Examining Spatio-Temporal correlations for Crime Prediction in New York City

Erika Nyström
nyerika@student.chalmers.se
Chalmers University of Technology (guest at Leiden University)
Gothenburg, Sweden

Ernest Vanmosuinck
e.r.o.vanmosuinck@umail.leidenuniv.nl
Leiden University
Leiden, Netherlands

Abstract

With increasing accessibility to spatio-temporal correlations in urban data, it is possible to extend previous research methods in the urban context. By collecting and utilising this information, novel improvements to public security and urban computing techniques can be examined. The utilisation of spatio-temporal data for improved crime prediction has previously been examined on a framework created purposefully for that task, but few attempts have been made with freely available open source Machine Learning models for the same task. By incorporating spatio-temporal correlations in a simple way to the examined dataset, some open source Machine Learning models performed really well on the crime prediction task. We concluded that weekly correlation had a higher impact on the performance of the predicting models than the daily correlation, showing a slightly improved score. We have also discovered that the spatial correlation had a higher affect on performance than the temporal correlation in regards of predicting crimes in New York City.

Keywords: Crime Prediction, Spatio-Temporal Correlation, Supervised Learning, Machine Learning

1 Introduction

With a majority of the human population living in urban areas around the word, improved and effective public security is essential. Crimes are highly affecting society and the ability to predict it entails large benefits for urban life. Crime prediction is related to sustainable urban development and citizen's life quality [19]. Crime and neighbourhood disorder have also been shown to correlate negatively to the health of urban residents [2], making the task of crime prediction a focus area in order to reduce the crime rate in a city. Some of the benefits of better understanding crimes includes targeted and more sensitive practices by the law enforcement in a city, making them able to mitigate the crime to a higher extent [1].

A lot of previous research regarding crime prediction are based on demographic data, but a major drawback with it is that it is difficult to achieve high "granularity" in communities since demographic features are relatively stable over an extended period and shared among several communities [19]. With increased collection of big data in an urban context, new sources and new models are applicable and examined to help improve the task of crime prediction [1]. Beyond urban data sources and demographic data, it has been shown that considering both time and location when predicting the crime rate has a positive impact on the prediction accuracy [19]. Moreover, the usage of Machine Learning techniques to predict crime events have increased significantly with recent model and accuracy developments [2]. Attempts to utilise spatio-temporal data and Machine Learning for crime prediction tasks have been investigated on other large cities in USA, such as Neural Network regression in Baltimore [2] and a comparative research of Naive Bayes, Support Vector Machines, Gradient Boosted Decision Trees and Random Forests for crime prediction and classification in San Francisco [1]. Another research paper was modelling spatio-temporal correlations by proposing a brand new model to predict crime rates in New York city [19], which is most inspiration has come from. Instead of using their proposed model (so called the TCP framework), our aim is to try out different open-source supervised Machine Learning techniques offered by scikit-learn [15] to answer our first research question on similar settings:

> *How well does an open source accessible supervised ML model perform on a crime prediction task of a year in New York City while incorporating spatio-temporal correlations?*

The models investigated for this task will be ensemble method Gradient Tree Boosting Regressor [8], Support Vector Machine Regression [14], Neural Network Regression [11], Random Forest Regression [13], Nearest Neighbours Regression [10] and Lasso [9]. These models will be compared on the same examined dates and location as the TCP framework [19] with some of the same data-sources, further explained in section 2. Once the optimal model have been selected, we will additionally investigate how much importance the different data-sources have on the crime prediction. The paper *Modelling Temporal-Spatial Correlations for Crime Prediction*, hereafter referred to the "baseline paper", proposes the TCP framework [19] and use several data sources from a variety of different fields and no extensive feature selection analysis. That proposes a small investigation of the used data sources and their contribution towards the result for the task of crime prediction, which our second research question aims to answer:

> *How does the chosen features affect the aRMSE of the crime prediction task?*

After the upcoming section of previous work, the data gathering and preprocessing will be further explained in section 2. In section 3 the problem is conceptualised and modelled, followed by an explanation of the chosen Machine Learning models for the task. Section 4 presents the results and section 5 will cover a discussion over the results and suggestions for further work. The paper ends with the conclusions and acknowledgements in section 6.

