Weeks	84.95	113.70	25.84	30.77	22.37	24.81	14.28	13.49	5.60	7.30	7.31	8.70	8.60	9.28	13.24	9.28	13.24	9.28	13.24	9.28	13.24
4 Weeks	85.60	140.41	26.00	32.84	22.34	26.55	14.55	14.05	5.62	8.76	7.34	9.20	8.60	9.85	13.37	9.85	13.37	9.85	13.37	9.85	13.37
Time	MSE																				
Area	Cluster197			Cluster198			Cluster198			Cluster197			Cluster198			Cluster197			Cluster198		
SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM
1 Week	10168.33	13705.60	976.53	1133.28	712.58	775.33	253.91	242.98	100.84	117.07	31.25	33.66	26.69	27.84	15.93	26.69	27.84	15.93	26.69	27.84	
2 Weeks	12062.70	15902.50	1044.05	1244.81	771.26	837.58	332.75	253.10	109.83	126.11	32.31	35.28	27.77	28.94	18.24	27.77	28.94	18.24	27.77	28.94	
3 Weeks	12614.73	20709.28	1074.21	1435.11	787.37	906.63	331.36	282.77	112.32	143.91	32.78	37.88	28.06	30.11	18.20	28.06	30.11	18.20	28.06	30.11	
4 Weeks	12800.90	31416.48	1086.59	1649.76	788.77	1101.54	341.07	304.73	113.14	177.25	32.96	40.62	28.09	33.19	18.47	28.09	33.19	18.47	28.09	33.19	
Time	RMSE																				
Area	Cluster197			Cluster198			Cluster198			Cluster197			Cluster198			Cluster197			Cluster198		
SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM
1 Week	101.68	117.07	31.25	33.66	26.69	27.84	15.93	26.69	27.84	15.93	26.69	27.84	15.93	26.69	27.84	15.93	26.69	27.84	15.93	26.69	
2 Weeks	120.63	126.11	32.31	35.28	27.77	28.94	18.24	27.77	28.94	18.24	27.77	28.94	18.24	27.77	28.94	18.24	27.77	28.94	18.24	27.77	
3 Weeks	126.15	143.91	32.78	37.88	28.06	30.11	18.20	28.06	30.11	18.20	28.06	30.11	18.20	28.06	30.11	18.20	28.06	30.11	18.20	28.06	
4 Weeks	128.01	177.25	32.96	40.62	28.09	33.19	18.47	28.09	33.19	18.47	28.09	33.19	18.47	28.09	33.19	18.47	28.09	33.19	18.47	28.09	
Time	Max error																				
Area	Cluster197			Cluster198			Cluster198			Cluster197			Cluster198			Cluster197			Cluster198		
SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM	SARIMA	LSTM
1 Week	305.10	357.26	74.65	96.25	85.31	92.05	51.68	49.34	15.40	18.98	3.73	2.22	0.81	4.07	0.17	0.81	4.07	0.17	0.81	4.07	
2 Weeks	290.41	395.03	77.72	96.62	84.45	87.45	48.51	52.81	23.60	26.47	4.96	4.34	1.09	5.76	0.41	1.09	5.76	0.41	1.09	5.76	
3 Weeks	311.24	480.99	84.66	115.99	83.40	69.11	50.18	58.19	26.56	41.54	5.44	6.47	1.21	7.76	0.41	1.21	7.76	0.41	1.21	7.76	
4 Weeks	338.07	556.61	87.48	120.24	77.94	85.83	51.03	58.90	28.99	90.76	5.83	9.77	1.23	12.42	0.42	1.23	12.42	0.42	1.23	12.42	

