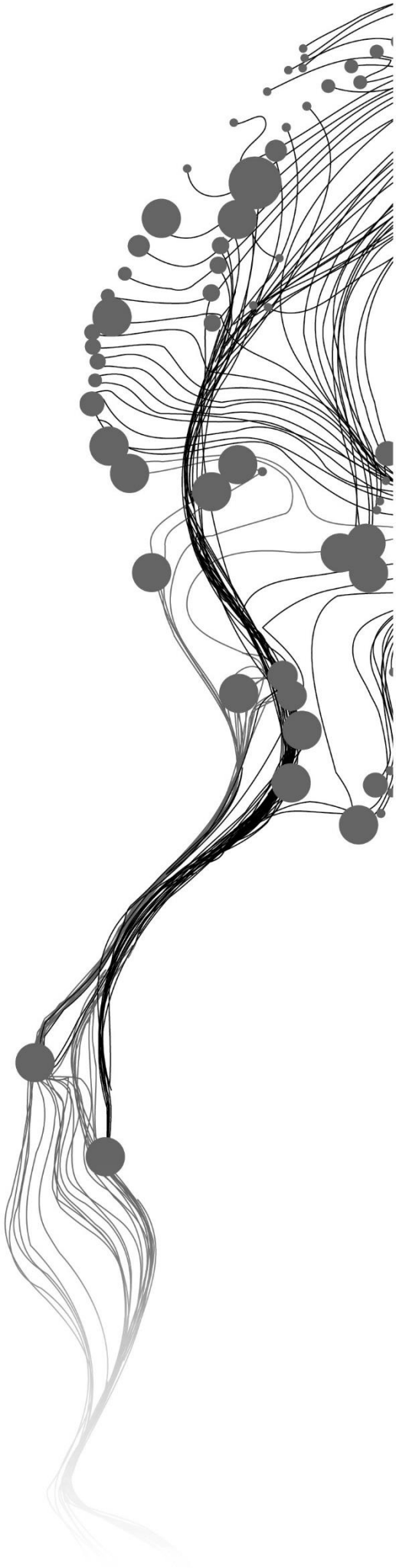
ON THE USE OF MIXED EFFECTS MACHINE LEARNING REGRESSION MODELS TO CAPTURE SPATIAL PATTERNS: A CASE STUDY ON CRIME

AFNINDAR FAKHRURROZI
FEBRUARY, 2019

SUPERVISORS:
Prof. Dr. Raul Zurita Milla
Dr. O. Kounadi



ON THE USE OF MIXED EFFECTS MACHINE LEARNING REGRESSION MODELS TO CAPTURE SPATIAL PATTERNS: A CASE STUDY ON CRIME

AFNINDAR FAKHRURROZI
Enschede, The Netherlands, February, 2019

Thesis submitted to the Faculty of Geo-Information Science and Earth Observation of the University of Twente in partial fulfilment of the requirements for the degree of Master of Science in Geo-information Science and Earth Observation.
Specialization: Geoinformatics

SUPERVISORS:
Prof. Dr. Raul Zurita Milla
Dr. O. Kounadi

THESIS ASSESSMENT BOARD:
Prof. Dr. M.J. Kraak
Dr. E. Izquierdo-Verdiguier (External Examiner, University of Natural Resources and Life Sciences, Vienna, Austria)

ABSTRACT

Machine learning regression models have recently gained popularity due to its ability to predict continuous outputs and reveal patterns from data. However, it is unclear whether these models perform well on spatial data due to frequently regression residuals are spatially autocorrelated, which violates the condition of independent and identically distributed. Recently, mixed effects machine learning regression model approach has been proposed to address the cluster in the data. In this research, we investigate the use of mixed effects machine learning regression models to capture spatial patterns. Random Forest (RF) regression, Support Vector Regression (SVR) and their mixed effects counterparts; namely Mixed Effects Random Forest (MERF) and Mixed Effects Support Vector Regression (MESVR) were chosen to develop models from spatiotemporal data. In this study, we analyse the performance using a real-world dataset to predict crimes in New York City. The model performance was evaluated with respect to the predictive power and the degree of spatial autocorrelation in the residuals. We conducted several experiments to evaluate the model performance using lagged spatial features; namely spatial lag and LISA's local moran quadrant, non-lagged spatial features and various combination of random and fixed effects features. Experimental results show that MERF outperforms the other models in the selected metrics. Additionally, MESVR also outperforms the SVR in terms of predictive models. We also observed that using lagged spatial features can reduce the spatial autocorrelation of regression residual and improve predictive performance. Therefore, we conclude that mixed effects machine learning regression models, in this case, MERF models can effectively learn from spatiotemporal data and can predict the continuous outputs accurately to reveal spatial patterns while keeping the spatial autocorrelation of the residuals low.

Keywords: machine learning, mixed effects models, spatial patterns, spatial autocorrelation, spatial features.

ACKNOWLEDGEMENTS

The first and the foremost I would like to thanks and praise to Allah SWT the Almighty for giving me a blessing, the strength and endurance for this study.

I would like to thanks to my supervisor, Prof. Dr. Raul Zurita Milla, for his supporting and valuable feedbacks on every discussion. I also would like to express my gratitude to my second supervisor, Dr. O. Kounadi for her constructive criticism to guide me in this MSc thesis.

My sincere thanks to my beloved parents, my dearest wife Anggun and my big family for always encourage, continuous pray and keep supporting me.

I would like to extend my deepest appreciation to RISETPro (Research and Innovation in Science and Technology Project) as part of World Bank project for giving me a chance to continue my study to higher level in ITC, University of Twente.

Many thanks to Research Center for Geotechnology, Indonesian Institute of Sciences (LIPI) as my institution and my colleague, Dr. Bambang Setiadi for giving me access to High-Performance Computing (HPC) grid at LIPI.

Last but not least, I would like to thanks to my Indonesian colleague, ITC_2017September.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1. MOTIVATION AND PROBLEM STATEMENTS

Recent developments in geospatial technologies have significantly improved the way we gather and access spatiotemporal data about humans and their environment (Kwan & Neutens, 2014). These technologies, such as GPS enabled smartphones, have sensors embedded and can record positions and movements. Besides smartphones, crowdsourced data from web applications, Twitter, Instagram and other social media platforms are good sources of geotagged information related to specific phenomena. The resulting proliferation of spatial data has remarkably influenced its complexity, dimension, and volume. However, it also provides opportunities for exploratory spatial research to reveals spatial patterns (Hagenauer & Helbich, 2013).

A spatial pattern in the distribution of a geographic phenomenon is defined by the arrangement of individual objects in two or three dimensions and their geographical relationships (Chou, 1995). The pattern itself may not be observed by eyes and need some statistical analysis to reveal and assured that the data correspond to it. Spatial Autocorrelation (SAC) analysis is one of the geographical techniques that can be used to capture the spatial pattern. SAC measures the degree of relationship between spatial entities in the neighbouring area (Chou, 1995; H. Wang, Guo, Liu, Liu, & Hong, 2013). In the statistical domain, Moran's I and Geary's C coefficient have been widely used to measure spatial autocorrelation (Chou, 1995).

The presence of SAC in the data might negatively affect classical regression model (Bertazzon, Johnson, Eccles, & Kaplan, 2015; Lichstein, Simons, Shriner, & Franzkreb, 2002; Santibanez, Lakes, & Kloft, 2015). SAC occurred when the dependent variable is autocorrelated; thus the assumption of independence is often violated (Lichstein et al., 2002). Moreover, SAC induces spatial autocorrelation of the residuals of the regression, which indicate that there are structural problems of the model (Y. Chen, 2016). The residuals of the model will likely to exhibit clustering or other patterns (Santibanez, Lakes, et al., 2015). The occurrence of these patterns in the residuals violates the assumption of statistical analysis that residuals are independent and identically distributed (Dormann et al., 2007). Normally, regression assumes that all residuals are taken from the population has constant variance and scattered randomly around zero. It indicates that there is a missing key of features or misspecification of the model that might lead the model to under or oversimplification (Esri, 2013; Chen, 2016). Apart from that, random noise in the data may induce spatial autocorrelation and may lead to misleading interference and resulting underestimating of the model (Rocha et al., 2018) Thus, these conditions will lead prediction model to be unreliable.

In the statistical domain, spatial autoregressive method has been proposed to handle SAC in order to develop inference model (Hua, Junfeng, Fubao, & Weiwei, 2016). However, regression analysis using statistical approach cannot cope with the variety, velocity, volume, and high dimension of large spatiotemporal dataset. This gives rise to the need for machine learning (Bzdok, Altman, & Krzywinski, 2018).

Machine learning is a branch of artificial intelligence and that is gaining popularity. It has been widely used in many applications. For instance, engineering, science, healthcare and business including earth science (Lary et al., 2018). It is often used by GIS practitioners in image processing and remote sensing applications (Lary et al., 2018). It allows the computer to learn from the data without being programmed

to perform specific tasks. Moreover, machine learnings techniques and methods can work with complex, high dimensional and tremendous amount of data and it can be used to develop regression and classification models.

Machine learning regression models are powerful to predict to reveal the patterns from big data (Bzdok et al., 2018). They are scalable, flexible and capable of splitting processes into smaller chunks which run simultaneously, i.e. parallelisation (Upadhyaya, 2013). However, these techniques pose a significant challenge to model spatial pattern as most machine learning regression are not intended to deal with spatial data (Santibanez, Lakes, et al., 2015). Moreover, excellent goodness of fit can be achieved when the data is highly clustered, and also the model might indicate to overfitting (Santibanez, Lakes, et al., 2015). In other words, when the density of the cluster (as each cluster has their own features and different with each other) in a data is high, the model trying to learn the detail from each cluster in the data as a concept but this concept can not be applied to new data. This situation negatively affected the ability of the model to generalize the learning. Apart from that, noise in the data can also induce the spatial pattern that might lead to overfitting (Rocha et al., 2018). A new approach to handle cluster in the data, mixed effects machine learning has been proposed (Cho, 2010; Hajjem, Bellavance, & Larocque, 2014; Luts, Molenberghs, Verbeke, Van Huffel, & Suykens, 2012; Seok, Shim, Cho, Noh, & Hwang, 2011).

Mixed effects models are well-suited for datasets that have cluster structure. Clustered data emerge when the datasets can be classified into many different groups (Galbraith, Daniel, & Vissel, 2010). Cluster structure can be longitudinal or hierarchical. Longitudinal structure arises when multiple observation measured within the same cluster, for instance, bare soil and forest land cover cluster. As for hierarchical cluster treating each observation into a separate cluster then merge the cluster that has similarity, for instance, deciduous forest landcover contained within forest landcover. Each cluster distinct from each other cluster. Mixed Effects Random Forest (MERF) approach showed significant improvements over vanilla random forest when random effects are substantials (Hajjem et al., 2014). Apart from that, mixed effects support vector regression (MESVR) using Least Square SVR (LS-SVR) for handling longitudinal data and highly unbalance data also has been proposed (Cho, 2010; Luts et al., 2012; Seok et al., 2011). However, it is noteworthy that MESVR approach library (code) for regression is unavailable.

Although several studies have been accomplished to reveal pattern using machine learning regression models in many disciplines (Wang et al., 2010; Kong et al., 2016; Czernecki et al., 2018; Schug et al., 2018) few have considered spatial autocorrelation (Rocha et al., 2018; Santibanez, Kloft, & Lakes, 2015; Santibanez, Lakes, et al., 2015; W. Yang, Deng, Xu, & Wang, 2018). Hence, research on machine learning regression model that considers spatial autocorrelation using spatiotemporal data remains challenges. This study aims to explore the suitability of mixed effects learning regression model to capture spatial pattern from spatial datasets.

1.2. RESEARCH AND IDENTIFICATION

Machine learning regression models are used to predict continuous outputs, and they have been applied in many disciplines. However, these techniques do not consider spatiotemporal data.

In this study, we focus on the development, analysis and evaluation of mixed effects learning models; in particular, we focus on MERF and MESVR to reveal spatial patterns while improving the prediction of continuous outputs. Experiments will be done to test the suitability of mixed effects machine learning to datasets that have clustered structure by geographical relationship. Several machine learning approaches, for instance, MERF, Random Forest (RF), MESVR and Support Vector Regression (SVR) will be evaluated using spatial datasets.

Finally, the performance of the model using mixed effects will be compared and evaluated against vanilla machine learning using evaluation metrics. Analysing the degree of spatial autocorrelation of the residuals and looking at the required computation time of each model type.

1.2.1. RESEARCH OBJECTIVE

The main objective is to investigate whether mixed effects machine learning model; RF and SVR able to capture spatial pattern, improve predictive performance while keeping the spatial autocorrelation in the regression residual low compared with their vanilla counterparts given spatial datasets.

1.2.1.1. RESEARCH SUB OBJECTIVES

1. Review vanilla machine learning regression model; RF, SVR, and their mixed effects counterparts; MERF and MESVR and relate them to spatial data.
2. Design, develop and evaluate general and mixed effects machine learning regression models using spatiotemporal (crowdsourced) data from variety domain.

1.2.1.2. RESEARCH QUESTIONS

The following research question will answer each research sub-objectives mentioned before.

Sub-objective 1:

1. How do vanilla RF, SVR and their mixed effects counterparts approach work?
2. How can machine learning regression model approaches be used to model spatial data?

Sub-objective 2:

1. Can MESVR approach be developed and if so, how to apply regression given spatial datasets?
2. How mixed effects machine learning approach deal with clustering in the data caused by geographical relationship?
3. How should the spatial features be applied to machine learning?
4. Which approaches perform better regarding predictive accuracy?
5. What is the difference between mixed effects and general machine learning regarding the degree of SAC in the residuals?

1.2.2. INNOVATION AIMED AT

The proposed work aims to investigate mixed effects machine learning to deal with spatial data. Previous studies have evaluated the performance of general machine learning regression model to consider spatial and temporal data using synthetic data and real data. The innovation of this research is to design, develop and compare the performance of a machine learning regression approach based on mixed effects and of general machine learning regression model that consider spatial autocorrelation of the data. Moreover, the innovation of this research also to develop of MESVR approach using object-oriented language and existing machine learning libraries. This work will provide a detailed report on the capability of each model to reveal spatial patterns and its accuracy using different experimental settings of spatial autocorrelation level from three real-world datasets.

1.3. PROJECT SET-UP

The outline of the research as follows:

a) Literature review
b) Objective and data exploration
c) Experimental setup
d) Model performance evaluation

1.3.1. PROJECT WORKFLOW

In the first stage of this research, a review of the literature would be carried out on machine learning regression models. Among several algorithms, special attention will be given to mixed effects approaches; which have been previously applied to clustered data such as MERF and MESVR. These approaches have not been tested with spatial data. In this stage, observation on literature will also be focused on how to design and develop MESVR algorithm using object-oriented language and existing SVR library since MESVR library is not published for the public.

The second stage to this research is data exploration including a) data acquisition and preparation, b) data pre-processing, c) features engineering and d) spatial patterns. In the first stage involved how to retrieve the data and observe the data. In the data pre-processing we removed the unwanted data by data extraction, then we cleaned the empty or no-data known as NULL/NaN in the datasets. Feature engineering was performed to obtain the features used to train the model, including complaint features, temporal features and lagged spatial features. Lagged spatial features consist of temporally lagged spatial lag and LISA's quadrant.

The third stage of this research is experimental setup. In this stage, we determined the approach on how to develop the model including varying features into several experiments, hyperparameter tuning and model evaluation in the cross-validation.

The final stage is to evaluate the performance of the regression models. Existing evaluation metrics will be used to evaluate the model, for instance, r squared (R2), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Median Absolute Error (MAD). To evaluate spatial autocorrelation to the residuals regression, true and predicted response, Moran's I was used. This method has been used widely to measure spatial autocorrelation (Dormann et al., 2007).