1.1 Previous work

The baseline paper *Modelling Temporal-Spatial Correlations for Crime Prediction* is the primary source of inspiration for this project [19]. The proposed framework TCP is implemented as follows: divide New York City into regions and assigning the features in the data to the different regions for a specific time slot. Then create a weight matrix to incorporate the spatio-temporal correlations and train the matrix on the previous time slots.

The features used to train the model consists of data from public security sources (2 pcs), meteorological sources (30 pcs), Points of interest in that region, human mobility (2 pcs) and public service complaints. The paper examines the intra-region temporal correlations; i.e. how much the crime rate differs between different time slots in a specific region, as well as the inter-region spatial correlation; how much spatial closeness affect similarity in crime rate for a given time slot. The framework was created to incorporate these spatial-temporal correlations and motivated by comparing *average-Mean-Root-Squared-Error* (aRMSE), partially displayed below in table 1, to other representative baselines models. All techniques performed better for short-term prediction, but the TCP framework outperforms all the other baseline models used. TCP is also more robust for distant future predictions compared to the other models and the conclusion is that spatio-temporal correlations can help crime prediction [19].

Unfortunately, neither the data used in the paper nor the code implementing the TCP framework are available online for full comparison.

Table 1. Performance in terms of aRMSE by [19]

	1-day	7-day
Lasso	2.8210	3.3956
TCP	**1.7205**	**1.7791**

Another research project examined the use of deep learning models, namely Convolutional Long-Short Term Memory Neural Network (CLSTM-NN), to model spatio-temporal correlations for the task of crime prediction [2]. The data was gathered only from one data source, covering for e.g. the type of crime, the date and time of its occurrence as well as its longitude and latitude. Similarly to the baseline paper examining New York City, the model predicts the crime rate based on features in the format of matrices, where each element of the matrices represent the number of crimes within a region defined by its location. The model captured temporal correlations by adding the feature matrices of several days back, concluding the best accuracy was obtained by not considering more than 5 days in advance. The use of past 7-days lead to a slight overfitting of the model for some of the parameters in the neural network tested. Once again, one of the conclusions were that spatio-temporal resolutions are relevant in the performance of the model [2].

Additionally, the task of predicting crime classes have been examined with spatio-temporal as well as demographic data in the city of San Francisco [1]. The model aims to classify the category of crime based on the time, place and demographic data. The classification models tested for this task was Naive Bayes, Random Forests, Support Vector Machines and Gradient Boosted Decision Trees. By collapsing some the crime categories into a fewer number of categories, Gradient Boosted trees and Support Vector Machines had the highest accuracy. The Gradient boosted trees achieved an maximum accuracy of 96% and 75% for two different examined classes and the support vector machine obtained 96% and 62% [1].This paper did however not use a grid to represent any inter-regional or intra-temporal correlations like the previous two projects.

2 Data

In order to model comparability with the baseline paper *Modelling Temporal-Spatial Correlations for Crime Prediction*, we have decided to use some of the data sources that they found. The data sources are:

- **311 Public-Service Complaint Source**: This dataset contains the non-urgent complaints made by the public to the city, using the 311 phone number [4]. This data is available for free on the city of New York website.
- **Public Security Data**: This will be in the form of New York Police Department (NYPD) Stop and Frisk (SAF) reports filled by the department. This data is submitted to the public by the city of New York, and is filled by agents after they have performed a light body search on an individual (frisk) [5]. We will also be using the NYPD Complaint Data Historic. This dataset includes all valid felony, misdemeanour, and violation crimes reported to the NYPD [6].
  Both of these sources are available for free on the city of New York website.
- **Human Mobility**: This data will be in the format of Taxi Pick-up and Drop-off points. This data is made

available by the NYC Taxi and Limousine Commission [17]. This data incorporates pick-up/drop-off points for taxi trips, and includes the trip distances, itemized fares, rate types, payment types, and driver-reported passenger counts. For this project, we will only be using the passenger count and trip fares.