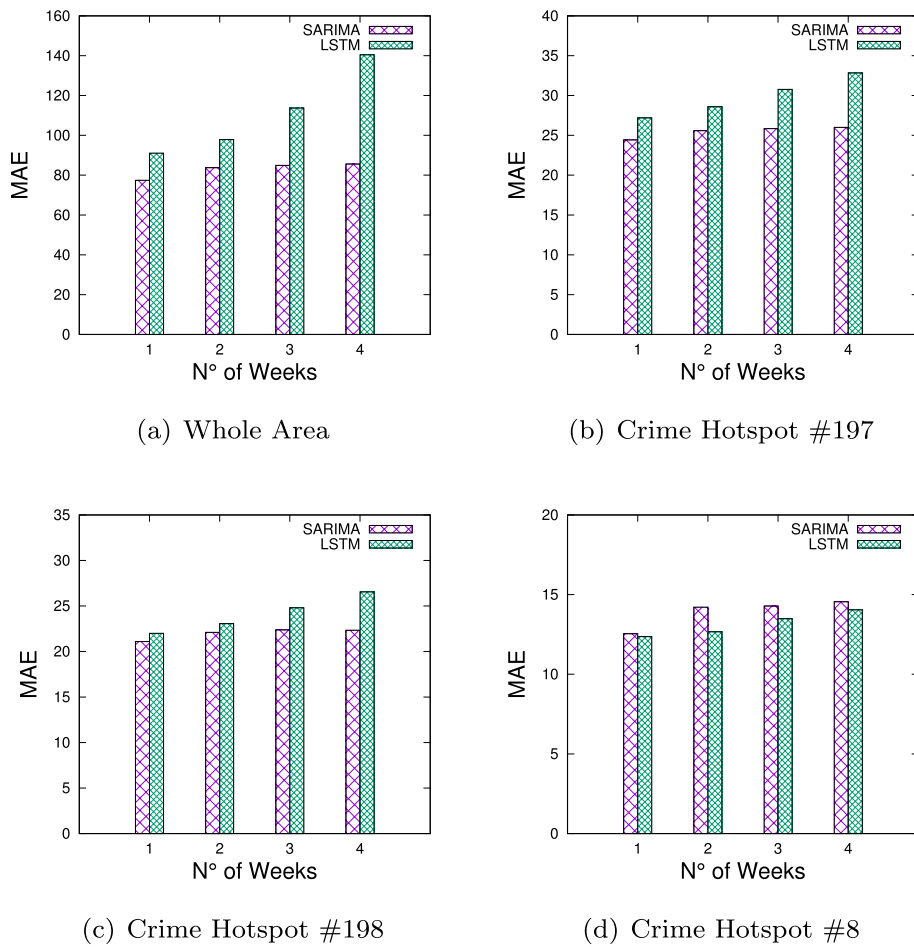


**Fig. 9** MAE for each hotspot. Mean Absolute Error (MAE) for the whole area and the top 5 largest crime hotspots, achieved by SARIMA and LSTM

LSTM for the whole area and the top five largest crime hotspots. The chart clearly shows that the smaller the hotspot, the lower the error. This is a reasonable outcome, that is, predictions are more precise when hotspot areas are smaller, thus providing city administrators and police officers with more detailed information for strategizing how to distribute resources and efforts among the various parts of the city.

*Higher forecasting accuracy when the forecasting horizon is shorter.* For example, the MAE assessed by LSTM-forecasts, by considering the whole area, monotonously increases from 91.06 (for one-week-ahead forecasts) to 97.86, 113.70 and 140.41 (for two-, three- and four-week ahead forecasts), and similarly all other indices and areas. This is a reasonable result, considering that forecasts are based on the previous historical trends: the more away is the forecasting timestamp from the most recent historical data, the less accurate the forecast. The increasing trend can also be seen in Fig. 10, which shows the MAE versus several weekly forecasting horizons. The increasing trend is more evident for the whole area and the largest cluster, and it is particularly marked for the LSTM-based forecasts.

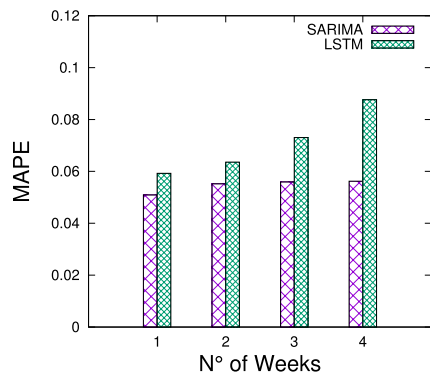
*SARIMA models outperform LSTM model (for large hotspots).* Percentage errors (MAPE column) show that the adopted SARIMA models (Table 6) forecast the number of crimes with an average error ranging from 5.09% (whole area, one-week



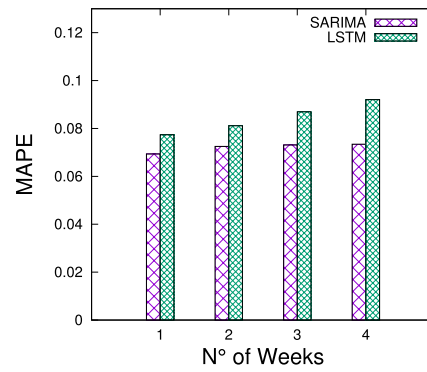
**Fig. 10** MAE vs n. of weeks. Mean Absolute Error (MAE) versus the number of weeks in the test set, achieved by SARIMA and LSTM, for the whole area and the top 3 largest crime hotspots

ahead) to 13.37% (crime hotspot #8, four-week ahead), which appears to be a very interesting result. On the other side, LSTM models assess MAPE values ranging from 5.93% to 12.81%, respectively. For a more complete view of these results, Fig. 11 shows the MAPE versus several weekly forecasting horizons. From the plot, we can observe that percentage errors of both SARIMA and LSTM models increase when the prediction horizon is longer and longer, and that generally SARIMA models outperform LSTM regressors (but for the smaller hotspot). Also, by observing the values in the Table 8 and Fig. 11, we can observe that the lower the hotspot area, the higher the percentage error. However, the MAPE index, as defined above, does not take into account the coverage level of each hotspot. The growth in forecasting errors is compensated by a more precise identification of the specific area where crime events will occur, thus giving more exhaustive information to city administrator and police officers for planning how to distribute resources and efforts in the different regions of the city.