1.3.2. THESIS OUTLINE

The thesis outline divided into six chapters. The first chapter contains a brief introduction to the scientific problem and existing solutions in the scientific literature. The output of this chapter is a hypothesis mixed effects machine learning regression models can capture spatial patterns. The second chapter presents existing methods and algorithms of machine learning based regression model especially RF, MERF, SVR and MESVR. These methods have been successfully applied to spatial data and or just applied only to non-spatial data. The third chapter explains the case study objective and data exploration. It contains data pre-processing including feature engineering and spatiotemporal autocorrelation analysis in both target and features. The fourth chapter explains the experimental set-up which contains the modelling method including cross-validation strategy and hyperparameter tuning on each machine learning algorithms used to create machine learning regression models. It also explains the model evaluation on both performance and spatial autocorrelation on residuals regression. The fifth chapter presents the results and comparison of each machine learning models. In this chapter also discusses the pros and cons of mixed effects models and their standard counterparts in terms of their ability to capture spatial patterns and their performance. The final chapter contains the conclusions and recommendations for future work.

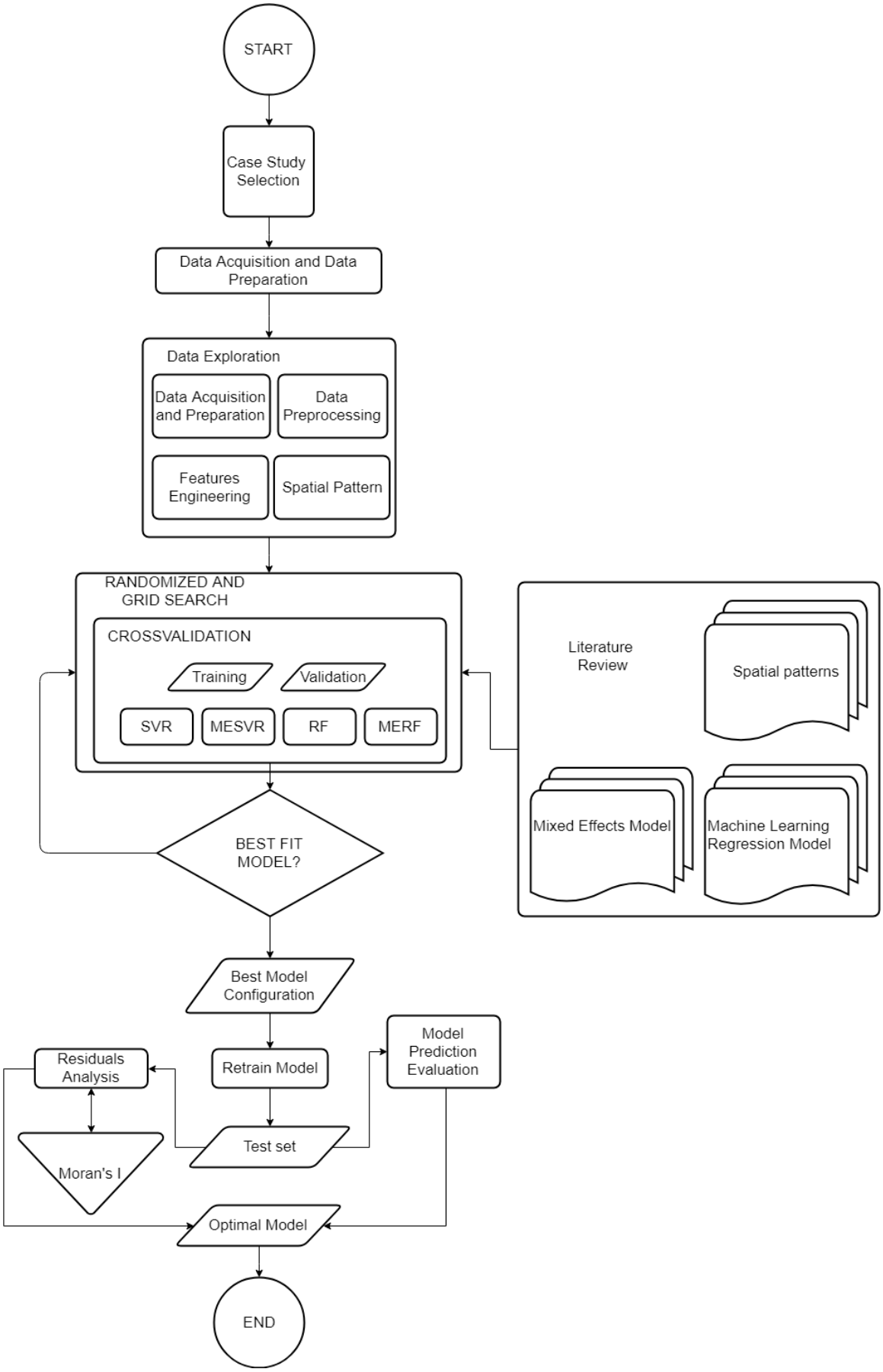


Figure 1. Flowchart of the project

2. LITERATURE REVIEW

This chapter covers the theoretical background and reviews of spatial pattern and existing relevant machine learning algorithms to the problem considered in this research. In the subsection 2.1, spatial patterns, their occurrences, shapes and how to measure in the dataset are discussed. The following subsection 2.2, I explain machine learning regression and their mixed model counterparts, their algorithms, parameters and how they train the model to learn from the data as well as relevant studies on which these algorithms have been applied. Moreover, the proper argument for the chosen algorithms applied to the problem statements are also described.

2.1. SPATIAL PATTERN

The origin of spatial patterns can be traced through two different ways; spatial dependence and spatial autocorrelation (Legendre et al., 2002). Spatial dependence and spatial autocorrelation have different meanings. Spatial dependence means that the response variable has spatially structure because it relies on the random features which have association with each other at different geographic location (Legendre et al., 2002). The equation of spatial dependence can be formulated:

$$y_i = \mu_y + f(x_i) + \varepsilon_i \tag{2.1}$$

This equation states response y at i location is global mean μ_y, the function of explanatory variables at location i or called 'local effects' and random error ε_i, $i = 1, \dots, n$. On the one hand, spatial autocorrelation assumed that response variable y at location i has relationship between y itself. The equation of spatial autocorrelation is given by:

$$y_i = \mu_y + \sum_{j=1}^{n} f(y_j - \mu_y) + \varepsilon_i \tag{2.2}$$

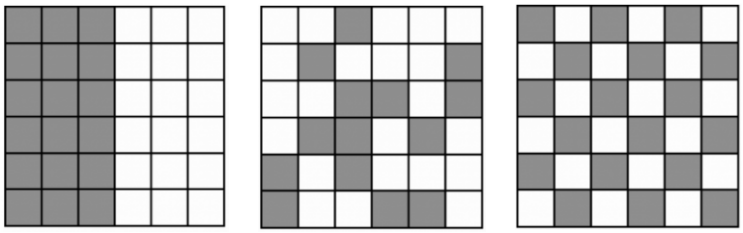
The model implies that the response at i-th unit is the global mean μ_y modulated by the sum of weighting function of response value at j-th units which neighbourhood of i and random error ε_i, $i, j = 1, \dots, n$.

Spatial autocorrelation analysis is used to measure the magnitude of spatial pattern (Chou, 1995). The concept of spatial autocorrelation primarily derived from the degree of geographical objects similarity in the space (Lichstein et al., 2002). It comes into two terms; the distance between geographical objects and its attribute or value. One of the common ones used statistical formula to compute the degree of spatial autocorrelation is using Moran's I (Zhao, Wang, & Shi, 2018). The Moran's I value can be obtained given by the equation:

$$I = n/S_0 \sum_{i=1}^{n}\sum_{j=1}^{n} z_i w_{i,j} z_j \Big/ \sum_{i=1}^{n} z_i z_i \tag{2.3}$$

Where n is the number of geographical units, $w_{i,j}$ is a spatial weight between i-th and j-th units, $z_i = y_i - \bar{y}$ is a global mean values and S_0 is the sum of spatial weight matrix, $i, j = 1, \dots, n$.

Moran's I value ranged from +1 to -1, positive value means strong positive autocorrelation and has clustering effects while negative value means otherwise and has scattered patterns. Zero value indicates there is no spatial correlation and has random patterns (Sawada, 2001). These patterns can be illustrated as in Figure 2.



(A) (B) (C)

Figure 2. (A) showing positive autocorrelation, (B) showing no correlation and (C) showing negative autocorrelation (Sawada, 2001)

Equally important as Global Moran's I in this study also consider Anselin's Local Moran's I coefficient known as Local Indicator of Spatial Association (LISA). Anselin's LISA measures the degree of spatial autocorrelation in a local specific context to allow further insight of clustering in the particular area (Feng, Chen, & Chen, 2018). The LISA formula is given by the equation:

$$I_i = (n-1)\frac{z_i \sum_{j=1}^{n} z_i w_{i,j} z_j}{\sum_{i=1}^{n} z_i z_i} \tag{2.4}$$

Using LISA, spatial datasets will be classified into four groups. Positive value of LISA indicates high values surrounding by high (HH group) and low value surrounding by low values (LL group) while negative value of LISA indicates high value surrounding by low values (HL group) and vice versa (LH group). The last two groups mentioned earlier considered as an outlier while the statistical significance of HH known as hot spots and LL known as cold spots (Anselin, 1995).

Alongside with spatial autocorrelation, dependent variable might be temporally autocorrelated due to seasonality (Hoef, London, & Boveng, 2010). Therefore, period, for instance, date, week, month and year could be random features that induced spatial dependence.

2.2. MACHINE LEARNING REGRESSION

Machine learning can be distinguished into four categories; supervised, semi-supervised, unsupervised and reinforcement learning. In supervised learning, the model is merely learning how to map given input features or explanatory variable x and given target or output variable y in the training sample datasets. The training sample acts as a supervisor in the learning process. In supervised learning, when the output variable is categorical or discrete value then it is classification, but when the output variable is continuous value, then it is a regression. In this study, we are interested in the regression problem. The simple formula to explain regression problem is given by the equation:

$$y = f(x) + \mathrm{b} \tag{2.5}$$

The purpose of regression is to estimate target y value using function f(x) from given input datasets and their errors term. Moreover, in the regression, the model learns from the data in various techniques to minimize the bias and variance until at some point the model prediction achieved the best fit. Many

machine learning regression algorithms can be used to predict continuous output, and two of them are RF and SVR.

Machine learning regression RF algorithm based on the predictive power, the capability to handle categorical features, adaptable to features of data or in other words, it does not require to normalize the data and minimal efforts to tune the parameter. Moreover, RF also more interpretable than different complex machine learning algorithms such as Neural Networks (Deng, 2018).

SVR algorithm apart from its predictive power, it also carries a feature namely kernel parameter which is used to map a lower dimension to higher dimensional data (Bhattacharyya, 2018; Kleynhans, Montanaro, Gerace, & Kanan, 2017). Therefore, SVR uses kernel trick to compute the inner products in the feature space.

2.2.1. RANDOM FOREST

The random forest regression model is one of ensemble learning. Ensemble method works by aggregating several base prediction estimators to decrease variance and bias. There are many kinds of ensemble method, for instance averaging, boosting and stacking. RF using averaging ensemble method, in which the final predicted value is the average value of all the decision trees. Hence, it allows better model predictive performance compared using only single base estimator (Pedregosa et al., 2011; Smolyakov, 2017), resistant to multicollinearity and insensitive to outliers (Breiman, 2001). The goal of RF is to minimize the variance of bagging by reducing trees correlation without increasing the bias (Hastie, 2017).

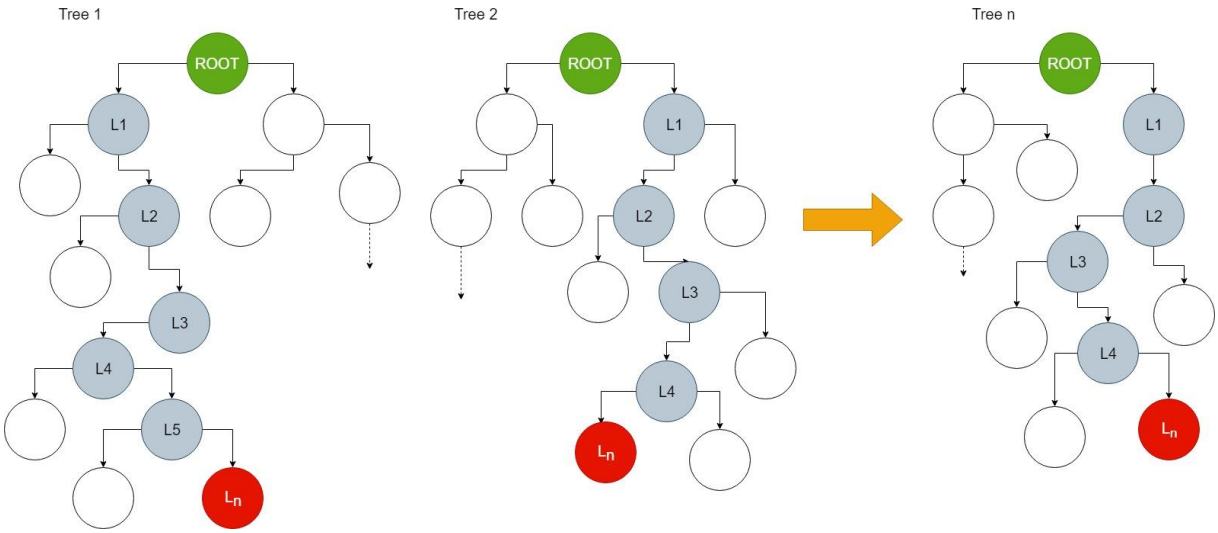


Figure 3. Tree model development concept in random forest

RF prediction estimators are composed of decision trees with different depths and leaves that spawned given the number of features in the datasets. The number of branches on each tree in the forest as shown in Figure 3, can be measured starting from the top or root until the red filled circle counted through several levels of split nodes (L1, L2, …, Ln). The more splits it has, the more depth information can be captured from the data, thus reducing bias. Each split node has the various number of samples, but at least it has one sample. The split node can be identified inside the orange filled circle as shown in Figure 4. As the tree in nature, it also has a leaf. Almost similar with the split node, the split leaf has various number of samples inside the leaf, but it required minimally one sample. In contrast with the split node, the leaf node does not have children. It can be seen in figure 4, the leaf inside the blue filled circle. Moreover, RF randomly uses a subset of features instead of all features and randomize the tree (Breiman, 2001; Hastie,

2017; Hengl, Nussbaum, Wright, Heuvelink, & Gräler, 2018; Smolyakov, 2017). A branch of a tree in RF based on the bootstrap sample from training datasets as can be seen in Figure 4.