Following the structure of the baseline paper, we have decided to look into the correlation of crime and region on a specific time-frame, selecting the start date as the 1st of July 2012, and the end date as the 30th of June 2013. This gives us a total of 365 days. This will cover the temporal factor. When it comes to the spatial factor, the baseline paper divided New York into $2km \times 2km$, and achieved a total of 133 regions. After some personal experimentation, we reached a total of 254 regions following the same operation. Furthermore, we decided to only use the regions that contained data for the NYPD Complaints, as this sources will be used as label value. This means that in the end, we have a total of 254 regions. This can be observed in figure 1, where the greyed cells will be ignored and the cells covering NYC will be used.

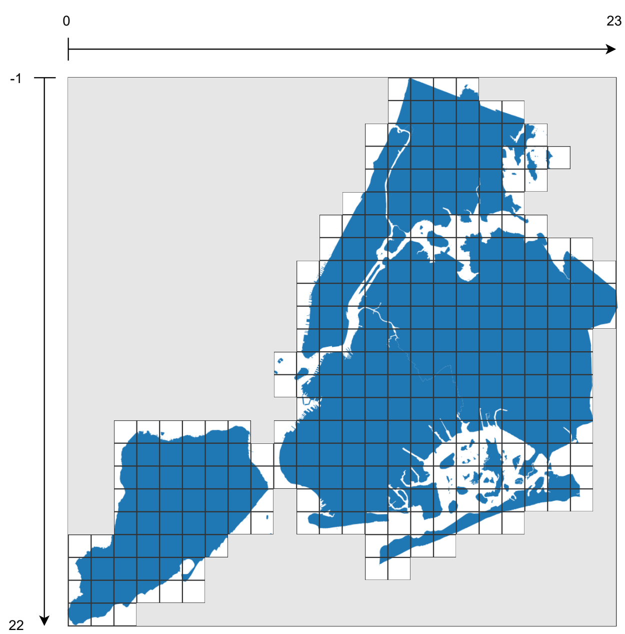

Figure 1. New York City divided in $2km \times 2km$ regions.

2.1 Data cleansing

Like previously mentioned, the baseline paper [19] only referred to their data sources. This meant that we had to fetch and pre-process the data ourselves, as it would be unusable without some prior pre-processing. This statement is further justified by the size of some of our data sources, namely the taxi data, which contained an average of 15,000,000 rows per month. Firstly, the data would need to be filtered for the desired time-range.

Secondly, all datasets used coordinate points but they did not all use the same format. Some datasets used the State Plane Coordinates System (SPCS) format. This format, which is proper to the United States [16], represents coordinates in the form of [x,y] values. The 311 complaints, stop and frisk reports and NYPD complaints datasets were all in this described format. The taxi data on the other hand used a longitude-latitude format. As the SPCS format was predominant across our data sources, we decided to apply this format to all other sources. This operation was performed using the Python library `pyproj`, which enables the projection of longitude/latitude to a specified SPCS plane by using the EPSG Geodetic Parameter Dataset [3]'s codes for World Coordinates (EPSG:4326) and New York Long Island (EPSG:2263). This process was rather long, as the taxi dataset was very large and contained a total average of 175,000,000 rows. Once the coordinates were modified to the right format, we were then able to index them using a modulo operation to place them on a grid of NYC. Indexing the entries with this method ensures that an entry is in the correct range [-1 to 22] for the y axis, and [0 to 23] for the x axis.