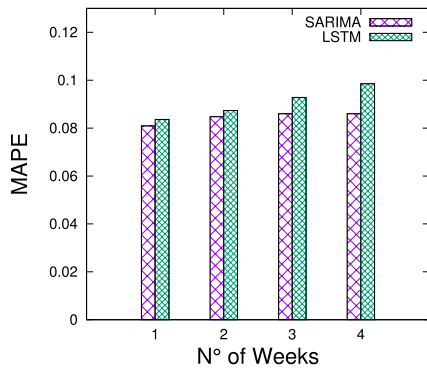
Finally, to understand whether the forecast errors can be approximated to normally distributed with mean zero and variance  $\sigma^2$ , we show in Fig. 12 the distribution of residuals (with overlaid the normal curve with the same mean and standard deviation as the



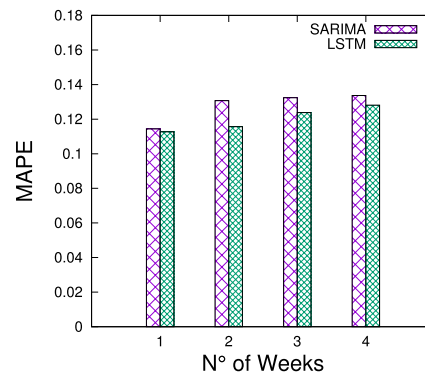
(a) Whole Area



(b) Crime Hotspot #197

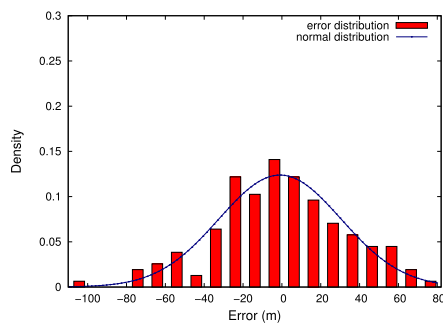


(c) Crime Hotspot #198

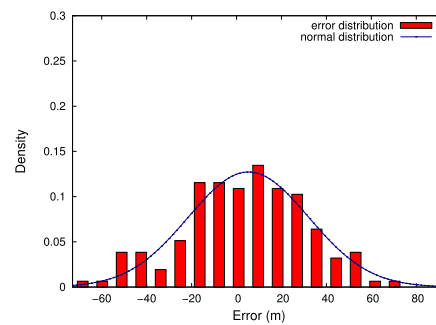


(d) Crime Hotspot #8

**Fig. 11** MAPE vs n. of weeks. Mean Absolute Percentage Error (MAPE) versus the number of weeks in the test set, achieved by SARIMA and LSTM, for the whole area and the top 3 largest crime hotspots



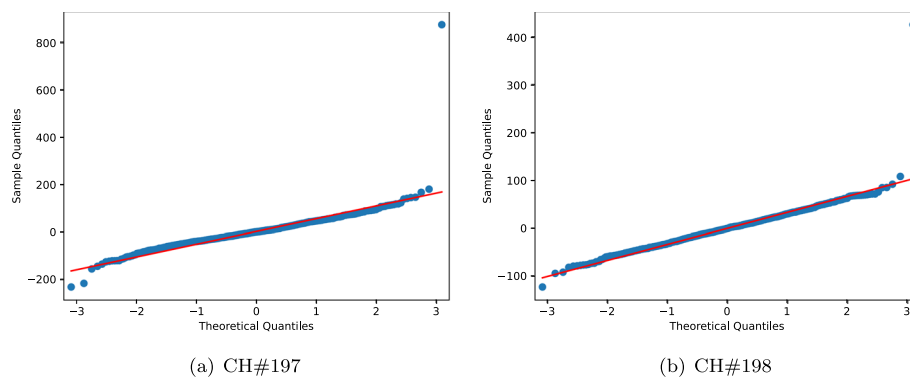
(a) CH#197



(b) CH#198

**Fig. 12** Distribution of the residuals. Distribution of the residuals (with the overlaid normal curve) on the test set, for the top 2 largest crime hotspots, for one-week ahead forecasting

distribution of forecast errors) for the two largest crime hotspots detected by SARIMA models. In particular, the figure presents the histograms of the forecast errors over one-week ahead forecasts, which show that the distributions of forecast errors are slightly



**Fig. 13** QQ-plot. QQ-plot for the top 2 largest crime hotspots

shifted towards positive or negative values compared to a normal curve (it should be centered on 0, in the ideal case). This is also confirmed by observing the Normal QQ plot (quantile-quantile plot) shown in Fig. 13, which can be exploited as a graphical tool to assess if residuals plausibly follow a normal distribution. Both plots graphically confirm that the residuals follow a normal distribution, as expected.

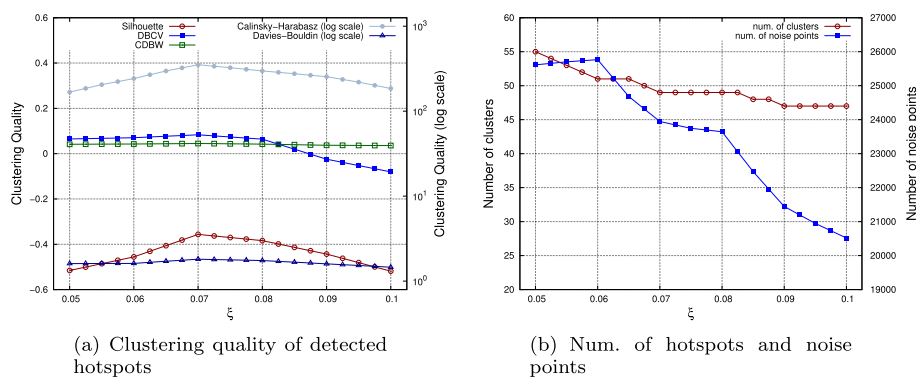
#### Comparative analysis with ST-OPTICS on hotspots detection and crime forecasting