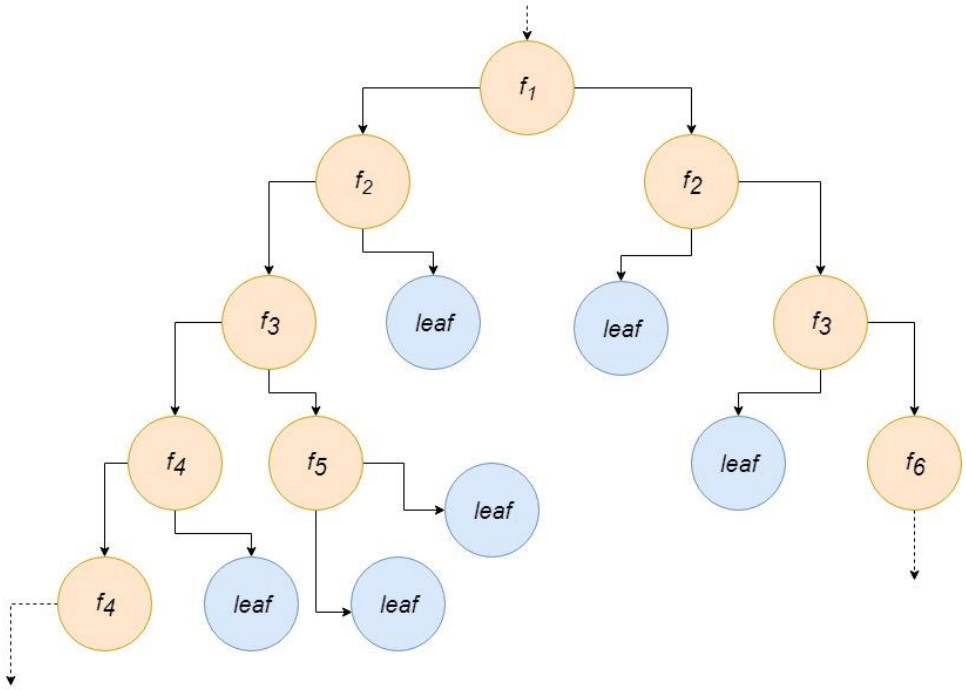


Figure 4. A branch of the tree from a subset of features (f_n, n = 1, …, n) in RF that use the bootstrap sample from training datasets

RF regression estimator is given by:

$$\hat{f}^I(x) = \frac{\sum_{i=1}^{I} t_i^*(x)}{I} \tag{2.6}$$

where $\hat{f}^I(x)$ is random forest estimator, individual bootstrap sample i, I is the total number of trees which represent the number of estimators and $t_i^*(x)$ is individual decision tree function:

$$t_i^*(x) = \mathrm{t}(\mathrm{x}; z_{i1,\dots}^* z_{in}^*) \tag{2.7}$$

where z_{in}^* $(\mathrm{n} = 1 \dots \mathrm{N})$ is n-th training sample from given datasets with x input features and y target.

Hence, to solve the problem when the dependent variable belongs to a particular location j given the equation 2.6:

$$\hat{f}^I(x_j) = \frac{\sum_{i=1}^{I} t_i^*(x_j)}{I} \tag{2.8}$$

$$such\ that, t_i^*(x_j) = \mathrm{t}(x_j; z_{i1,\dots}^* z_{in}^*) \tag{2.9}$$

The optimum value of the RF parameter such as the number of branches, samples split and sample leaf node is required to find out through hyperparameter tuning. However, the creators of this method recommend to use $n_{features} = \frac{1}{3} m$ where m is the number of features of the data and the minimum split node is five (Hastie, 2017).

Relevant studies

Using real estate data to predict the price of housing and in order to make it spatially autocorrelated, Santibanez, Kloft, et al. (2015) interpolate the training set features and target using zip code level and measure the RF performance through RMSE and SAC residuals regression using Moran's I. Parameter tuning was done using grid search repeated five cross-validation. They claimed the results obtained have relatively good RMSE but also has high clustering pattern in the residuals. However, there is a major drawback such as using default parameter tuning to train all the models. Thus, these models seem underestimated by observing at r-squared results. The results would be misleading because the errors prediction rate becomes high.

RF regression was also used to predict the spatiotemporal pattern of concentration and distribution of particle matters with size less than ten micrometres in China (G. Chen, Knibbs, et al., 2018; G. Chen, Wang, et al., 2018). The training datasets were aggregated to a province resolution. Parameter tuning was done using ten folds cross-validation and 500 iterations. The RMSE results were good despite SAC analysis was not performed.

Their approaches are adopted in this research by using zip code resolution to aggregate input features and the response. Moran's I also applied to evaluate the SAC residuals and ability of the model to capture the spatial patterns. R-squared also applied to evaluate the model prediction accuracy. However, there are slightly different to evaluate model performance, this study used MAE, MAD to evaluate and compare RF and MERF and use RMSE to evaluate SVR and MESVR model. All the metrics were used to evaluate and compare mixed effects models. Consider time allocated to MSc thesis; instead of using 500 iterations, we use random search with maximum 200 iterations. Also, in the cross-validation method used in this research, we consider the spatial structures in the data. Hence, we used seven folds instead of five nor ten folds.

Recently, random forest generic framework to predict spatiotemporal features has been proposed (Hengl et al., 2018). The framework uses buffer distance from observation points or geographical coordinates used as features. The model performance claimed to have less biased and able to capture spatial patterns. A similar approach was also applied to this research for using zip code as features in the learning process to create a gap between cluster. Moreover, we also use spatially lagged of response and LISA's quadrant to inform the models the distance and weight of response surrounding cluster.

2.2.2. SUPPORT VECTOR REGRESSION

SVR is a supervised learning model and used for regression despite being originally created for classification (Jin, Sun, Wang, Wang, & Yan, 2013). SVR uses structural risk minimization approach rather than the empirical risk minimization to optimize the model through minimizing error within a certain threshold (Bhattacharyya, 2018) and model complexity (Baydaroğlu, Koçak, & Duran, 2018; Jin et al., 2013). Hence, the SVR model is robust to solve non-linear problems (Smola & Sc Olkopf, 2004; H. Yang, Huang, Chan, King, & Lyu, 2004)

SVR algorithm solves the non-linear problem using kernel tricks. SVR solves non-linear problems using kernel tricks. These kernels calculate the similarity of the samples in a high dimensional feature space. Hence transforming a non-linear problem into a linear one. The illustration of SVR can be seen in Figure 5.

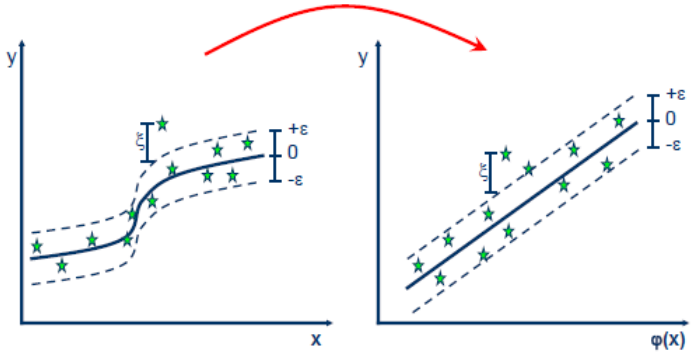


Figure 5. SVR solves non-linear problem using kernel function (Sayad, 2010)

Mathematically, non-linear SVR is formulated given by:

$$y = f(x) = \langle w, \varphi(x) \rangle + b \tag{2.10}$$

where w is weight vector, $\varphi(.)$ is the feature mapping function and b is independent and identical distributed errors or bias terms. Furthermore, SVR use Vapnik's epsilon ε loss function that defines a margin or errors tolerance. The higher value of ε, the larger errors are being tolerated. In contrast, set ε value to zero means every error will be penalized.

$$L_\varepsilon(y_i, f\langle x_i, w\rangle) = \begin{cases} 0, & if\ |y_i - f\langle x_i, w\rangle| \leq \varepsilon \\ |y_i - f\langle x_i, w\rangle|\ , & if\ |y_i - f\langle x_i, w\rangle| \geq \varepsilon \end{cases} \tag{2.11}$$

SVR solves linear regression in n-dimensional data with n > 1 using loss function and reducing model complexity by minimizing vector |w| which is induced slack variable $\xi_i,\ for\ i = 1, \dots, n$ to estimate deviation of training samples that located outside ε margin (Cherkassky & Ma, 2004), such that:

$$\begin{gathered} minimize\left(\frac{1}{2}\|w\|^2 + C\sum_i^n(\xi_i + \xi_i^*)\right) \\ subject\ to \begin{cases} y_i - f\langle x_i, w\rangle - b \leq \varepsilon + \xi_i \\ f\langle x_i, w\rangle + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases} \\ |\xi|_\varepsilon := \begin{cases} 0 & if\ |\xi| \leq \varepsilon \\ |\xi| - \varepsilon & otherwise \end{cases} \end{gathered} \tag{2.12}$$

Where C as regularization parameters is introduced. It is used as penalty factor; a huge constant value of C may induce overfitting while a minimal value of C may induce underfitting. According to Vapnik (1995), optimization formula can be transformed into dual problem α_p and α_p^* for each data point as follow:

$$f(x_i) = \sum_{p,q=1}^{n} (\alpha_{pi} - \alpha_{pi}^*) K(x_{pi}, x_{qi}) + b \tag{2.13}$$

where $\alpha_i \geq 0$ and $C \geq \alpha_i^*$ and $K(x_{pi}, x_{qi})$ is kernel function for $i, p, q = 1, \dots, n$. There are three commonly used kernels; namely linear, polynomial and gaussian radial basis (Üstün, Melssen, & Buydens, 2006). In this study, we use radial basis function (RBF) kernel because it performs better than two others

to solve non-linear problem (Cawley, Talbot, Guyon, & Saffari, 2007). Using RBF kernel, gamma γ as free parameter of RBF is introduced. The RBF kernel function formula is given by the equation 2.14:

$$K(x_{pi}, x_{qi}) = \exp\left(-\gamma \left\| x_{pi} - x_{qi} \right\|^2\right), \gamma \geq 0 \quad (2.14)$$

The small gamma value means the kernel has large width or large variances, while the large gamma value means the variance might be small. Also, a large value of gamma may lead to high bias and low variance and vice versa.

SVR parameters; C, ε and γ value are the foundation of the SVR model. Therefore, it is important to select the most appropriate hyperparameters of SVR to ensure good generalization of the model (Cherkassky & Ma, 2004).

Relevant Studies

Several studies using SVR and consider spatial autocorrelation in the dataset have been proposed. To start with, multi-scale SVR proposed by Ballabio & Comolli (2010). Their approach was using more than one kernel with the same function to train the model. The first kernel was used to estimate of response while the second kernel function to estimate the residuals of fitted models. The model claimed has slightly better performance than vanilla SVR and kriging regression. However, it gets more complicated in the training process as the number of kernels increases and the model likely overestimated. Their approach was adopted in this MSc thesis to use mixed models instead using multiple kernels.

Santibanez, Lakes, et al. (2015) and Santibanez, Kloft, et al. (2015) using generated spatial datasets and real spatial datasets respectively, assessed SVR using radial basis kernel and claimed the model perform better than RF regarding RMSE and SAC in the regression residuals. It because the regularization step to generate more simple function and the strength of RBF as a kernel. Parameter tuning was done using five folds cross-validation, C and sigma variation. Their approach drawback already explained in the previous subsection. However, their approach is adopted in this research for the kernel selection and Moran's I to evaluate SAC. It is slightly different in the parameter selection, we utilize gamma and epsilon to optimize model instead of sigma.

Considering the temporal factor, Rocha et al. (2018) evaluated the performance of SVR using synthetic data to simulate hyperspectral data to predict leaf traits. To reduce overfitting, ten cross-validations was used. They used Durbin Watson method to evaluate serial autocorrelation of residuals of regression. The noise in the data that produce clusters makes predictive model become overfitting. In this research, we used seven folds in cross-validation to reduce overfitting in the model considering the size of data, features and spatial structures. However, in this study, the noise is naturally coming from the data.

2.3. LINEAR MIXED EFFECTS MODEL

The mixed model is a statistical model that comprises fixed effects and random effects. There are various types of mixed models such as Bayesian generalized linear mixed models, non-linear mixed models, linear mixed models, etc. In this study, we focused on the linear mixed effects model. In nature, data often has multifaceted data structures, such as containing a cluster of dependent variable (Zuur & Ieno, 2016) and linear mixed effects model become popular to solve cluster in the data (Blood, Cabral, Heeren, & Cheng, 2010; Zhang, Jie, Sun, & Pieper, 2016). Recall in the first chapter, there are two kinds of clustering in the data, hierarchical and longitudinal cluster as illustrated in Figure 6.

Linear mixed effects model has proved to solve the serial correlations problem in the dataset when using longitudinal data (Meng, Huang, Vanderschaaf, Yang, & Trincado, 2012). Serial correlation occurs when lagged version of particular variable highly correlated with itself over various time intervals. Other than that, it also able to handle correlated error structures found in temporal and spatial statistics (Hoef et al., 2010). Thus, using the linear mixed model approach, the error structures caused by spatial autocorrelation in the regression residuals theoretically can be diminished.

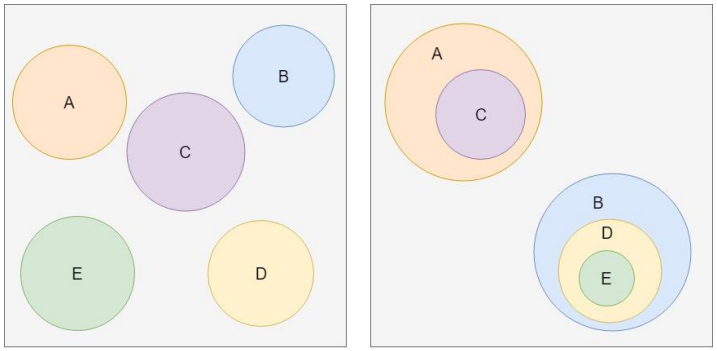


Figure 6. (A) Illustration of longitudinal and (B) hierarchical clustering in the data

Mathematically, the linear mixed effects algorithm is formulated by:

$$y_i = X_i\beta + Z_i b_i + \epsilon_i \tag{2.15}$$

where $y_i = [y_{i1}, \dots, y_{in_i}]^T$ is a vector response for n_i observations in cluster i, $X_i = [X_{i1}, \dots, X_{in_i}]^T$ is matrix of fixed effects features, $Z_i = [Z_{i1}, \dots, Z_{in_i}]^T$ is matrix of random effects features, $\epsilon_i = [\epsilon_{i1}, \dots, \epsilon_{in_i}]^T$ is an unknown vector errors, $b_i = (b_{i1}, \dots, b_{in_i})^T$ is an unknown vector of random effects coefficients in the cluster i, and β is an unknown vector of fixed effects coefficients. In linear mixed effects, it assumes that b_i and ϵ_i are independent and identical distributed as $b_i \sim N(0, D)$ and $\epsilon_i \sim N(0, R_i)$ where N is referred to normal distribution, while D and R_i are diagonal matrices features of b_i and ϵ_i respectively.

2.3.1. MIXED EFFECTS RANDOM FOREST

MERF algorithm was proposed by Hajjem et al. (2014) to tackle clustered and unbalanced repeatable measurements in the datasets. MERF is like linear mixed effects model as in the equation 2.15, except the fixed effects $X_i\beta$ replaced with random forest function $\hat{f}^I(x)$ as in the equation 2.6 to estimate fixed features coefficients.

$$y_i = \hat{f}^I(x) + Z_i b_i + \epsilon_i \tag{2.16}$$

Given equation 2.16, $Cov(y) = diag\left(cov(y_1), \dots, (y_n)\right)$. Covariance matrix of repeated measurement vector y_i for cluster i-th is $Cov(y_{,i}) = Z_i D Z_i^T + R_i$, hence there might be a correlation between cluster that is induced between cluster variance in term of random effects or within cluster variation in term R_i. It holds if D and R_i are diagonal and $cov(Z_i D Z_i^T) > 0$ and $cov(R_i) > 0$ even though ϵ_i are random and

independent distributed errors (Hulin & Zhang, 2006). R_i is diagonal matrix of the variances of errors $\sigma^2 I_{ni}, i = 1, \dots, n$. MERF uses out of bag prediction to estimate non-linear model using bootstrap dataset that does not contain record from original subset. Furthermore, Hajjem et al. (2014) also implemented expectation-maximization (EM) in order to estimate response y_i.