2.2 Data Grouping and Merging

Once all the datasets were processed and formatted, we were able to group them by date and coordinates. For the 311 complaints, the stop and frisk reports and the NYPD complaints, the entries can be counted to represent a number of complaints or reports for a specific day at a certain coordinate point. As mentioned in the data sources (section 2), we wanted to exploit the passenger count and the trip fare from the taxi dataset. Instead of counting the entries, we summed the values for passenger count and trip fares based on date and coordinates. We separated the pick-up entries and drop-off entries, as they could indicate different information in terms of spatial factors.
After the grouping of each datasets, we were easily able to merge them on their shared axes (date and coordinate).

3 Proposed Model

The data was formatted in a similar fashion to the baseline paper and incorporating the spatio-temporal correlations was adjusted accordingly. Once this have been more extensively explained in section 3.1, the different Machine Learning models picked for the regression task will be further explained and motivated in section 3.2. Finally, this section will be concluded with the measures taken to prevent overfitting of the examined models.

3.1 Problem formulation

New York City is divided into R=254 regions, and the investigated time-span of 1 year is divided into T=365 time slots. $\mathbf{Y}_r^t$ denotes the number of crimes in region r at time t, and $\mathbf{X}^t = [\mathbf{X}_1^t, \mathbf{X}_2^t, ..., \mathbf{X}_r^t]$ is the feature matrix of all regions in time slot t.

In order to capture the spatial correlations, the model will consider the neighbours information $(\mathbf{X}^t, \mathbf{Y}^t)$ denoted $\mathbf{N}_j^t$ where $j \in N, W, E, S$ (direction of neighbour; North, West, East, South, and ignore the diagonal neighbours) as additional features to model the crime rate $\mathbf{Y}_r^t$:

$$f_{spatial}(\mathbf{X}_r^t, N_N^t, N_W^t, N_E^t, N_S^t) = \mathbf{Y}_r^t \quad (1)$$

The model will also incorporate the temporal correlations by considering the l=lag previous days information $(\mathbf{X}_r^{t-l}, \mathbf{Y}_r^{t-l})$ features and the crime rate as additional features to model the crime rate $\mathbf{Y}_r^t$:

$$f_{temporal}(\mathbf{X}_r^t, \mathbf{X}_r^{t-l}, \mathbf{Y}_r^{t-l}) = \mathbf{Y}_r^t \quad (2)$$

Making the final model incorporating both the spatial and temporal correlations:

$$f(\mathbf{X}_r^t, N_N^t, N_W^t, N_E^t, N_S^t, \mathbf{X}_r^{t-l}, \mathbf{Y}_r^{t-l}) = \mathbf{Y}_r^t \quad (3)$$

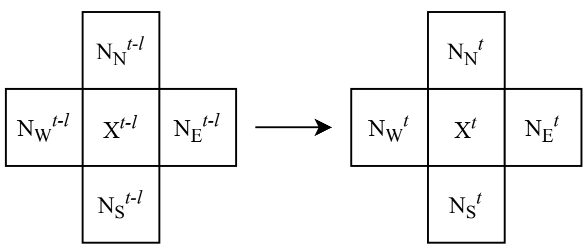

Figure 2. Visualisation of model incorporation of the spatio-temporal correlations

3.2 Model selection

Several "baseline" open source supervised machine learning models from scikit-learn [15] were tested with the default parameters. The regression versions of the aforementioned classification models for crime prediction in San Francisco [1] were all tested (excluding Naive Bayes as it is principally used as a classification model). A simple Neural Network was also tested based on the deep neural network model suggested for a similar crime prediction task in Baltimore [2].

3.3 Model Tuning

Initially, all features need to be numerical in order for the regression models to be able to compute their results. The dates were converted into UNIX timestamps, and the indexed regions were separated into two features; one feature covering the x-axis and another one covering the y-axis.