To make our evaluation more accurate and complete, we performed a comparative analysis of the proposed approach, based on CHD for hotspot detection, with a similar approach based on ST-OPTICS [27], which is a density-based clustering algorithm specifically designed to analyze spatio-temporal data. ST-OPTICS was selected among others since it was purposely designed for clustering datasets characterized by time-based features, and thus is not directly comparable with the other spatial clustering algorithms previously mentioned (see Table 5). In a nutshell, ST-OPTICS is a modified version of the OPTICS algorithm, achieved by extending the notion of density-reachability. It exploits two radiuses,  $\epsilon_1$  and  $\epsilon_2$ , where the  $\epsilon_1$  defines the reachability with respect to spatial attributes, and  $\epsilon_2$  defines the reachability w.r.t. non-spatial (temporal) attributes; on the basis of such definitions, a point  $p_i$  is considered in the neighborhood of  $p_j$  if the distance between  $p_i$  and  $p_j$  is less than  $\epsilon_1$  w.r.t. spatial attributes, and less than  $\epsilon_2$  w.r.t. non-spatial attributes. The ST-OPTICS implementation we exploited is publicly available,<sup>3</sup> and it takes as input parameters  $(\epsilon_2, \text{min\_pts}, \xi)$ , where  $\epsilon_2$  is a threshold value on the maximum radius w.r.t. the non-spatial attributes,  $\text{min\_pts}$  is the minimum number of neighbors required to define a core-point, and  $\xi$  determines the minimum steepness on the reachability plot that constitutes a cluster boundary. The reachability plot takes into account both spatial and non-spatial radiuses. It is also worth noting that  $\text{min\_pts}$  and  $\xi$  are exploited as in the well-known OPTICS- $\xi$  algorithm.

To perform the comparative analysis between the results achieved by ST-OPTICS and CHD, we first evaluated the characteristics of the most five relevant hotspots detected by the two algorithms, and then the forecasting performance achieved for crime prediction in each hotspot. The dataset exploited for the comparative analysis is that one

<sup>3</sup> ST-OPTICS implementation on Github ([https://github.com/eren-ck/st\\_optics](https://github.com/eren-ck/st_optics)) [42]





**Fig. 14** Hotspots detection: ST-OPTICS clustering quality, num. of hotspots and num. of noise points vs  $\xi$ , with  $k = 64$  and  $\epsilon_2 = 24$

described in Sect. "Data description", and predictions have been compared versus different forecasting horizons.

As a first result, ST-OPTICS has been applied to discover spatial hotspots from the geo-referenced crime data. In order to detect high-quality crime-dense regions, an input parameters tuning has been done to achieve the best results of the algorithm. In particular, the clustering quality has been evaluated by computing the internal indexes (Silhouette, DBCV, CDBW, Calinsky-Harabasz, Davies-Bouldin) adopted in Sect. "Crime hotspots: results and discussion", by varying  $\xi$  from 0.05 to 0.1 and  $\epsilon_2$  from 4 to 24 (with step size equal to 4). The results are reported in Figure 14a, which shows the performance achieved by varying  $\xi$ , fixed  $\epsilon_2 = 24$  and  $k = 64$  (which corresponded to the optimal performance within the faced scenario). In particular, Figure 14b shows that the best quality of detected hotspots is achieved for  $\xi^* = 0.07$ . Comparing such results with those reported in Sect. "Crime hotspots: results and discussion", we notice that CHD performs better than ST-OPTICS considering Silhouette, Calinsky-Harabasz and Davies-Bouldin indexes, while ST-OPTICS is better on the DBCV index. On the other side, Figure 5b shows how the number of noise points (blue curve) and the number of detected hotspots (red curve) vary with respect to  $\xi$  values. We can observe that for  $\xi^*=0.07$ , the number of detected noise points is 23,947, while the number of detected clusters is 49. With respect to CHD, ST-OPTICS detects an higher number of noise points (23,947 versus 18,929) and a lower number of hotspots (49 versus 200). The results shown below only refer to the run with the best combination of parameters (i.e,  $\xi=0.7$ ,  $\epsilon_2 = 24$ ,  $k = 64$ ).

The comparative forecasting performance analysis on the hotspots detected by ST-OPTICS and CHD has been done by focusing on the five most numerous clusters returned by the two algorithms. In particular, as SARIMA models have shown higher predictive accuracy in Sect. "Experimental evaluation and results", we exploit here these regressive models to compare the achieved results. Table 9 reports the values of four error measures (MAPE, MAE, MSE, RMSE) achieved by SARIMA models on the five largest hotspots detected by ST-OPTICS and CHD (sorted by decreasing size), versus one-, two-, three- and four-week-ahead forecasting horizons. Looking at the values reported in the table, we can observe that the first two largest clusters detected by ST-OPTICS (clusters #0 and #4) and CHD (clusters #197 and #198) are very different in terms of number of points, while the other ones have comparable sizes. Also, by