The EM algorithm is used to estimate parameters for multiple features to solve imbalance in the data (Borman, 2004). EM algorithm to find optimum y_i as proposed by Hulin & Zhang (2006) is as follows:

Set iteration index r as $r = 0, 1, 2, \dots, n$

$step\ 0. Set\ r = 0, \hat{\sigma}^2_{(0)} = 1, \hat{b}_{i(0)} = 0, \widehat{D}_{(0)} = I_q$

$step\ 1. Set\ r = r + 1, Estimate\ y^*_{i(r)}, \hat{f}^I(X_i)_r\ and\ \hat{b}_{i(r)}\ using$

i. $y^*_{i(r)} = y_i - Z_i \hat{b}_{i\,(r-1)}, i = 1, \dots, n$
ii. Build multiple trees in the forest using random forest algorithm with $y^*_{ij(r)}$ as response set to x_{ij} features using bootstrap training sample from the training sets $(y^*_{ij(r)}, x_{ij}), i = 1, \dots, n, j = 1, \dots, n_i$.
iii. Get random forest $\hat{f}^I(x_{ij})_r$ of $f(x_{ij})$ model using out of bag prediction
Let $\hat{f}^I(X_i)_r = \left[\hat{f}^I(x_{i1})_{(r)}, \dots, \hat{f}^I(x_{in_i})_{(r)}\right]^T, i = 1, \dots, n$
iv. Compute $\hat{b}_{i(r)}$ as
$\hat{b}_{i(r)} = \widehat{D}_{(r-1)} Z_i^T \hat{V}^{-1}_{i(r-1)} \left(y_i - \hat{f}^I(X_i)_r\right), where\ \hat{V}_{i(r-1)} = Z_i \widehat{D}_{(r-1)} Z_i^T + \hat{\sigma}^2_{(r-1)} I_{ni}, i = 1, \dots, n$

$step\ 2. Update\ \hat{\sigma}^2_{(r)}\ and\ \widehat{D}_{(r)}\ as$

$$\hat{\sigma}^2_{(r)} = N^{-1} \sum_{n=1}^{n} \left\{ \hat{\epsilon}^T_{i(r)} \hat{\epsilon}_{i(r)} + \hat{\sigma}^2_{(r-1)} \left[n_i - \hat{\sigma}^2_{(r-1)}\ trace\left(\hat{V}_{i(r-1)}\right) \right] \right\},$$

$$where\ \hat{\epsilon}_{i(r)} = y_i - \hat{f}^I(X_i)_r - Z_i \hat{b}_{i\,(r-1)}$$

$$\widehat{D}_{(r)} = N^{-1} \sum_{n=1}^{n} \left\{ \hat{b}^T_{i(r)} \hat{b}_{i(r)} + \left[\widehat{D}_{(r-1)} - \widehat{D}_{(r-1)} Z_i^T \hat{V}^{-1}_{i(r-1)} Z_i \widehat{D}_{(r-1)} \right] \right\}$$

$step\ 3.\ do\ loop\ step\ 1\ and\ 2\ until\ convergence$

These EM steps in MERF can be explained as; initially, set the default value for variance, random effects coefficient and diagonal matrix of unknown variance $\widehat{D}$. Next step is to calculate the response variable at cluster i as $y^*_{i(r)}$ as $y_i - Z_i \hat{b}_{i(r-1)}$. Next step, estimate fixed effects using random forest with out of bag prediction given $\left(y^*_{ij}, x_{ij}\right)$. Estimated $\hat{f}^I(x_{ij})$ from random forest are used to find the random effects coefficient $\hat{b}_i$ at particular cluster i. Last step is to compute variance $\hat{\sigma}^2$ and matrices $\widehat{D}$ from estimated residuals and random effects respectively. This algorithm runs iteratively until its convergences.

Additionally, MERF algorithm utilizes generalized likelihood (GLL) to calculate training loss in the model development. GLL will eventually reach convergence when the model achieved the best fit as shown in Figure 7.

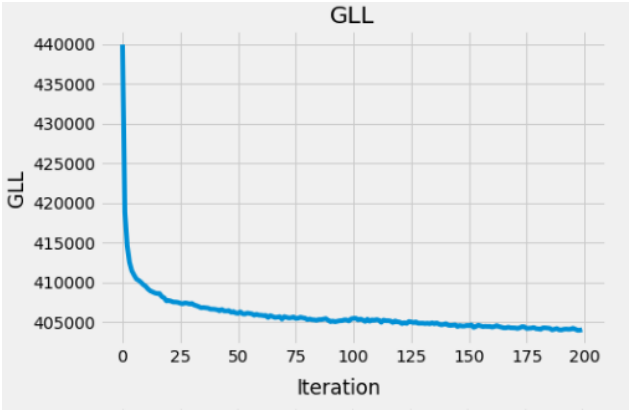


Figure 7. GLL in the training process reach convergence with 200 iterations

Hajjem et al. (2014) conducted a testing using simulation and real datasets. Model performance evaluation was done using Prediction Mean Square Error (PMSE). Five models were tested and compared to MERF performance. The MERF method performed better because it had lower PMSE compared with standard RF.

2.3.2. MIXED EFFECTS SUPPORT VECTOR REGRESSION

The MESVR algorithm was proposed by Cho (2010) to handle longitudinal cluster in the sample datasets. It is almost similar to MERF except in this algorithm use LS-SVR instead of vanilla SVR. LS-SVR proved has better accuracies, able to handle extensive datasets and better processing computation time compared vanilla SVR (Guo, Li, Bai, & Ma, 2012; Steinwart & Thomann, 2017). It is used to solve the non-linear problem. Given equation 2.10 and 2.15, MESVR can be formulated as follows:

$$y_{ij} = \langle w, \varphi(x_{ij}) \rangle + Z_{ij} b_i + b_0 + \epsilon_{ij} \tag{2.17}$$

Where x_{ij} assumed related with y_{ij} as (y_{ij}, x_{ij}), y_{ij} is response variable of j-th sample at cluster i. $\varphi(.)$ is non-linear mapping function, Z_{ij} is random effects features, b_i is random effects parameter vector normally distributed as $N(0, D)$, $\epsilon_i \sim N(0, R_i)$ and b_0 is the bias. Sample observation j = 1, …, n, cluster i = 1, …, n_i.

The optimization problem can be estimated given equation 2.17, known D and R_i :

$$minimize \left(\frac{1}{2}\|w\|^2 + \frac{\lambda_1}{2} \sum_{i=1}^{N} (b_i D^{-1} b_i) + \frac{\lambda_2}{2} \sum_{i=1}^{N} \sum_{j,k=1}^{n_i} \left(\epsilon_{ij} R_i^{-1} \epsilon_{ik} \right) \right) \tag{2.18}$$

$$subject\ to\ y_{ij} = \langle w, \varphi(x_{ij}) \rangle + Z_{ij} b_i + b_0 + \epsilon_{ij}$$

where λ_1 and λ_2 are the regularization parameter. The Langrangian function obtained given equation 2.17 and 2.18:

$$L = \ minimize\left(\frac{1}{2}\|w\|^2 + \frac{\lambda_1}{2}\sum_{i=1}^{N}(b_i D^{-1} b_i) + \frac{\lambda_2}{2}\sum_{i=1}^{N}\sum_{j,k=1}^{n_i}\left(\epsilon_{ij} R_i^{-1} \epsilon_{ik}\right)\right.$$

$$\left. + \sum_{i=1}^{N}\sum_{j,k=1}^{n_i} \alpha_{ij}(y_{ij} - \ \langle w, \varphi(x_{ij})\rangle + \ Z_{ij} b_i + b_0 + \ \epsilon_{ij})\right) \quad (2.19)$$

where α_{ij} is Langrangian multipliers. Given equation 2.19, optimum parameters can be estimate by their first order derivative. It can be written in the simple formula as:

$$\begin{bmatrix} 0 & 1N_n^T \\ 1N_n & K + \frac{1}{\lambda_1} ZDZ^T + \frac{1}{\lambda_2} R \end{bmatrix} \begin{bmatrix} b_0 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}$$

Where N_n is n_i observation at cluster i, 0 and 1 are vectors of zeros and ones respectively, k $= 1, \ldots, n_i$,K is kernel matrix $K(x_{ij}, x_{kl})$, Z is diagonal matrix $Z = diag(Z_1, \ldots, Z_n)$, D is diagonal matrix $D = diag(D, \ldots, D)$ and R also diagonal matrix $R = diag(R_{i,\ldots,} R_n)$, $y_i = [y_{i1}, \ldots, y_{in_i}]^T$, $\alpha_i = [\alpha_{i1}, \ldots, \alpha_{in_i}]^T$. The final MESVR estimator equation given (x_0, z_0) as follow:

$$\hat{y}(x_0, z_0) = \ b_0 + \sum_{i=j}^{n}\sum_{j=1}^{n_i}(\alpha_{ij} K(x_{ij}, x_0) + \ b_i z_0) \quad (2.20)$$

where b_0 and α_{ij} is solved with linear regression. Hence, to estimate $\hat{y}$:

$$\hat{y} = \ \hat{b}_0 1_{N_n} + K\hat{\alpha} + Z\hat{b}$$

Furthermore, Cho (2010) tested the algorithm using the RBF kernel and GCV function to estimate the optimal value of hyperparameters. The result proved slightly increased performance and prediction over standard LS-SVR and linear regression using simulation and real datasets (Cho, 2010; Seok et al., 2011).

Relevant studies

There might be an intra-cluster correlation influenced by random effects (Hajjem et al., 2014; Verbeke, Molenberghs, & Rizopoulos, 2010). In another words, the prediction at particular cluster could be affected by random effects parameters (Westfall, 2016). In this MSc thesis, we have hypothesized that the MERF and the MESVR able to capture spatial patterns through correlation between cluster via cluster variance regarding random effects (spatial autocorrelation) or within-cluster variation regarding errors (spatial dependence).

Apart from that, we adopted the MERF framework to develop MESVR. It is assumed that the MERF framework is modular and flexible. Thus, the non-linear regression in MERF was replaced with SVR to compute fixed effects. Given the MERF framework equation in 2.16, the random forest $\hat{f}^I(x)$ was replaced with the non-linear mapping function of SVR $\langle w, \varphi(x_{ij})\rangle$. Hence, the equation for MESVR using MERF framework can be calculated using this formula:

$$y_i = \ \langle w, \varphi(x_{ij})\rangle + \ Z_i b_i + \ \epsilon_i \quad (2.21)$$

The rest functions inside existing MERF are used, for instance, EM and GLL. The out of bag prediction in random effect replaced with cross-validation split in the training set.

3. CASE STUDY: OBJECTIVE AND DATA EXPLORATION

This chapter contains the objective of the selected case study. A brief description of the study area and data exploration in the subsection. Data exploration contains; a) data acquisition and preparation, b) data pre-processing, c) feature engineering and d) spatial-temporal autocorrelation.

3.1. OBJECTIVE AND STUDY AREA

The case study in this research is crimes that occurred in New York City, New York, USA. As can be seen in Figure 8, New York City was divided into five boroughs; Bronx, Queens, Brooklyn, Manhattan and Staten Island. It also has approximately 248 unique number of zip codes that split the area into finer resolution. Moreover, New York City also becomes the most populous city in the USA (Barron, 2018). Despite the most populous city, the number of crimes in New York City began to decline dramatically in the late 20th century (Fox, 2018; New York City Police Department, 2018). Therefore, the objective of this case study is to capture the crime pattern and predict the number of crimes at the zip code level based complaint features in New York City.

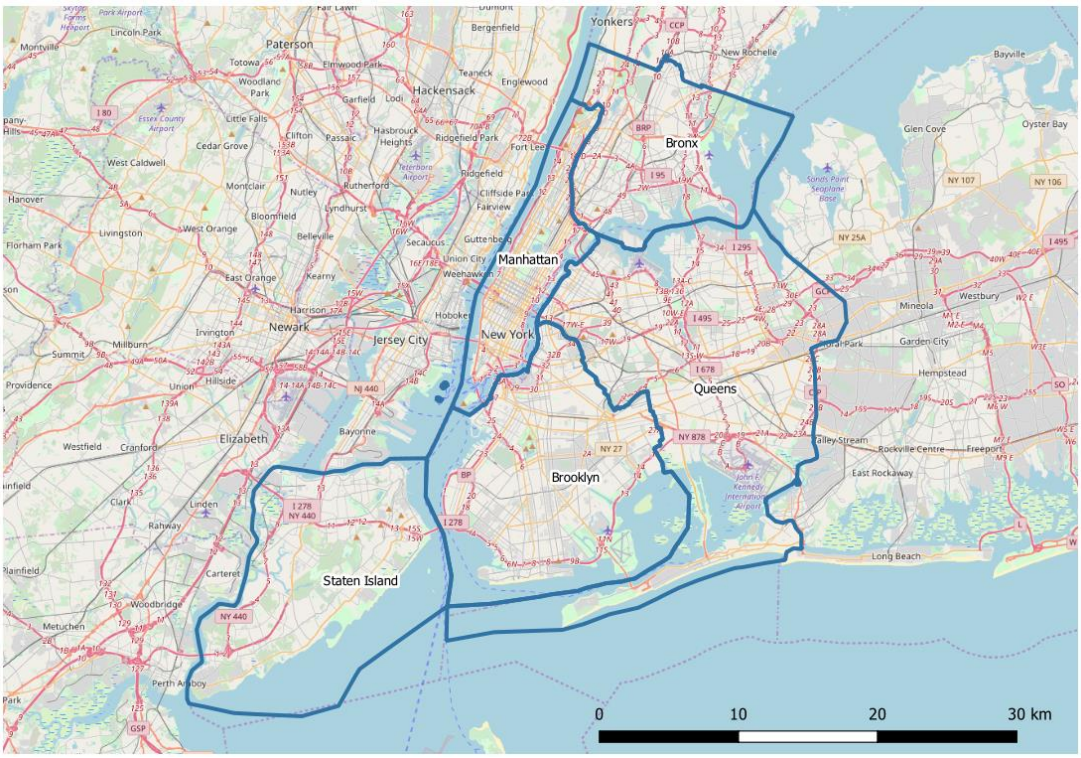


Figure 8. The study area for investigating the ability of machine learning regression to capture spatial patterns situated in New York City, United States.

3.2. DATA ACQUISITION AND PREPARATION

The spatiotemporal dataset used in this research was gathered from crowdsourced GIS data, that are publicly available online through the NYC open data website (NYC Information Technology & Telecommunications, 2019). The datasets contain three different sources, and each source has its purpose; namely the features, the response variable and the spatial aggregator.

The dataset for explanatory features was acquired from the 311 Service Requests database that was collected from 2010 to present. Originally, it contained 41 columns, around 18 million rows and yielded more than 10 GB file size with csv format extension. The dataset for the response variable was obtained from the New York City Police Department (NYPD) Complaint Data Historic. It contains all valid crimes that were reported to NYPD from the year 2006 until 2017. It originally consisted of 35 columns with more than six million rows and generated more than 1.5 GB file size in csv format. Both datasets have columns on latitude and longitudinal coordinates. The generic overview of both datasets can be seen in Table 3.1. The spatial aggregator data is New York City zip code boundary (Department of Information Technology & Telecommunications (DoITT), 2018). The spatial dataset, zip code boundary, has geometry type of multipolygon.

Table 3.1. Generic overview of complaints and crimes dataset

	Complaint Datasets	**Crime Datasets**
Datatype	**Count**	**Count**
float64	5	6
int64	1	2
object	35	16
Memory Usage: 5.7+ GB		Memory Usage: 1.1+ GB
Disk Usage: 10.6+ GB		Disk Usage: 1.5+ GB

3.3. DATA PRE-PROCESSING

The aim of data pre-processing is to provide good quality training dataset. The performance of the model has linear relationship with the quality of the training sample data that fed into the model itself (Malik, 2018). There are three main steps of data pre-processing; namely data extraction, data cleaning.