Standardizing the data during preprocessing is also often necessary for some machine learning models [12]. Hence, all models were tested in a pipeline first applying a standardizer from scikit-learn [12]. The first one, `StandardScaler`, was chosen based on its simplicity and high performance. Another data preprocessor `Normalizer` available from scikit-learn was also tested to examine the need for a scaling in this project. During the tuning of the models, very low aRMSE values were found for some of the models despite no hyperparameter tuning. The first tuning attempt was made with only a `train_test_split` [7], providing the results presented in table 3 in section 4. The result was further cross-validated by applying a `ShuffleSplit` [7] consisting of 5 independent splits of testing and training data for some of the models, presented in table 4.

3.4 Model Evaluation

The metric used to evaluate the models is the average-root-mean-squared-error, referred here as aRMSE, similar to the baseline paper[19]. This metric is commonly used to evaluate regression models. For baseline evaluation, a "dummy" regressor has been created and compared to the other models. This "dummy" regressor simply takes the number of NYPD crime complaints from the previous day as its prediction (or seven days prior, for the seven-days lag dummy).

4 Results

Firstly, we applied the scaling methods mentioned in section 3.3 to the data with a 1-day lag to examine potential benefits. Those results can be viewed in table 2. Looking at the aRMSE values returned by those models, we can see that using a pipeline did not greatly improve the results of the GradientBoostingRegressor, nor the results of the RandomForestRegressor. Applying the `StandardScaler` to those models did not have a significant impact on the aRMSE score, and is not great enough to motivate using a pipeline as it removes the

Table 2. Performance in terms of aRMSE for some models with different preprocessing steps for 1-day lag

Model	No pipeline	StandardScaler	Normalizer
`GradientBoostingRegressor`	0.0020	0.0020	43.694
`RandomForestRegressor`	0.0105	0.0065	29.9667
`SVR`	76.5317	2.1776	76.5275
`Lasso`	0.0249*	1.0548	69.8160
`KNeighbourRegressor`	39.2322	3.8080	20.5271
`MLPRegressor`	4.0e11	0.7294	50.7594

possibility of comparing the performance with the "dummy" regressor, explained in section 3.4. The Lasso model was also in the best performing model, but during our experimentation, Lasso raised a `ConvergenceWarning`, which signals that the model did not converge. Thus it would be misleading to consider this model among the best-performing.

By running the selected models on the training data obtained from `train_test_split`, the following result were obtained for the two different examined lags:

Table 3. Performance in terms of aRMSE for tested models

Model	1-day	7-day
GradientBoostingRegressor	**0.0020**	**0.0013**
RandomForestRegressor	0.0105	0.0052
SVR	76.5317	72.3758
Lasso**	0.0249	0.0246
KNeighboursRegression	39.2322	39.5818
MLPRegressor	4.0e11	3.5e11
dummy Regressor	19.3845	20.0013

From the baseline paper [19], the TCP framework outperformed the LASSO model significantly with aRMSE-values of 1.72 for 1-day lag and 1.78 for 7-day lag in the baseline paper as shown in table 1 above. Since it is not clear if they used a pipeline for their models or not, it is impossible to compare their results of Lasso to ours. The results they obtained are however lower than two of our tested models in table 3; GradientBoostingRegressor and RandomForestRegressor, despite no hyperparameter tuning. It does need to be specified that our attempt of capturing the spatio-temporal relationship of the data differs a bit from the baseline paper [19], as we aimed for simpler framework and a smaller amount number of features.

To minimise the risk of overfitting the models, the highest performing models were also tested with a `ShuffleSplit` cross-validator for 1-day lag, providing similar results in table 4 below. The differences between tables 3 and 4 are very small.