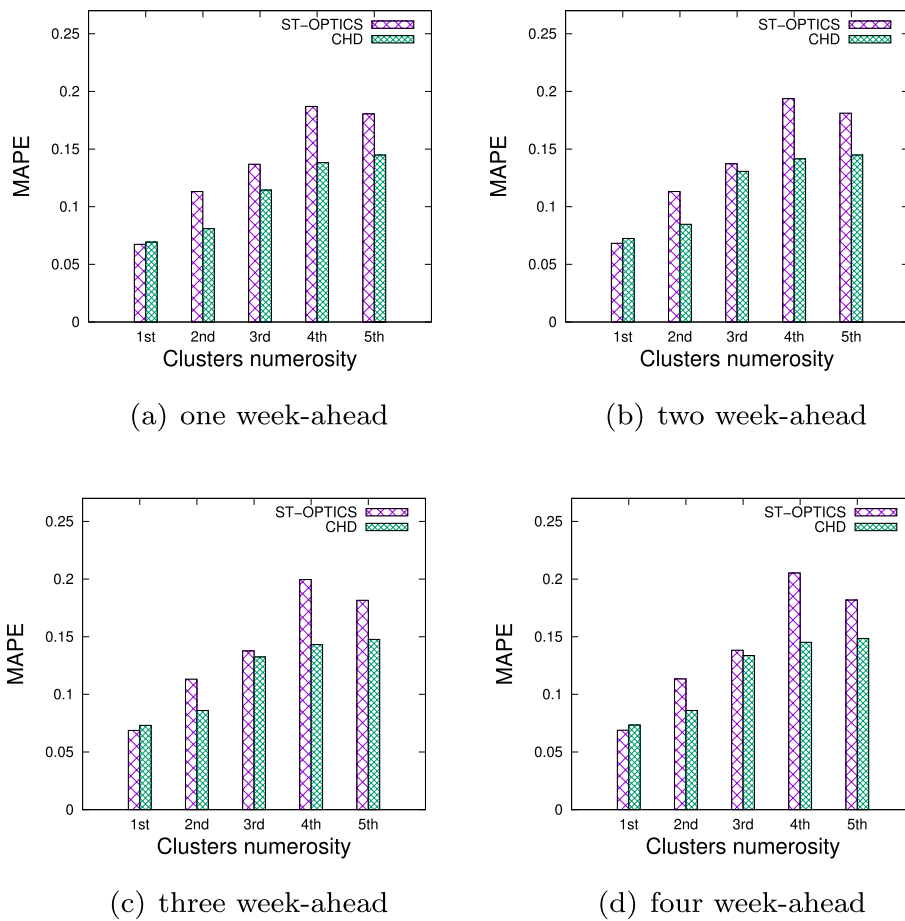
**Table 9** Crime forecasting: MAE, MAPE, MSE and RMSE for the top five most numerous crime hotspots in Chicago, detected by ST-OPTICS and CHD

1 week-ahead												
Cluster Size	ST-OPTICS					CHD						
	Cluster ID	Num. of crimes	MAPE (%)	MAE	MSE	RMSE	Cluster ID	Num. of crimes	MAPE (%)	MAE	MSE	RMSE
I	#0	800,243	<b>6.74</b>	31.98	1660.33	40.75	#197	613,759	6.93	24.43	976.53	31.25
II	#4	220,115	11.31	15.35	366.13	19.13	#198	389,164	<b>8.09</b>	21.09	712.58	26.69
III	#6	188,930	13.68	13.34	279.35	16.71	#8	183,456	<b>11.44</b>	12.55	253.91	15.93
IV	#2	101,132	18.70	10.79	172.70	13.14	#21	110,956	<b>13.82</b>	8.50	113.97	10.68
V	#1	79,267	18.05	9.10	127.25	11.28	#15	90,311	<b>14.48</b>	7.36	85.71	9.26
V	#1	79,267	18.11	9.14	125.85	11.22	#15	90,311	<b>14.48</b>	7.36	89.49	9.46
2 week-ahead												
Cluster Size	ST-OPTICS					CHD						
	Cluster ID	Num. of crimes	MAPE (%)	MAE	MSE	RMSE	Cluster ID	Num. of Crimes	MAPE (%)	MAE	MSE	RMSE
I	#0	800,243	<b>6.83</b>	32.26	1670.76	40.87	#197	613,759	7.25	25.58	1044.05	32.31
II	#4	220,115	11.31	15.35	366.20	19.14	#198	389,164	<b>8.48</b>	22.10	771.26	27.77
III	#6	188,930	13.72	14.35	280.20	16.74	#8	183,456	<b>13.07</b>	14.21	332.75	18.24
IV	#2	101,132	19.37	11.10	193.67	13.92	#21	110,956	<b>14.15</b>	8.70	117.44	10.84
V	#1	79,267	18.11	9.14	125.85	11.22	#15	90,311	<b>14.48</b>	7.36	89.49	9.46
3 week-ahead												
Cluster Size	ST-OPTICS					CHD						
	Cluster ID	Num. of crimes	MAPE (%)	MAE	MSE	RMSE	Cluster ID	Num. of Crimes	MAPE (%)	MAE	MSE	RMSE
I	#0	800,243	<b>6.87</b>	32.50	1696.90	41.19	#197	613,759	7.31	25.84	1074.21	32.78
II	#4	220,115	11.32	15.36	366.54	19.15	#198	389,164	<b>8.60</b>	22.37	787.37	28.06
III	#6	188,930	13.77	14.80	280.65	16.75	#8	183,456	<b>13.24</b>	14.28	331.36	18.20
IV	#2	101,132	19.97	11.59	203.42	14.26	#21	110,956	<b>14.32</b>	8.79	121.32	11.01
V	#1	79,267	18.16	9.15	126.25	11.24	#15	90,311	<b>14.77</b>	7.51	89.98	9.49