Data Extraction

Data extraction has significant role in improving learning time and ramping up the size of the data. The aim of data extraction process is to acquire the baseline appropriate input data matrix as learning sample. Extracting information in the dataset was performed by removing unrelated attributes and selecting the important one that has an impact on the learning process. It performed by extracting an appropriate length of time period. The length of the year for both datasets is different. Crime dataset was obtained by extracting the data started from year 2009 through 2017 while complaint dataset from year 2010 to 2017. To extract information, both datasets that contain valid crimes and complaints were loaded into physical memory and extracted using panda's dataframe.

Apart from removing unrelated features, we renamed the attribute name and converted its datatype in order to make them compatible to PostgreSQL standard and further reduced file size in both disk and physical memory. Hence, efficient computation time involving these datasets can be achieved.

Data extraction resulted in massive reduction in the size of the data. The amount of data that resides in the memory went down from 5.7+ GB to 1.1+ GB for complaint dataset and 1.1+ GB to 440+ MB for crime dataset. The output of data extraction on both complaint and crime datasets after passing through this step are listed in Table 3.2 and Table 3.3 respectively.

Table 3.2. Complaint datasets after extracting and removing unrelated attributes.

Attributes	Datatype	Description
id	int64	indexing
complaint_type	object	types of complaint
zip	object	zip code reported
x	float64	x state plane in meters
y	float64	y state plane in meters
latitude	float64	geographic coordinate wgs84
longitude	float64	geographic coordinate wgs84
created_date	datetime64[ns]	date incident reported

Table 3.3. Crime datasets after information extraction and unrelated attributes removal.

Attributes	Datatype	Description
id	int64	indexing
report_date	datetime64[ns]	date complaints reported
offense_desc	object	types of crime incident
x	float64	x state plane in meters
y	float64	y state plane in meters
latitude	float64	geographic coordinate in wgs84
longitude	float64	geographic coordinate in wgs84

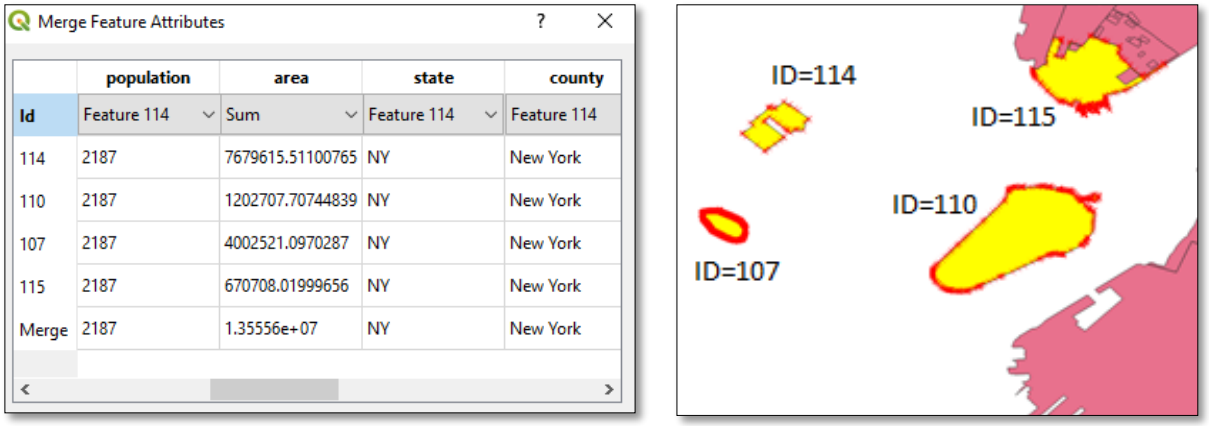
Data Cleaning

Data cleaning has the purpose to fix the missing data in the datasets that come from various sources, for instance, input data error, programming error, error on data transfer, etc. Missing data reparation by filling any value, such as median, mean, backfill, etc, is not usually simple generic and appropriate solution, because it depends on the attribute knowledge domain (Brownlee, 2013; Malik, 2018).

There are two kinds of types of missing data that occurred in the complaints and crimes datasets; NaN/NULL and zero in the latitude, longitude, offense_desc and complaint_type column. The percentage of missing data in latitude and longitude in complaints and crime datasets are low by 9.03% and 3.31% respectively. Therefore, we removed all of the data that has missing value.

Data cleaning was also performed to spatial data, precisely to their geometry identity to remain consistent according to Open Geospatial Consortium (OGC) compliance. Thus, data duplication and ambiguity in the later processing stage can be avoided, such as SQL query in the feature engineering. Data cleaning to spatial data was performed using SQL/PostgreSQL – PostGIS.

According to OGC compliance, multipolygon is valid if and only if all elements are valid and there are less than two elements that intersect. However, the boundaries of any two elements may touch a limited number of points (The PostGIS Development Group, 2018). Hence, the implementation of multipolygon in the PostgreSQL through postgis should be one and more tuple may contain different attribute data but originate from similar or equivalent geometries. This is not the case in this study; multiple data that have similar zip code but have more than one geometry had to be merged to avoid data ambiguity and duplication as shown in Figure 9.



	population	area	state	county
Id	Feature 114	Sum	Feature 114	Feature 114
114	2187	7679615.51100765	NY	New York
110	2187	1202707.70744839	NY	New York
107	2187	4002521.0970287	NY	New York
115	2187	670708.01999656	NY	New York
Merge	2187	1.35556e+07	NY	New York

(A) (B)

Figure 9. (A) Realization of spatial data in the geodatabase showing the same geometry split into four tuples. (B) visualization of spatial data showing the geometry filled with solid yellow colour has the same zip code level

3.4. FEATURE ENGINEERING

In this stage, raw data transformed into features. These features represent problem wrapper to the predictive models. Feature engineering might improve model accuracy on the test set (Shekhar, 2018). Features engineering was accomplished using SQL/PostgreSQL query and Python's library Pandas. A data manipulation method that was used to extract features from the raw datasets is one hot encoding. One hot encoding is a process transforming categorical or discrete features into one hot numeric array.

As it can be seen in Figure 10, complaint_type column contains the discrete value, such as blocked driveway, noise residential and noise vehicle. Using one hot encoding method, these values are converted to three features; namely blocked_driveway, noise_residential and noise_vehicle. Their values converted to binary value that machine learning regression can understand better.

Feature extraction for the explanatory features was performed using SQL/PostgreSQL query to extract the number of complaints. The feature selection method was based on the ten largest number of complaints, for instance, noise residential, water problem, street condition, generic noise and blocked driveway. Another method to select the other features; graffiti, dirty condition and noise vehicle was purely random selection to find out that these features contribute to initiate crimes in a particular area. Features were selected and extracted from complaint_type column and aggregated to zip code level and particular temporal resolution. Spatial aggregation to zip code resolution was performed using SQL GROUP BY and ST_Intersects functions. Thus, each tuple has its geometry from specific zip code. One hot encoding also applied to date column. Using pandas, datetime was split into a week, month and year.

complaint_type text
Noise - Vehicle
Noise - Commercial
Noise - Commercial
Noise - Residential
Noise - Street/Sidewalk
Blocked Driveway
Blocked Driveway
Blocked Driveway
Noise - Vehicle

(A)

noise_residential numeric	noise_vehicle numeric	dirty_cond numeric	graffiti numeric	street_condition numeric	blocked_driveway numeric
0	0	0	0	6	0
9	5	0	118	8	0
6	1	2	1	12	0
1	0	1	0	8	0
0	0	0	0	2	0
5	2	3	0	10	0
2	0	2	2	3	0

(B)

Figure 10. (B) is one hot encoding product from (A)

Same treatment to response variable, one hot encoding applied to offense_desc established from five crime categories; namely larceny, assault, burglary, robbery and drugs. The selection for one feature, which is drugs was obtained based random selection while the others based major crime occurrences in New York City (Newsday, 2018). The complete feature engineering output can be seen in Table 3.4 below.

Table 3.4. Complaint features along with lagged spatial features and response variable

Complaint	**Lagged spatial features**	**Response**
noise_residential	y_offset_splag	crime_num
noise_vehicle	quadrant_0	
dirty_condition	quadrant_1	
graffiti	quadrant_2	
street_condition	quadrant_3	
blocked_driveway	quadrant_4	
generic_noise		
water_problem		

Feature construction was also performed to create two lagged features; time offset spatial lag of response variable and time offset quadrant of LISA's indicator. Response variable shifted to $t + 1$, where t is time of year. Spatial lag of response value was obtained using equation 2.2 by calculating the weighted sum of each observation neighbour of shifted response. Spatial lag was used to estimate the spatial correlation between response at a particular cluster and its neighbour. In case offset quadrant of LISA's, the value was obtained by calculating local moran of shifted response variable using equation 2.4. LISA's quadrant label was used to estimate correlation significance between cluster. As LISA's quadrants contains discrete value, thus they were one hot encoded to inform the model the significance of cluster of each response in particular cluster. The lagged spatial features can be seen in Table 3.4

As RF and SVR model able to learn the data from each cluster and make it equally comparable regarding performance and ability to capture spatial patterns with MERF and MESVR method, the zip code and month were one hot encoded as features. Zip code as features informs the model that particular sub-sample data belongs to a specific cluster object. The same treatment applied to month features to improve model performance. The result as shown in Table 3.5 and Table 3.6.

Table 3.5. Zip code id matrix being transposed as features set in the training set

Cluster features [cf_ohe]				
Zip code	**cluster_Z_1**	**cluster_Z_2**	**……..**	**cluster_Z_n**
Z_1	1	0	……..	0
Z_2	0	1	…….	0
……	…….	……	…….	……
Z_n	0	0	…….	1

Table 3.6. Month matrix being transposed as features set in the training set

Month features				
Month	**month_1**	**month_2**	**…..**	**month_12**
1	1	0	…..	0
2	0	1	…..	0
……	…….	……	…..	……
12	0	0	…..	1

Baseline sample training constructed as the cartesian product between zip code boundary data, weekly data and monthly generated series. Hence, when there is no complaint nor crime reported in particular time scale and place, then the data will be filled with 0. The final input data matrix is obtained by inner joining between explanatory, response and baseline datasets. It can be seen in Table 3.7

Table 3.7. Input data matrix for the machine learning algorithm

		Data dimension			
	Response	1st dim	2nd dim (offset SL response)	3rd dim (offset LISA)	n dim
Training data	y_{ij}	x_{i1_1}	$x_{i1_2}\,(t-1)$	$x_{i1_2}\,(t-1)$	x_{in_i}
Testing data	.	.	.	.	.

Feature Scaling

There are several methods to normalise the features. The choice of the method depends on the data distribution and the occurrence of outliers in the data. We observed that the number of outliers in the dataset is quite high. Apart from outliers, we also observed the distribution of the data in each covariate and zip code are not always normal distribution. Thus, we conclude the appropriate feature scaling method to use robust scaling. Robust scaling method works by removing the median and uses interquartile range. Hence, this method robust to outliers.

Mathematically, it is calculated using formula:

$$X\_Scaled = \frac{X - median\ (X)}{IQR}$$

where IQR is interquartile range.

Features Exploration

To determine the random effects variable from complaint features, we compared the pattern of data distribution of each feature in the training set with the response variable. As it can be seen in Figure 11, dirty_condition and noise_vehicle have a similar pattern. Hence, we select these features as random features.

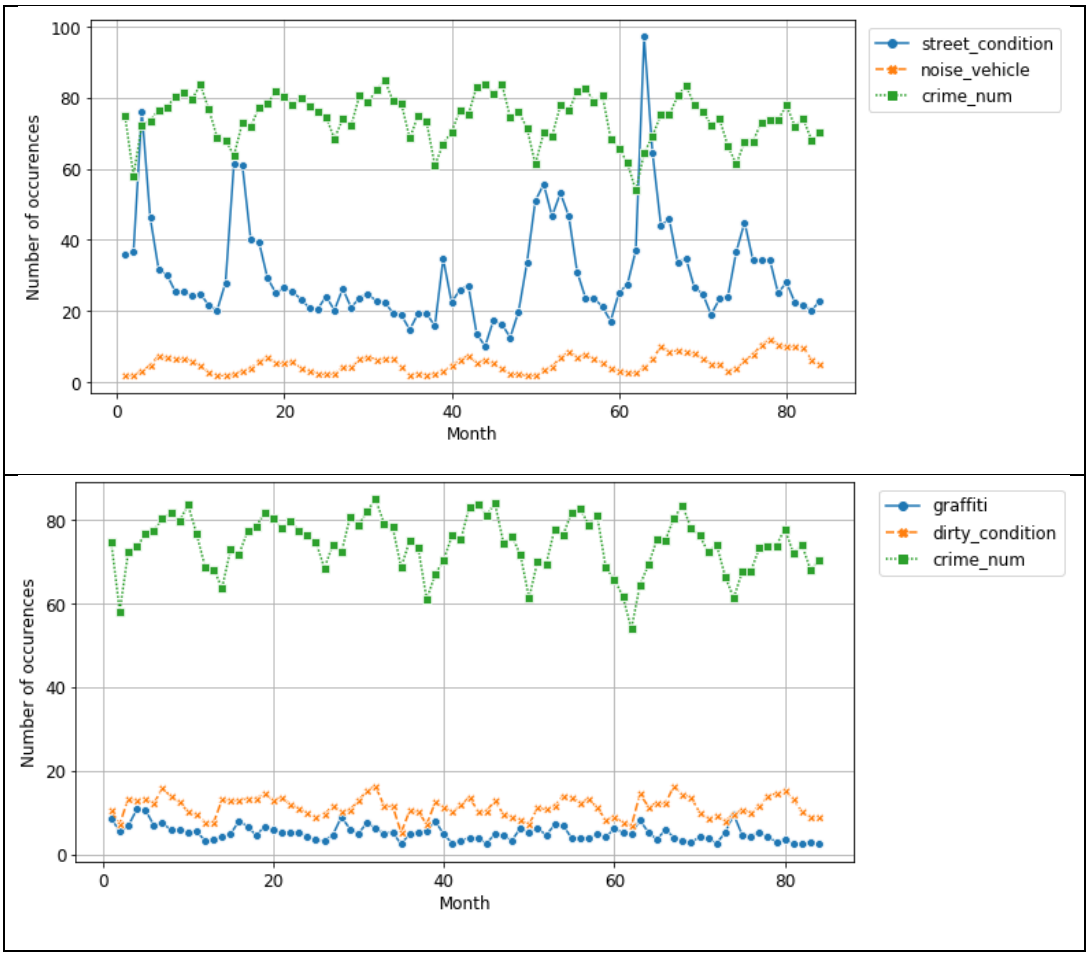


Figure 11. Data distribution of selected features and response from the year 2010 – 2016

3.5. SPATIAL PATTERN

Spatial autocorrelation analysis in this stage was performed to the response variable to get the whole picture of pattern in the datasets before the learning process takes places. The output will be used to consider the model performance. The degree of spatial autocorrelation on the response value in the monthly aggregated dataset was calculated using Global Moran's I. The result of I as shown in Figure 12, shows the degree of clustering is fluctuating through the time. Although it is fluctuating, the average value of Global Moran's I quiet high ranging from 50 – 57% of the area are clustered. Additionally, the Global Moran's I value of the response variable is different between monthly and weekly dataset. However, it can be concluded that there is a pattern in the dataset.