Table 4. Performance in terms of aRMSE for some models with ShuffleSplit for 1-day lag

Model	aRMSE
GradientBoostingRegressor	**0.0123**
RandomForestRegressor	0.00318

To enable a feature analysis with the optimal model; GradientBoostinRegressor, a correlation matrix covering a selected amount of features was calculated in figure 3 below. Visualising all 45 features included in the data would not be feasible,

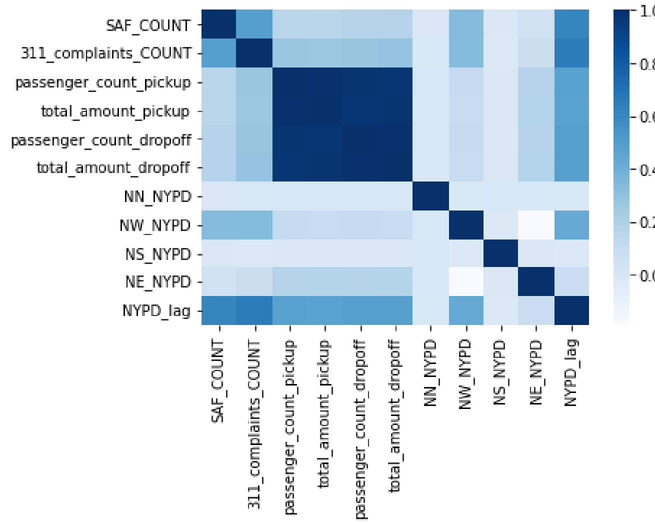

Figure 3. Visualisation of feature correlations

so the features for neighbours and lag except for their actual NYPD value (the label) was ignored.
The correlation matrix in figure 3 displays that the number of stop-and-frisks (SAF), 311 complaints and the lag NYPD complaints seem to highly seem to correlate to each other the most. In order to investigate how important the different features in our data are for the model to predict the crime rate, the best performing model according to table 3 above was also tested with a subsets of the features. The results are displayed in table 5 where all features containing any of the specified information, both for the examined day, its neighbours and the lag, are removed.

Table 5. Performance in terms of aRMSE for different features subtracted for GradientBoostingRegressor

Data subtraction	1-day	7-day	# features
all SAF features	0.0018	0.0014	39
all 311 features	0.0018	0.0014	39
all taxi features	0.0019	0.0011	22
all neighbour features	0.0021	0.0013	17
all lag features***	*0.0018*	*0.0013*	38

The difference in aRMSE between the tables 3 and 5 are minimal. Only one feature subtraction seem to decrease the accuracy, namely the neighbouring features. The number of features considered in the model, 17, are however significantly smaller than the original 45. Removing all lag features, in this model incorporating the temporal correlations, seem to reduce the aRMSE more. The difference here between 1-day and 7-day for all lag features (***) in table 5 stem from the different divisions used during `train_test_split`, where different indexes were chosen for training data for lag=1 and lag=7. Similar to the close correlation between stop-and-frisk

(SAF) and 311 displayed in figure 3, the difference in aRMSE between removing all SAF features or all 311 features, are negligible.
All results can be viewed on the project's GitHub repository [18].

5 Discussion

To answer our first research question, we can state that some open source accessible supervised Machine Learning models perform very well on the crime prediction task of New York City between 1st of July 2012 and 30th of June 2013. Some of the tested model in table 3, namely GradientBoostingRegressor and RandomForestRegressor, performed very well on the provided data consisting of public complaints, public security and human mobility data. These two outperformed the dummy predictor used for evaluation significantly, but other models tested in table 3 performed worse than the dummy predictor, implying that they are not successfully applicable on this task. The highest performing model, GradientBoostingRegressor, according to our result table 3, was also the model that performed the best for the classification task of crime prediction in San Francisco [1]. They were able to get an accuracy ranging between 70 and 90 % without incorporating any spatio-temporal correlations, suggesting that our results could feasible since our model have access to other kinds of information. However, an accuracy of around 99,99% typically imply overfitting the data or some other error. Two ways to reduce the risk of overfitting are to either adding more data, or to apply cross-validation. Adding more data concerning the examined time period and location was in this case not feasible due to the date and time range limitations. Some ways of cross-validation have been applied, with both `train_test_split` as well as `ShuffleSplit`, but no actual understanding regarding if the models are overfitted or not have been collected from it.