**Table 9** (continued)

4 week-ahead												
Cluster Size	ST-OPTICS						CHD					
	Cluster ID	Num. of crimes	MAPE (%)	MAE	MSE	RMSE	Cluster ID	Num. of crimes	MAPE (%)	MAE	MSE	RMSE
I	#0	800,243	<b>6.89</b>	32.67	1715.69	41.42	#197	613,759	7.34	26.00	1086.59	32.96
II	#4	220,115	11.34	15.41	371.94	19.29	#198	389,164	<b>8.60</b>	22.34	788.77	28.09
III	#6	188,930	13.83	15.15	283.46	16.84	#8	183,456	<b>13.37</b>	14.55	341.07	18.47
IV	#2	101,132	20.52	11.81	209.45	14.47	#21	110,956	<b>14.51</b>	8.93	125.68	11.21
V	#1	79,267	18.18	9.17	126.81	11.26	#15	90,311	<b>14.85</b>	7.55	91.53	9.57

The best results are reported in bold



**Fig. 15** MAPE achieved by SARIMA model, for the top 5 most numerous clusters detected by ST-OPTICS and CHD

comparing MAPE, MAE, MSE and RMSE, we can observe that forecasts achieve generally lower errors on the hotspots detected by CHD than on those ones detected by ST-OPTICS. This result, in part due to the lower numerosity of the clusters, shows higher forecasting accuracy on the hotspots detected by CHD. As a more complete view of the MAPE results, Fig. 15 shows the MAPE versus several weekly forecasting horizons. From the plot, we can observe that percentage errors are lower on CHD-detected hotspots than on ST-OPTICS-detected hotspots (except for the largest cluster).

**Comparison with other crime forecasting approaches on the Chicago Crimes dataset**

With the aim of making a comparative analysis for crime forecasting more accurate and complete, we report here some comparative results between MD-CrimePredictor and some other approaches selected from the crime forecasting literature (i.e., [21–23]). Specifically, to ensure a fair and consistent comparison, we selected four algorithms that have been specifically applied to the Chicago crime data, i.e., the same dataset we exploited to evaluate MD-CrimePredictor as well. The approaches have been compared in terms of MAPE, which is a scale-independent metric (making it suitable for comparisons between different datasets or models) largely used in the crime forecasting

**Table 10** Comparative results on crime forecasting with other approaches proposed in literature on the Chicago crimes dataset, for one day-ahead forecasts

<b>Id</b>	<b>References</b>	<b>Model</b>	<b>Training period</b>	<b>Forecasting period</b>	<b>#days</b>	<b>MAPE</b>
#1	Reference [21]	Long-Short Term Memory and Spatial-Temporal Graph Convolutional Network	Jan 2015–Dec 2019	Jan 2020–Mar 2020	69	0.39
#2	Reference [22]	Spatial-Temporal Convolution Encoder	Jan 2016–Aug 2017	Sep 2017–Dec 2017	105	0.47
#3	Reference [23]	Dense Convolutional Network with Unsupervised Domain Adaptation (no feature construction)	Jan 2015–Jun 2015	Jul 2015–Dec 2015	184	0.23
#4	Reference [23]	Dense Convolutional Network with Unsupervised Domain Adaptation	Jan 2015–Jun 2015	Jul 2015–Dec 2015	184	0.14
#5	MD-CrimePredictor	CHD and Seasonal Auto-Regressive Integrated Moving Average (SARIMA)	Jan 2001–Dec 2018	Jan 2019–Dec 2019	365	0.12

performance evaluation [1]. Table 10 summarizes the results of the comparison, showing for each approach (*i*) the exploited models, (*ii*) the period of the Chicago crimes dataset exploited as training set, (*iii*) the period of the dataset exploited as test set, (*iv*) the total number of forecasted days, and (*v*) the related MAPE index for one-day-ahead forecasts, as reported in the corresponding references [21–23] (reviewed in Sect. "Related work"). By observing the table, it is worth noting that the MD-Crime-Predictor has been tested considering the longer time horizon (365 days), while the other approaches have been tested on time horizons no longer than 6 months (184 days for the approaches proposed in [23]). As a second thought, it can be seen that MD-CrimePredictor over-performs the other methodologies w.r.t. the MAPE index (0.12), resulting slightly more effective than the second best result reported in the table (0.14). The comparison confirms the goodness of the presented approach, even when considering short (one-day-ahead) time windows.