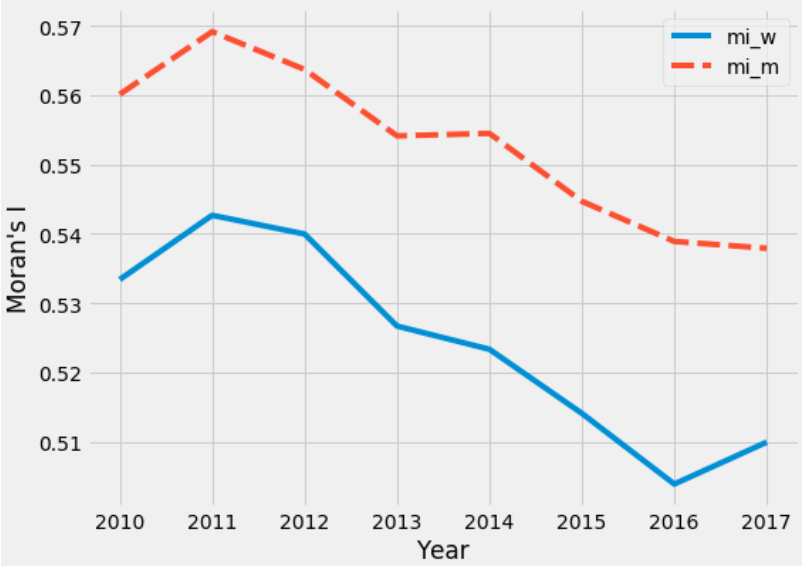


Figure 12. Global Moran's I value of response variable fluctuating through time, the blue line is Global Moran's I on the weekly dataset, while the red line is acquired on the monthly dataset

4. CASE STUDY: EXPERIMENTAL SET-UP

The subsections in this chapter give a detailed description of the experiments designed to develop the machine learning models. This includes a description of the cross-validation strategy and hyperparameter tuning. Moreover, the selected configuration to evaluate model performance and its ability to capture spatial patterns is also explained. Besides this, the hardware and software configuration to run the machine learning models are also described too.

In this research, we performed experiments on four machine learning algorithms using two different temporal scale datasets which are monthly and weekly datasets. These datasets contain explanatory and response variable. These features we obtained through feature engineering discussed in the previous chapter. The features contain four major features; namely complaints, temporal, lagged spatial features and zip code. Lagged spatial features contain spatial lag and LISA's quadrant.

The experiments divided into two big domains; namely no lag (NL) and with lagged spatial features (WL). NL experiment consists of complaint, temporal and zip code features. While, WL consists of complaints, temporal, lagged spatial features and zip code. Beside this, we also varied the experiment with combination of random and fixed features. We have already discussed and chosen features as random effects, fixed effects or both of them in the previous chapter. The overview of the detailed experiment as shown in Table 4.1. However, vanilla RF and SVR model only trained using fixed effects.

Table 4.1. Detail experiments to develop machine learning models

Experiments Code	Random/ Fixed Variable (RV/FV)	Features				
		Complaints	Temporal ($month$)	Spatial Lag ($t-1$)	LISA's Quadrant ($t-1$)	Zip Code
V+MNL/WNL	FV	●				(OHE)
V+MWL/WWL	FV	●		●	●	(OHE)
ME+MNL/WNL - 1	FV RV	●	●			C
ME+NL – 3, ..., 15*	…	…	…	…	…	…
ME+MWL/WWL – 10	FV RV	●	●	● ●	●	C
ME+MWL/WWL – 11	FV RV	●	●	●	●	C
ME+WL – 1, …, 24*	…	…	…	…	…	…

V = Vanilla machine learning regression model (R for RF and S for SVR*)

ME = Mixed effects + machine learning regression model (M for MERF and MS for MESVR*)

M/W = Monthly dataset / Weekly dataset

NL/WL = No Lagged / With Lagged features

C = Cluster

* = see appendix

4.1. MODELLING

Here, we explain the process followed to develop a model to predict the future values of a response variable based on several features. It is important to develop a good predictive model as the prediction model outcome can be used to take further action to reduce crime. Therefore, we need to tune up the model performance.

Model generalization often can be adjusted through hyperparameter tuning (Probst, Wright, & Boulesteix, 2018). There are several steps to tune up the model hyperparameter. Hyperparameter is parameters that cannot directly be learned during learning and have to be set to the machine learning before training the model. Hyperparameter tuning is used to find the best machine learning regression parameters

configuration. Furthermore, it relies on trial and error experiments to determine the best parameter setting through model evaluation using different parameter combinations. Although there are many methods to obtain the best parameters, grid search and randomized search method were selected used in this research. Because they are widely used (Bergstra, James & Bengio, 2012). The modelling configuration in the hyperparameter tuning can be seen in Table 4.2.

Table 4.2. Modelling configuration used in hyperparameter tuning

	Modelling Setup							
	RF		**MERF**		**SVR**		**MESVR**	
	Monthly	Weekly	Monthly	Weekly	Monthly	Weekly	Monthly	Weekly
Feature scaling					●	●	●	●
Subsampling								●
Randomize search	●	●			●	●	●	●
Grid search			●	●	●	●		
Zip Code [OHE]	●	●			●	●		
Zip Code [Cluster]			●	●			●	●
Group K-Fold	7	7	7	7	7	7	7	3
NL features	256	257	*	*	256	257	*	*
WL features	262	263	*	*	262	263	*	*

* the number of features is varied on random and fixed features combination.

Grid search is an expensive way to find the best parameters of the machine learning over pre-defined ranges for each parameter. Bergstra et al. (2012) proved that grid search is reliable in a problem with a relatively small number of parameters and become inefficient when the dimensionality of parameters increased. In contrast, randomized search is more efficient when working with high-dimensional parameter space. Thus, in this study, grid search was applied to datasets that have a small number of parameters.

Unlike grid search, randomized search uses a fixed number of parameter space, that is sampled from a specified uniform distribution (Zhigljavsky & Pintér, 1991). Moreover, randomized search is iterative through the number of parameters being sampled (Peck & Dhawan, 1995). Hence, there is a trade-off between the number of iteration and efficiency. However, both grid search and randomized search were used to find the best parameters.

In the modelling configuration, as shown in Table 4.2, the datasets were split seven folds in the training set and hold out a year in the test set except for MESVR trained with weekly dataset use three folds. In the training set consists of spatial structure in the form of multipolygon. Thus, to retain the structure in the data, we use group k-fold with a year as a grouping parameter to split training set in the cross-validation.

4.1.1. CROSS-VALIDATION

Cross-validation is an essential step in the training process of data-driven models (Cawley & Talbot, 2010). It helps to reduce overfitting by partitioning datasets into k-fold. Instead of putting all of the data into the training, cross-validation split datasets into two subsets; training subset and validation test as shown in Figure 13.

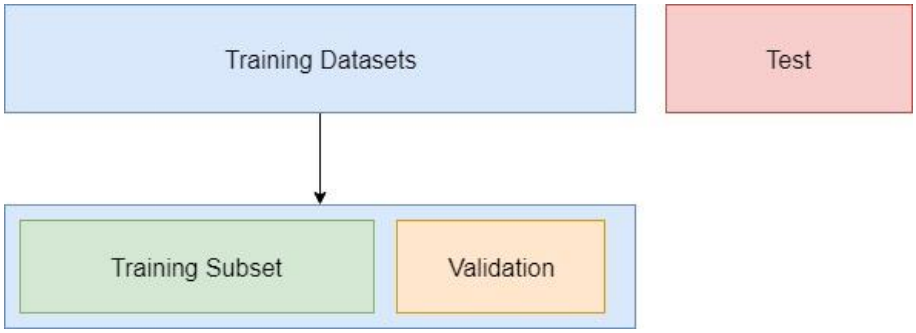


Figure 13. Cross-validation split the dataset into training, validation and test set

In the cross-validation, we were careful to choose the best method to split datasets, since we were working with longitudinal spatial datasets as shown in Figure 14. To retain spatial structure in the dataset, we picked group k-fold and use temporal slicing on year variable as grouping parameter split.

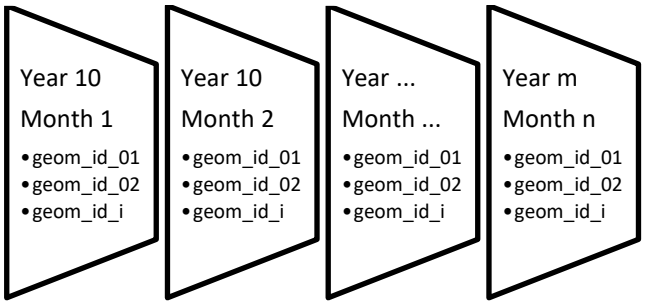


Figure 14. Structure of monthly scale spatial datasets used to develop machine learning model with m = 2010, …, 2016, n = 1, …, 12 and i = 1, …, 248

Group k-fold has a similar approach to leave one out group cv except it divides the dataset into k-fold. Moreover, it also does not use a similar group twice in two different folds as shown in Figure 15. Using this approach, we split the training set into seven folds.

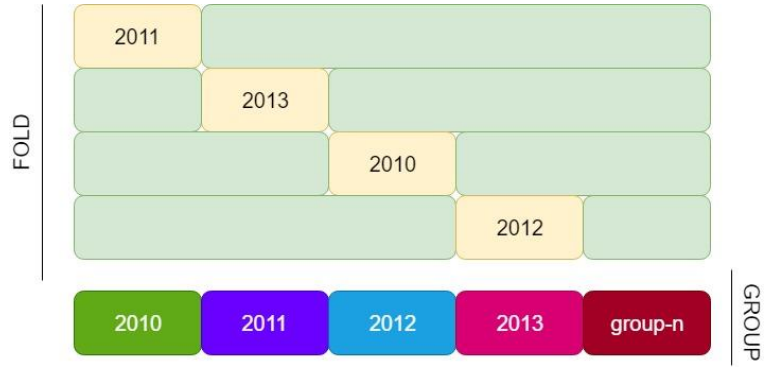


Figure 15. Illustration of group k-fold, the yellowish block is validation test set.

4.1.2. RANDOM FOREST

RF has feature selection ability through its feature importance. As we already discussed in chapter 3, in the training process, the random forest grows the number of trees. In each tree has several nodes as each node contain features and the split the dataset into two which have the same response value. Thus, it is possible that RF to calculate the feature importance from how much of each feature decreases the variance in a tree. Feature importance can be used to further reduce data dimensionality without affecting the model performance.

Features Importance

As can be seen in Table 4.1 and Table 4.2 we have different the number of features for a different domain (with lagged spatial features and without lagged) and temporal scale dataset. It translates four different experiments. Training set on monthly dataset has 256 features for without lagged and 262 features for without lagged spatial features. As for weekly dataset, we have 257 and 263 features for without lagged and with lagged domain respectively. Using the default RF parameter, the model trained using without lagged features; the average accuracy measured using r-squared obtained are 0.91 and 0.86 for monthly and weekly dataset respectively using without lagged features. These results can be seen in Figure 16 and Figure 17.

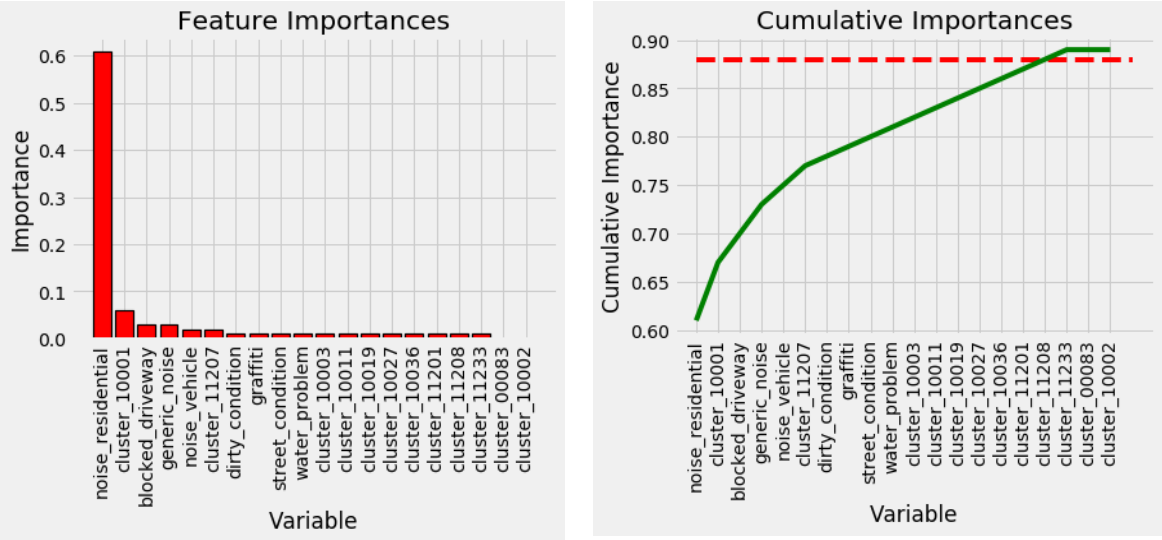


Figure 16. Feature importance result and cumulative importance on the monthly dataset

The relative differences in the feature importance as shown in Figure 16, noise residential has the most influence compared with others. Interesting result, there are two clusters as features that have significant importance. The feature selection process also considers the contribution of each variable to the overall importance. The cumulative importance graph as shown in Figure 16 might help to cut off the unimportant features. The red dashed line was drawn at 93% of total cumulative importance in the monthly dataset to select the most important features. As a result, month features have been dropped from the training set.

Feature selection for weekly scale dataset has a similar result with feature importance for monthly dataset except for the threshold to remove the unimportant feature. The upper threshold can be acquired to filter unimportant is around 88% as shown in Figure 17. The threshold that was used to filter features is an arbitrary threshold. This means we can adjust the value of the threshold when the result is not good.

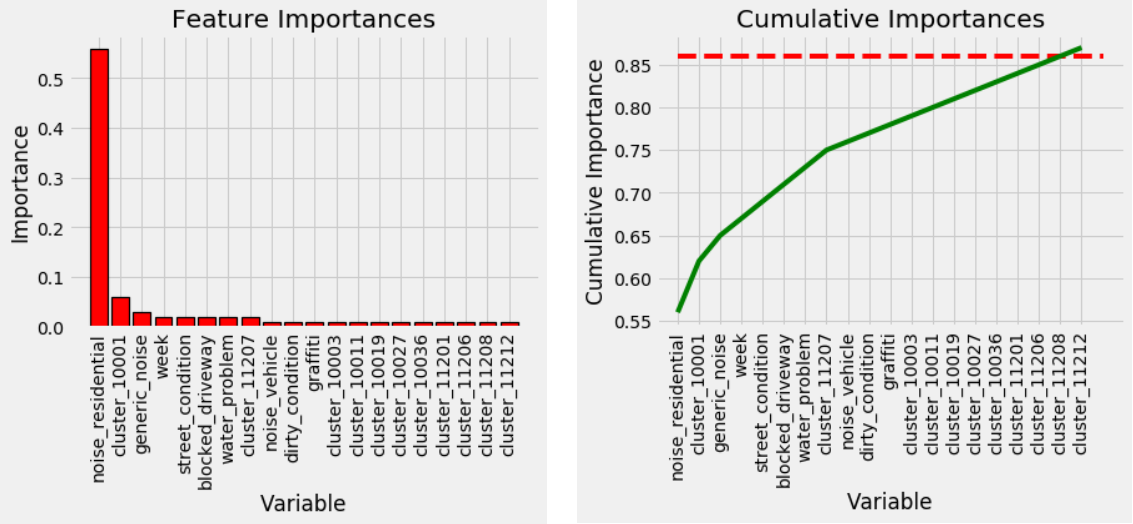


Figure 17. Feature importance and cumulative importance on the weekly dataset

An interesting observation to feature importance was when all features included in the training set to both weekly and monthly set. As it can be seen in Figure 18, lagged features become dominant and make the other features irrelevant. It has a strong correlation with response variable that could indicate serial correlation. Moreover, the accuracy of r-squared score goes up significantly by 0.95.