One reason for the very high performance of our models could be that the selected data sources were the optimal ones in terms of what features to consider and their correlation to the actual crime rate, and the remaining features in the baseline paper are merely noise making it harder for the models to predict with a high accuracy. The selection process of what data sources to use compared to the baseline paper [19] was based on what data was accessible online. However, since the data differs between the baseline paper and our implementation, it is difficult to assess the improvement we have managed to create in this task of crime prediction for New York City to the feature extraction process or the actual models themselves.

Moreover, both the baseline paper [19] as well as the research on crime prediction in Baltimore [2] stated that taking a too big lag would decrease the accuracy. Both papers suggested using a 5-day lag, compared to the lag of 7-day we used in our data. By looking at our results in 3, a longer lag outperformed the results of the shorter one. A 7-day lag correspond to the same day the week before, implying that temporal correlations might be stronger for a weekly period than a daily one. According to research on Baltimore [2], a 7-day lag led to a slight overfitting of the model, something that could be a reason for our startling results.

Regarding our second research question, most features used in our data seem to be important by comparing the tables 3 and 5. If the lag = 1-day, having both all the SAF and all 311 features seem superfluous, as the accuracy does not decrease and the features are highly correlated. If storage is an issue and/or limitation, removing all taxi features could be considered based on deduction of total number of features examined with the negligible increase in aRMSE. Finally, by removing the lag features, the aRMSE seem to decrease even further. This implies that the temporal correlations in this case could in fact be superfluous and becoming noise for the model.

5.1 Research suggestions

With the numerical form of our dataset, it would be simple to apply other models that were not tested in this project. Since the data used in the baseline paper is not available, we cannot state anything about the transferability of their data and how well other models would perform on their data. Therefore we propose further research to continue by attempting to test their data on the same models we chosen as well as testing the TCP framework on our data. By investigating this, it is possible to state if the performance improvement found in this project stems from a more optimal preprocessing and data selection, or the TCP framework merely being overcomplicated.

One other thing that was discovered during tuning of the model, was that applying a StandardScaler sometimes decreased the performance of the models, which seems counterintuitive. Another proposition is to implement and test other dummy regressors, such as for example taking the mean of some previous and upcoming days NYPD value as prediction for the examined day, or applying a probabilistic distribution, for e.g. the Gaussian distribution and see how well it would perform. Finally, it would be interesting to add more data into our models to see if that would increase or decrease the performance of the models, giving a direction of whether the data sources in the baseline paper [19] were valuable or increasing complexity and noise.

6 Conclusions

By incorporating spatio-temporal relationships to the task of predicting the crime rate of New York City, surprisingly good results can be expected using GradientBoostingRegressor and RandomForestRegressor. Some of the data used for this task; namely public complaints, public security and taxi data seem to be redundant in some cases for a shorter time lag if all other features remain. No extensive comparison to the baseline paper [19] is possible, making it hard to detect potential flaws with our preprocessing of the used data used. Based on the previously mentioned results, our contributions with this project are:

- Only taking the closest neighbouring values (North, West, South and East) and the values from the same region for a lag l is indeed a feasible way to incorporate spatio-temporal correlations into the model.
- GradientBoostingRegressor and RandomForestRegressor could be high performing open source Machine Learning models applicable on crime prediction tasks incorporating spatio-temporal data.
- The same weekday last week have a higher correlation to the crime rate of today compared to yesterday for this data of New York City.
- The spatial correlations does seem to be more important the temporal correlations considering the change in aRMSE. This change however, is still extremely small and an indication of an possibly overfitted model.

But like previously mentioned, the highest performing models of our results does have an abnormally high accuracy, implying that they could be prone to overfitting the data. This is strongly encouraged to examine further by applying similar data sources to the models for other tasks, i.e. other years of crime prediction in other cities.