### Conclusion

This paper presented the design and implementation of MD-CrimePredictor (Multi-Density Crime Predictor), an approach based on multi-density clustering and regressive models to automatically detect high-risk crime areas in urban environments, and to reliably forecast crime trends in each area. First, the algorithm detects multi-density crime hotspots by applying a multi-density clustering algorithm, where densities, shapes, and the number of the detected regions are automatically computed by the algorithm without any pre-fixed division in areas. Then, a specific regressive model is discovered from each detected hotspot, analyzing the partitions discovered during the previous step. The final result of the algorithm is a spatio-temporal crime forecasting model, composed of a set of crime hotspots, their densities, and a set of associated crime predictors. Forecasting models are extracted by exploiting both SARIMA and LSTM models, and a comparative experimental analysis is presented in terms of error measures. The experimental evaluation of the proposed approach, performed on a large area of Chicago (involving

more than two million crime events), has shown higher accuracy of the first method with respect to the second one. We also offer a comparative evaluation of CHD in contrast to ST-OPTICS, making a comparison regarding crime prediction accuracy between hotspots identified through CHD and those identified through ST-OPTICS. Moreover, we have also presented a comparative analysis with other crime forecasting methods proposed in the literature, and specifically tested on Chicago crime data. Overall, the results show the effectiveness of the approach proposed in the paper, by achieving good accuracy in spatial and temporal crime forecasting over rolling time horizons.

In future work, other research issues may be investigated. First, we further explore the application of other multi-density approaches for the detection of crime hotspots, with the aim to perform a comparative evaluation between different clustering algorithms (multi-density vs classic density-based approaches) in crime spatial analysis. Second, we will study how other urban events can affect crime trends, and how such data can be correlated to criminal activities.

#### **Acknowledgements**

Not applicable.

#### **Author contributions**

EC and AV designed the study; PL and AV carried out data collection; EC, PL and AV carried out the analysis and interpretation of the results; EC, PL, and AV helped to write the manuscript; EC coordinated the whole research study and paper submission. All authors read and approved the final manuscript.

#### **Funding**

This research has been supported by the "PNRR MUR project PE0000013-FAIR", the "ICSC National Centre for HPC, Big Data and Quantum Computing" (CN0000013) within the NextGenerationEU program, and the European Union—NextGenerationEU - National Recovery and Resilience Plan (Piano Nazionale di Ripresa e Resilienza, PNRR) - Project: "SoBigData.it—Strengthening the Italian RI for Social Mining and Big Data Analytics"—Prot. IR0000013 - Avviso n. 3264 del 28/12/2021, and by the Italian Ministry of University and Research, PRIN 2022 "INSIDER: INtelligent Service Deployment for advanced cloud-Edge integRation", grant n. 2022WWSCRR, CUP H53D23003670006.

#### **Availability of data and materials**

The analyzed dataset is available at <https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present-Dashboard/5cd6-ry5g>.

#### **Declarations**

##### **Ethics approval and consent to participate**

Not applicable.

##### **Consent for publication**

Not applicable.

##### **Competing interests**

The authors declare that they have no competing of interests.

Received: 6 May 2023 Accepted: 7 May 2024

Published online: 17 May 2024

#### **References**

1. Butt UM, Letchmunan S, Hassan FH, Ali M, Baqir A, Sherazi HHR. Spatio-temporal crime hotspot detection and prediction: a systematic literature review. *IEEE Access*. 2020;8:166553–74.
2. Zhu Q, Zhang F, Liu S, Li Y. An anticrime information support system design: application of k-means-VMD-BiGRU in the city of Chicago. *Inf Manag*. 2022;59(5): 103247. <https://doi.org/10.1016/j.im.2019.103247>.
3. Zhu Q, Zhang F, Liu S, Wang L, Wang S. Static or dynamic? characterize and forecast the evolution of urban crime distribution. *Expert Syst Appl*. 2022;190: 116115.
4. Cesario E. Big data analysis for smart city applications. In: Sakr S, Zomaya AY, editors. *Encyclopedia of big data technologies*. Berlin: Springer; 2019.
5. Cesario E, Uchubilo PI, Vinci A, Zhu X. Multi-density urban hotspots detection in smart cities: a data-driven approach and experiments. *Pervasive Mob Comput*. 2022;86: 101687.
6. Law J, Quick M, Chan PW. Analyzing hotspots of crime using a Bayesian spatiotemporal modeling approach: a case study of violent crime in the greater Toronto area. *Geogr Anal*. 2015;47:1–19.