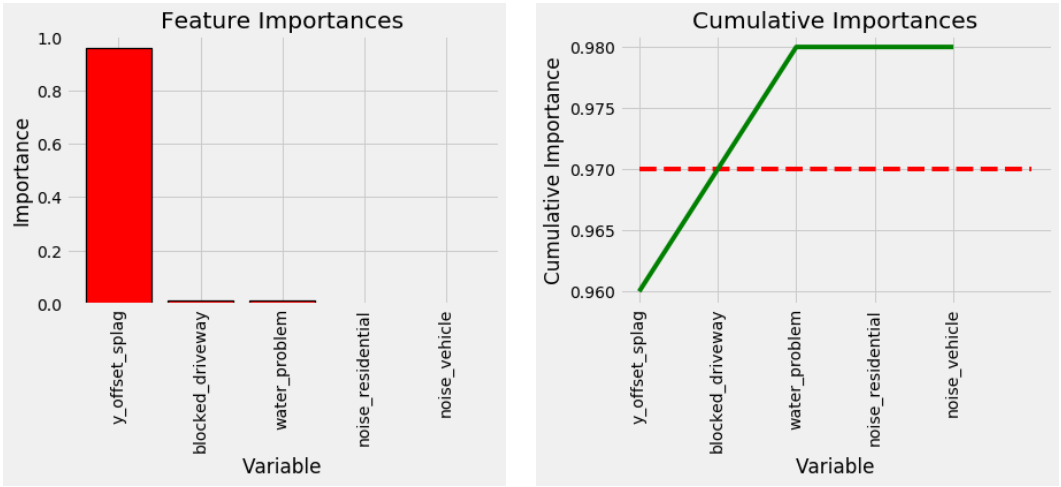


Figure 18. Feature importance and cumulative importance result when lagged features were included

Hyperparameter Tuning

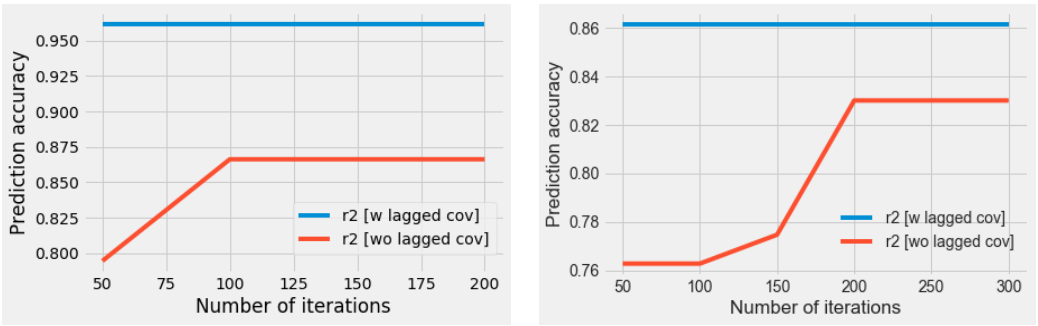
Recall in chapter 2; there are four RF parameters to tune up; the number of trees, maximum tree depth, minimum samples split and leaf. Randomized search was accomplished to each different temporal scale datasets. Parameter distribution grid configuration as shown in Table 4.3 Randomized search configuration was used to find the best RF parameters. It runs iteratively to find the best configuration. The number iteration was set to 10, 25, 50, 100 and 200. Group k-fold seven folds were used to tune the parameters.

Hyperparameter tuning is close to trial and error to find the best parameter. For weekly data, we did the second round of iteration with a higher number of number estimators and adding iterations because the results were not good enough.

Table 4.3 Randomized search configuration was used to find the best RF parameters

Hyperparameter Tuning		
	Monthly	**Weekly**
Parameters	**Tuning**	**Tuning**
n_estimators	random integer (50 – 500)	random integer (100 – 1000)
max_depth	random integer (10 – 200)	random integer (10 – 200)
min_samples_split	random integer (2 – 50)	random integer (2 – 50)
min_samples_leaf	random integer (5 – 100)	random integer (5 – 100)

Randomized search found the best configuration parameter on a different number of iterations. As can be seen in Figure 19, the best parameter configuration found in a different number of iterations. For monthly dataset, the model trained using non-lagged spatial features the best parameter found for 100 iterations, as for weekly data found for 200 iterations. Both different datasets using lagged spatial features found for 50 iterations. The optimum RF parameter from randomized search as shown in Table 4.4.



(A) (B)

Figure 19. (A) Hyperparameter tuning using vanilla RF on monthly dataset (B) weekly dataset. The blue line, the model trained with lagged spatial features, while the red line without lagged spatial features.

Table 4.4. Optimum RF parameter configuration for both monthly and weekly scale dataset

Optimum RF Parameter Configuration				
	NL		**WL**	
Parameter	**Monthly**	**Weekly**	**Monthly**	**Weekly**
n_estimators	450	552	279	707
max_features	all features	all features	all features	all features
max_depth	36	140	194	128
min_samples_split	3	27	23	41
min_samples_leaf	11	9	5	6

4.1.3. SUPPORT VECTOR REGRESSION

Modelling with SVR were performed using RBF kernel. RBF kernel is based on distance. Recall equation in 2.14, the features that have a large range of value will dominate in the computation of kernel matrices. Hence, it is required to scaling the features values to get better model generalization (Valenzuela, Zhang, & Selpi, 2017). In SVR modelling, scaled datasets were used to train the model. Beside this, we found that feeding the model in the hyperparameter tuning with the whole training subset and validation set on weekly dataset was very expensive. This because SVR solves the problem in quadratic order. SVR training becomes expensive as the size and the dimension of the data are increased. Hence, we decided to subsample dataset become smaller size.

Subsampling Training Set

Good approach to subsampling dataset is by randomly sampling. However, we did not randomly subsample the dataset as we consider spatial structure in the dataset as shown in Figure 14, data distribution and its pattern through time as shown in Figure 20 and the degree of spatial autocorrelation of the response variable as shown in Figure 12. We decided to subsample the data from 2012 – 2014. This dataset was used to tune hyperparameter and retrain the final model for SVR and MESVR.

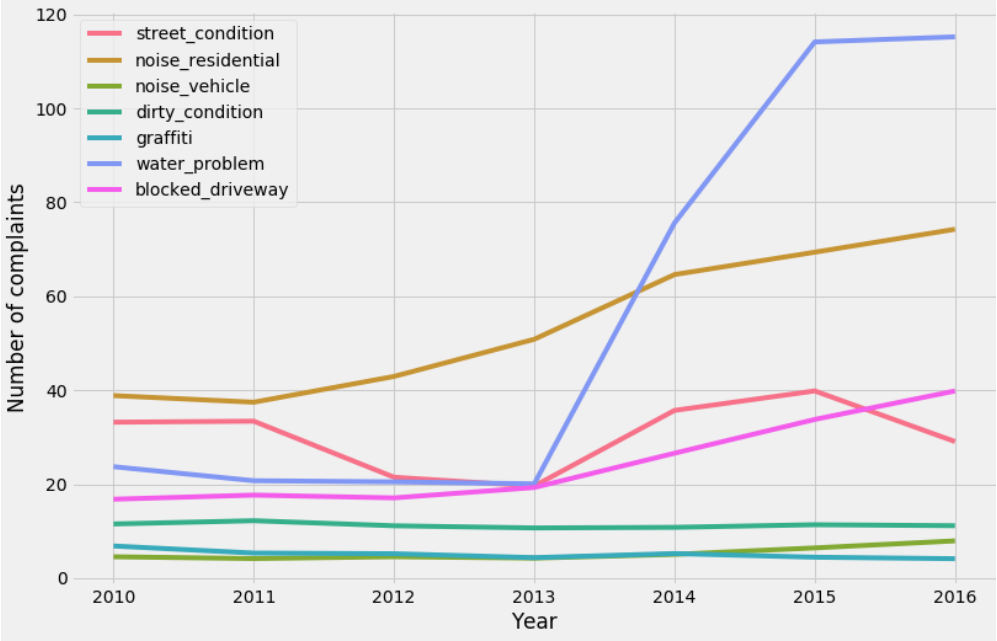


Figure 20. Line chart showing the series of the data distribution of each complaint feature

Hyperparameter Tuning

Apart from randomized search, we also used grid search to find the best estimate of SVR parameters. There are three parameters used to tune up the model; gamma, c and epsilon value. Similar treatment with randomized search on RF, it was also performed to two different scales of datasets. Slightly different with RF, we used all the features including lagged spatial features in the hyperparameter tuning. Thus, the parameter setting for non-lagged features to train the model was using the same set as lagged spatial features. The configuration of parameter distribution used to find the best parameter with randomized search can be seen in Table 4.5. The number of iterations was set to 10, 25, 50, 100, 150 and 200.

Table 4.5. Parameter distribution configuration to find optimal SVR parameters.

Hyperparameter Tuning		
	Monthly	**Weekly**
Parameters	**Configuration**	**Configuration**
kernel	RBF	RBF
gamma	random (1-100)	random (1-100)
c	random (1- 100)	random (1- 100)
epsilon	random (0.1-10)	random (1-100)

The range of C and epsilon parameter was chosen according to Cherkassky & Ma (2004). Range C parameter was obtained from:

$$C = \max(\,|\bar{y} + 3\sigma_y|, |\bar{y} - 3\sigma_y\,|\,)$$

where $\bar{y}$ is mean of the response variable, σ_y is the standard deviation of resthe ponse variable. The range for epsilon when the number of samples is large obtained by

$$\varepsilon = \tau\sigma\sqrt{\frac{\ln n}{n}}$$

where n is the number of samples, the value of τ is constant and Cherkassky & Ma (2004) proposed 3 for good estimation. The optimum parameter of SVR using monthly datasets obtained using 150 iterations of randomized search as shown in **Error! Reference source not found.**.

However, further observation to the model measured with r-squared using weekly dataset using all features has low prediction accuracy, about 0.60. Then, we further tuned up the parameter using grid search using the best randomized search parameter as a baseline. The grid parameter as shown in

Table 4.6, we add c value 1000 and gamma value 0.001 to the distribution parameter grid. Using this grid configuration, the model gains prediction accuracy significantly with more than 20% with r-2 squared measured 0.82. The optimum SVR parameter can be seen in Table 4.7

Table 4.6. Grid distribution parameter used to find best SVR parameter for weekly dataset

Hyperparameter Tuning	
Parameter	**Configuration**
kernel	RBF
gamma	[0.001, 0.01]
c	[100, 1000.0]
epsilon	0.1

Table 4.7. Optimum SVR parameter configuration

Optimum SVR Parameter Configuration		
Parameter	**Monthly scale**	**Weekly scale**
gamma	0.1	0.001
c	100	1000
epsilon	10	0.1

4.1.4. MIXED EFFECTS RANDOM FOREST

The tuning on MERF's hyperparameters was performed using grid search. Unlike vanilla RF, there are two rounds to tune up the model performance. To start with, we tuned the MERF parameter and secondly, we tuned up the model through the combination of random and fixed features.

MERF uses out of bag prediction and sub-sample dataset to find an optimum predicted response. We observed that by default only two parameters that can be tuned; namely the number of estimator and iterations. Hence, we use grid search to find the best parameters of MERF. All of the features excluding lagged spatial features were used to tune the MERF's parameter.

Using configuration as shown in Table 4.8, translates 25 combination fits per fold and renders 175 fits.

Table 4.8. Hyperparameter tuning configuration to find optimum MERF parameters

MERF Parameter Tuning Configuration	
Parameters	**Configuration**
n_estimators	[50, 100, 150, 200, 250]
n_iterations	[50, 100, 150, 200, 250]
max_features	n_features

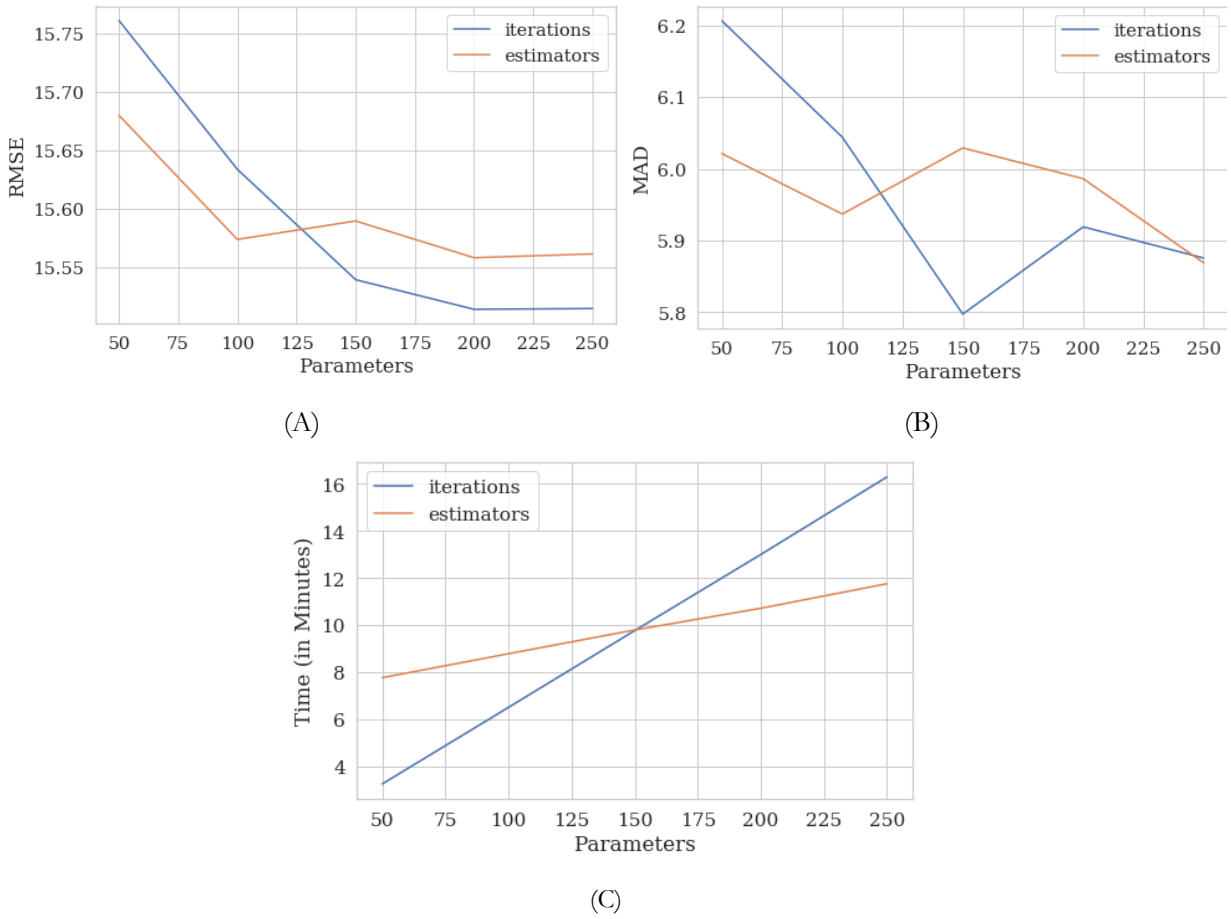
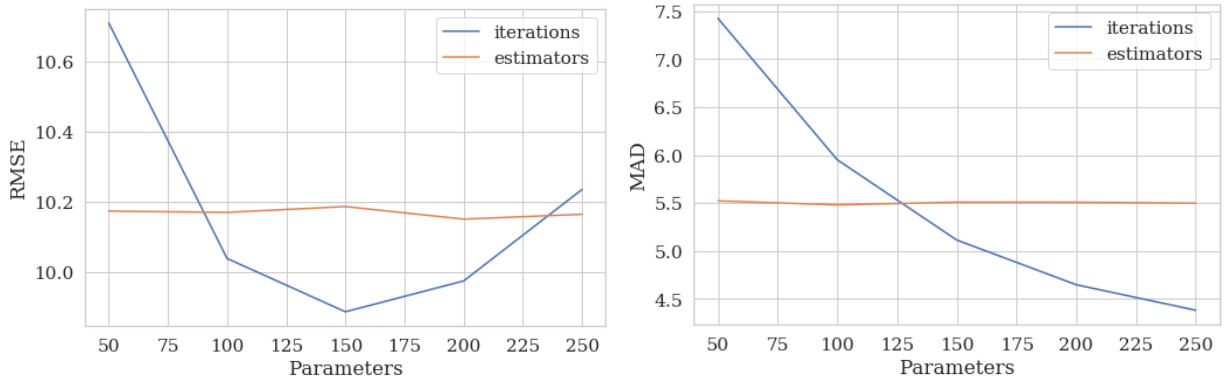


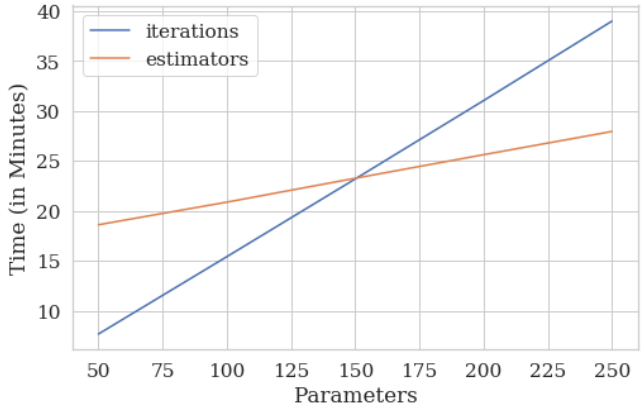
Figure 21. Parameter tuning result on MERF with the monthly dataset (A) measured using RMSE, (B) measured using MAD, (C) computation required to train the model.