Acknowledgments

We want to thank our supervisor Bram Renting who has been very valuable in our research and providing meaningful insights.

References

[1] Addarsh Chandrasekar, Abhilash Sunder Raj, and Poorna Kumar. 2015. Crime Prediction and Classification in San Francisco City.

[2] Nicolás Esquivel, Orietta Nicolis, Billy Peralta, and Jorge Mateu. 2020. Spatio-Temporal Prediction of Baltimore Crime Events Using CLSTM Neural Networks. *IEEE Access* 8 (2020), 209101–209112. https://doi.org/10.1109/ACCESS.2020.3036715

[3] MapTiler. [n. d.]. *Coordinate Systems Worldwide*. Retrieved December 21st, 2021 from https://epsg.io/

[4] City of New York. 2012,2013. *311 Service Request from 2010 to Present*. Retrieved November 30th, 2021 from https://data.cityofnewyork.us/Social-Services/311-Service-Requests-from-2010-to-Present/erm2-nwe9

[5] City of New York. 2012,2013. *The Stop, Question and Frisk Data*. Retrieved November 30th, 2021 from https://data.cityofnewyork.us/Public-Safety/The-Stop-Question-and-Frisk-Data/ftxv-d5ix

[6] City of New York. 2021. *NYPD Complaint Data Historic*. Retrieved November 30th, 2021 from https://data.cityofnewyork.us/Public-Safety/NYPD-Complaint-Data-Historic/qgea-i56i

[7] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. [n. d.]. *Cross-validation: evaluating estimator performance*. Retrieved January 10th, 2022 from https://scikit-learn.org/stable/modules/cross_validation.html?highlight=cross_validation

[8] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. [n. d.]. *Gradient Tree Boosting*. Retrieved December 21st, 2021 from https://scikit-learn.org/stable/modules/ensemble.html#gradient-tree-boosting

[9] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. [n. d.]. *Lasso*. Retrieved December 21st, 2021 from https://scikit-learn.org/stable/modules/linear_model.html#lasso

[10] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. [n. d.]. *Nearest Neighbors*. Retrieved December 21st, 2021 from https://scikit-learn.org/stable/modules/neighbors.html#nearest-neighbors-regression

[11] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. [n. d.]. *Neural network models (supervised)*. Retrieved December 21st, 2021 from https://scikit-learn.org/stable/modules/neural_networks_supervised.html#regression

[12] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. [n. d.]. *Preprocessing data*. Retrieved January 10th, 2022 from https://scikit-learn.org/stable/modules/preprocessing.html#standardization-or-mean-removal-and-variance-scaling

[13] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. [n. d.]. *Random Forests*. Retrieved December 21st, 2021 from https://scikit-learn.org/stable/modules/ensemble.html#random-forests

[14] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. [n. d.]. *Support Vector Machines*. Retrieved December 21st, 2021 from https://scikit-learn.org/stable/modules/svm.html#regression

[15] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research* 12, Oct (2011), 2825–2830.

[16] U.S Geological Survey. 2021. *What is the State Plane Coordinate System? Can GPS provide coordinates in these values?* Retrieved November 30th, 2021 from https://www.usgs.gov/faqs/what-state-plane-coordinate-system-can-gps-provide-coordinates-these-values?qt-news_science_products=0#qt-news_science_products

[17] Taxi and Limousine Commission. 2012,2013. *TLC Trip Record Data*. Retrieved November 30th, 2021 from https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page

[18] Ernest Vanmosuinck and Erika Nyström. 2021. *Spatio-Temporal Correlation on Crime Prediction in NYC*. https://github.com/ernestvmo/spatio-temporal-crime-prediction

[19] Xiangyu Zhao and Jiliang Tang. 2017. Modeling Temporal-Spatial Correlations for Crime Prediction. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management* (Singapore, Singapore) *(CIKM '17)*. Association for Computing Machinery, New York, NY, USA, 497–506. https://doi.org/10.1145/3132847.3133024