7. Catlett C, Cesario E, Talia D, Vinci A. Spatio-temporal crime predictions in smart cities: a data-driven approach and experiments. *Pervasive Mob Comput.* 2019;53:62–74.
8. Liu P, Zhou D, Wu N. VDBSCAN: varied density based spatial clustering of applications with noise. In: 2007 International Conference on Service Systems and Service Management, IEEE. 2007. p. 1–4.
9. Cesario E, Talia D. Distributed data mining patterns and services: an architecture and experiments. *Concurr Comput Pract Exp.* 2012;24(15):1751–74.
10. Mitra S, Nandy J. KDDClus: a simple method for multi-density clustering. In: Proceedings of International Workshop on Soft Computing Applications and Knowledge Discovery (SCAKD 2011), Moscow, Russia. Citeseer. 2011. p. 72–6.
11. Canino MP, Cesario E, Vinci A, Zarin S. Epidemic forecasting based on mobility patterns: an approach and experimental evaluation on COVID-19 data. *Soc Netw Anal Min.* 2022;12(1):116.
12. Amini A, Saboohi H, Wah TY. A multi density-based clustering algorithm for data stream with noise. In: 2013 IEEE 13th International Conference on Data Mining Workshops, 2013. p. 1105–12.
13. Amini A, Saboohi H, Herawan T, Wah TY. Mudi-stream: a multi density clustering algorithm for evolving data stream. *J Netw Comput Appl.* 2016;59:370–85.
14. Ester M, Kriegel H-P, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. *Kdd.* 1996;96:226–31.
15. Pankratz A. Forecasting with univariate Box-Jenkins models: concepts and cases. Hoboken: John Wiley & Sons; 2009.
16. Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput.* 1997;9(8):1735–80.
17. Tayebi MA, Ester M, Glasser U, Brantingham PL. Crimetracer: activity space based crime location prediction. In: 2014 IEEE/ACM International Conference On Advances in Social Networks Analysis and Mining (ASONAM), 2014. p. 472–80.
18. Chen H, Chung W, Xu JJ, Wang G, Qin Y, Chau M. Crime data mining: a general framework and some examples. *Computer.* 2004;37(4):50–6.
19. Liang W, Wu Z, Li Z, Ge Y. Crimetensor: fine-scale crime prediction via tensor learning with spatiotemporal consistency. *ACM Trans Intell Syst Technol.* 2022;13(2):33–13324.
20. Wang H, Kifer D, Graif C, Li Z. Crime Rate Inference with Big Data. In: Proceedings of the 22nd International Conference on Knowledge Discovery and Data Mining, ACM. 2016. p. 635–44. <https://doi.org/10.1145/2939672.2939736>.
21. Han X, Hu X, Wu H, Shen B, Wu J. Risk prediction of theft crimes in urban communities: an integrated model of LSTM and ST-GCN. *IEEE Access.* 2020;8:217222–30. <https://doi.org/10.1109/ACCESS.2020.3041924>.
22. Li Z, Huang C, Xia L, Xu Y, Pei J. Spatial-temporal hypergraph self-supervised learning for crime prediction. In: 2022 IEEE 38th International Conference on Data Engineering (ICDE). IEEE, 2022. <https://doi.org/10.1109/icde53745.2022.00269>.
23. Zhou B, Chen L, Zhao S, Li S, Zheng Z, Pan G. Unsupervised domain adaptation for crime risk prediction across cities. *IEEE Trans Comput Soc Syst.* 2023;10(6):3217–27. <https://doi.org/10.1109/TCSS.2022.3207987>.
24. Safat W, Asghar S, Gillani SA. Empirical analysis for crime prediction and forecasting using machine learning and deep learning techniques. *IEEE Access.* 2021;9:70080–94.
25. Cesario E, Lindia P, Vinci A. Detecting multi-density urban hotspots in a smart city: approaches, challenges and applications. *Big Data Cognit Comput.* 2023. <https://doi.org/10.3390/bdcc7010029>.
26. Nanni M, Pedreschi D. Time-focused clustering of trajectories of moving objects. *J Intell Inf Syst.* 2006;27:267–89.
27. Agrawal K, Garg S, Sharma S, Patel P. Development and validation of optics based spatio-temporal clustering technique. *Inf Sci.* 2016;369:388–401.
28. Rousseeuw PJ. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J Comput Appl Math.* 1987;20:53–65.
29. Moulavi D, Jaskowiak PA, Campello RJGB, Zimek A, Sander J. Density-based clustering validation, p. 839–47.
30. Halkidi M, Vazirgiannis M. A density-based cluster validity approach using multi-representatives. *Pattern Recognit Lett.* 2008;29(6):773–86.
31. Calinski T, Harabasz J. A dendrite method for cluster analysis. *Commun Stat.* 1974;3(1):1–27.
32. Davies DL, Bouldin DW. A cluster separation measure. *IEEE Trans Pattern Anal Mach Intell PAMI.* 1979;1(2):224–7.
33. Hyndman RJ, Athanasopoulos G. Forecasting: principles and practice. Melbourne: OTexts.com; 2014.
34. Shumway RH, Stoffer DS. Time series analysis and its applications: with R examples. Springer Texts in Statistics. 3rd ed. New York: Springer; 2011.
35. Cowpertwait PSP, Metcalfe AV. Introductory time series with R. 1st ed. Berlin: Springer; 2009.
36. Cryer JD, Chan KS. Time series analysis: with applications in R. Springer Texts in Statistics. Berlin: Springer; 2008.
37. Gers FA, Schmidhuber J, Cummins F. Learning to forget: continual prediction with LSTM. *Neural Comput.* 2000;12(10):2451–71.
38. Cilimkovic M. Neural networks and back propagation algorithm. Institute of Technology Blanchardstown, Blanchardstown Road North Dublin. 2015;15(1)
39. Liu Y, Li Z, Xiong H, Gao X, Wu J. Understanding of internal clustering validation measures. In: 2010 IEEE International Conference on Data Mining, 2010. p. 911–6.
40. Lederer J. Activation functions in artificial neural networks: a systematic overview. *CoRR.* 2021;abs/2101.09957.
41. Wilson DR, Martinez TR. The need for small learning rates on large problems. In: IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No.01CH37222), 2001;1:115–1191.
42. Cakmak E, Plank M, Calovi DS, Jordan A, Keim D. Spatio-temporal clustering benchmark for collective animal behavior. In: Proceedings of the 1st ACM SIGSPATIAL International Workshop on Animal Movement Ecology and Human Mobility, 2021. p. 5–8.

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.