As it can be seen in Figure 21, using root mean squared error (RMSE) and median absolute deviation (MAD) as metric evaluation, the number of iterations is convergence when it reached 200 iterations. It is also similar to the number of estimators. The selection of parameter configuration also considers the computation time required to train the model. Thus, the configuration parameter for the monthly dataset is using 200 for the number of iterations and estimators.

As for the weekly dataset, as shown in Figure 22, the best configuration for the number of iterations is 150 while the number of estimators is 100. As the model performance already reached its peak. Thus, adding more estimator does not change the performance of the model. These configurations were used to tune up the model in the second round.



(A) (B)

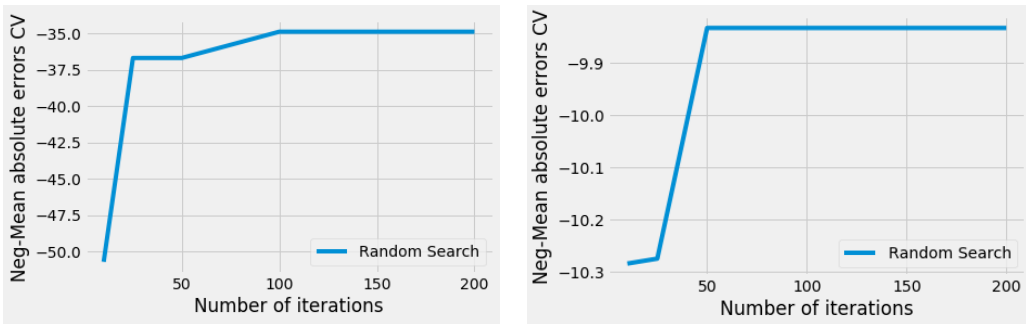


(C)

Figure 22. Parameter tuning result on MERF using weekly dataset, (A) measured using RMSE (B) measured using MAD (C) computation time

4.1.5. MIXED EFFECTS SUPPORT VECTOR REGRESSION

The tuning on MESVR's parameters is similar to MERF. Slightly different from MERF, we can place the best configuration parameter of fixed effects estimator, which is SVR. Therefore, in the first round, we performed parameter tuning using randomized search. Randomized search was performed using complaint and lagged spatial features since the zip code as spatial representation in the data placed to the cluster. Randomized search parameter distribution was using the similar set as vanilla SVR as shown in Table 4.5.



(A) (B)

Figure 23. Tuning SVR parameter using randomized search using (A) monthly dataset and (B) weekly dataset

As it can be seen in Figure 23, using randomized search, the best parameter configuration of SVR trained using monthly dataset found for 100 iterations and 50 iterations for SVR parameter trained using weekly dataset. The optimum SVR parameter can be seen in Table 4.9. These optimum parameters were used to further tune up the model through random and fixed features combinations.

Table 4.9. Optimum SVR parameter as fixed effects function in MESVR

Optimum SVR Parameter Configuration		
Parameter	**Monthly scale**	**Weekly scale**
gamma	0.7	7.3
c	73.8	8.2
epsilon	20	0.02

4.2. MODEL EVALUATION

Optimum parameter configuration for each machine learning regression algorithm has been used to retrain the model. Retraining the model using fully training set was performed to acquire the optimized model. The optimized model then evaluated using testing set in term of predictive performance and ability to capture spatial pattern. Moreover, to evaluate the ability to predict and capture spatial pattern on unseen data, the model re-trained using training and test set as a new piece of information to obtain a final model. These models will be used to predict crime in New York City in the year 2017.

There are five kinds of metrics evaluation to evaluate the model. Firstly, model prediction performance evaluation metrics will use mean absolute error (MAE) since mean absolute errors insensitive to outliers and weighted equally (C. Chen, Yan, Zhao, Guo, & Liu, 2017; Roy & Larocque, 2012). Mean absolute error formula:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Secondly, the metric was used to evaluate the model prediction error is the median absolute deviation (MAD). This metrics also robust to outliers by taking the median of all absolute errors of the residual's regression given i samples. Mathematically, it is calculated using this formula:

$$MAD\ (y, \hat{y}) = median(|y_i - \hat{y}_i|)$$

These two metrics were used to evaluate RF models and their mixed effects counterparts since random forest robust to outliers. Whereas, SVR and MESVR models were evaluated using root mean squared errors (RMSE) since these algorithms do not make fully resistant to outliers in the data. However, both MERF and MESVR were evaluated using all of the metrics mentioned here. RMSE is formulated given by:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Such that, RMSE penalizes the magnitude of errors higher than MAE. Thus, the value of RMSE will be higher or equal to MAE. To evaluate the prediction accuracy, R^2 was selected.

It is formulated given by:

$$\hat{R}^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$$

The last metric is Moran's I. It was used to evaluate the model ability to capture spatial patterns. The equation of Moran's I can be found in chapter 2 at equation 2.3. Additionally, computation time during training the model also assessed.

4.3. HARDWARE AND SOFTWARE

This subsection gives a brief description on hardware and software were used in the experiments.

Software

Standard (also known as "vanilla") machine learning regression algorithms, RF and SVR are provided by Scikit-learn (Pedregosa et al., 2011).[1] One of the most popular machine learning libraries in Python. The MERF algorithm was also available as a python package was provided by Hajjem et al. (2014).[2]

QGIS was used to visualize the base map and the zip code map. The PostgreSQL/SQL database container and the PostGIS extension were used to store and pre-process datasets. Psycopg version 2 was used to connect Python to the database.

Source code development was done using Python language with Jupyter Notebook as an Integrated Development Environment (IDE). The PySal library was used to process geospatial data along with GeoPandas. An extension of Pandas that allows spatial operation on geometric types.

High Performance Computing (HPC) environment using job scheduler and use qsub utility to send the batch job queue of model training to computing nodes.

Hardware

The RF and SVR models were run using all of the resources excluding GPU since there is no API nor middleware in Scikit-learn to split and distribute the processes to an external resource such as GPU. Therefore, the learning processes of the models are only made use of the raw power available CPU. It was parallelly distributed the processes into physical and logical cores. It also uses a large amount of physical RAM as temporary storage of the result set of models.

Two types of were used in this thesis; PC and High-Performance Computing (HPC). LIPI HPC facilities were used to run a task that needs heavy computation such as MERF and MESVR models.[3] The HPC and PC configurations are described in Table 4.10.

Table 4.10. PC and HPC configuration were used to train the model

Hardware Configuration		
	PC	**HPC (1 Node)**
CPU	AMD x64 8 cores / 16 threads	2 x Intel Xeon E5-2650 2.00 GHz (16 Cores)
HDD	4 TB	100 GB
RAM	DDR 4 32 GB	128 GB RAM

[1] https://scikit-learn.org/stable/index.html
[2] https://github.com/manifoldai/merf
[3] http://grid.lipi.go.id

5. CASE STUDY: RESULTS AND DISCUSSION

The cross-validation and model generalization results are presented in this chapter. These models were evaluated using the metrics explained in chapter 4. These metrics were used to pick the best model in the cross-validation. The best configuration of a selected model of each algorithm was used to build the final model and to predict test set (known as "unseen dataset"). The model performances that were trained from the various time scale dataset and features configuration on both model selection and final model performance are compared and explained in this subsection.

5.1. CROSS-VALIDATION

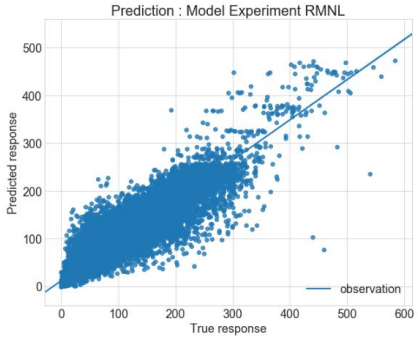
The results of the cross-validation model are presented, compared and evaluated to select the best model in term predictive performance and ability to capture spatial pattern. Two kind approaches to evaluate the vanilla machine learning regression models and their mixed effects counterparts. Vanilla model evaluations were performed by checking model performance in each split. Mixed effects model were evaluated by checking the performance of all the experiments (see Appendix I for the detail).

The models were trained using seven and three group k-folds (MESVR on the weekly dataset) to avoid overlapping and retain the spatial structure. As for random features, we use five random features; a) two features from both complaints features, b) two lagged spatial features and c) temporal features. To discover the influence of lagged spatial features to model performance and SAC residuals level, we split the cross-validation process into lagged and non-lagged model domains. Hence, in the cross-validation stage, we have 15 and 24 unique combinations of non-lagged and lagged models respectively performed at two temporal scales.

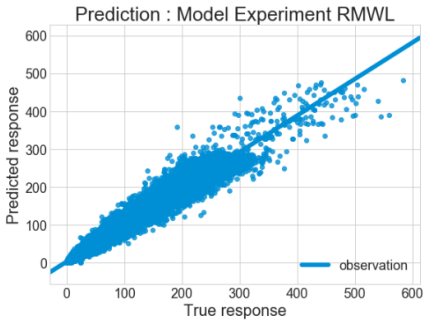
However, both approaches compare the model prediction performance with the degree of spatial autocorrelation in the regression residuals. The aim is to discover not only the best predictive model but also the lowest SAC level in the residuals.

5.1.1. RANDOM FOREST

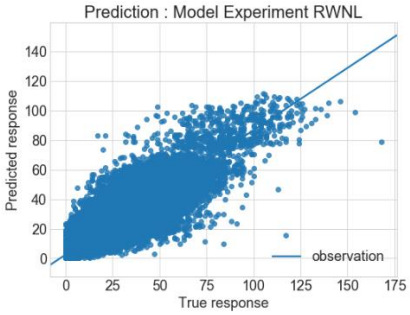
As it can be seen in Figure 24, the model trained using lagged spatial features has better predictive performance compared trained using only complaints variable. The points are converging close to the regression line. This means that the model might have the lowest standard deviation compared with others. Moreover, recall that lagged spatial feature; namely spatial lag has the highest feature importance might due seasonality in the response variable (see chapter 3 in the features exploration). In regression, the difference between true value and predicted values is known as regression residual. These model residuals were evaluated in term of SAC in each cross-validation split.



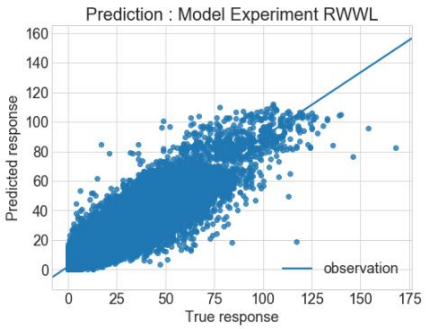
(A) The model trained with monthly scale dataset without lagged features



(B) The model trained with monthly scale dataset with lagged features



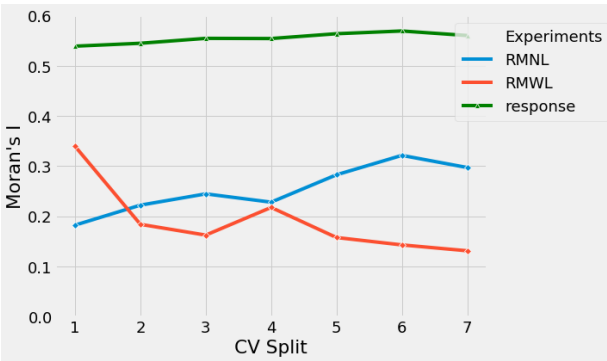
(C) The model trained with weekly scale dataset without lagged features



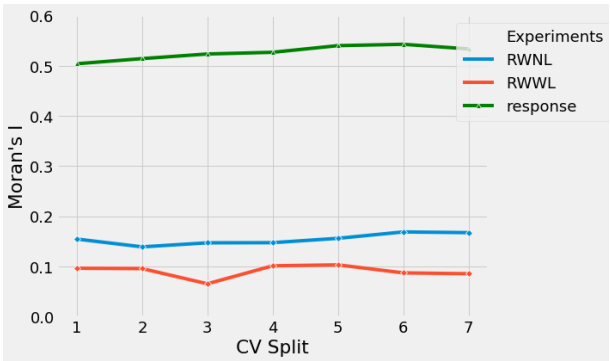
(D) The model trained with weekly scale dataset with lagged features

Figure 24. The prediction errors of each experiments using vanilla random forest to various scale dataset and feature configuration shows (B) has better prediction accuracy compared with the others.

The SAC of the response variable weekly set is slightly lower than the monthly set around 0.03 as shown in Figure 12 and Figure 25. This could be the weekly datasets have finer resolution than monthly. From Figure 25, we can infer that model trained using monthly scale dataset without lagged features as input features have the highest SAC in the regression residuals.



(A)



(B)

Figure 25. SAC of regression residual on each RF model in cross-validation stage (A) trained with the monthly set (B) trained with the weekly set.

The model prediction accuracy was evaluated using r-squared metric. It can be seen in Figure 26; model RMWL has higher accuracy of around 0.96 compared with the others. From Figure 26, we can also infer that there is a negative correlation between SAC and model prediction accuracy for model RMWL and RWWL. Conversely with model RMNL and RWNL has a positive correlation. This means that when the predictive performance of the model is high, the SAC value of the regression residuals also high. This could be an issue in the regression since the errors of regression should be independent and identically distributed over all region. However, the model trained using weekly dataset as in RMWL and RWWL experiments the SAC residuals are relatively constant. We can also infer from Figure 26, as for model trained using lagged spatial features has lower SAC residuals.

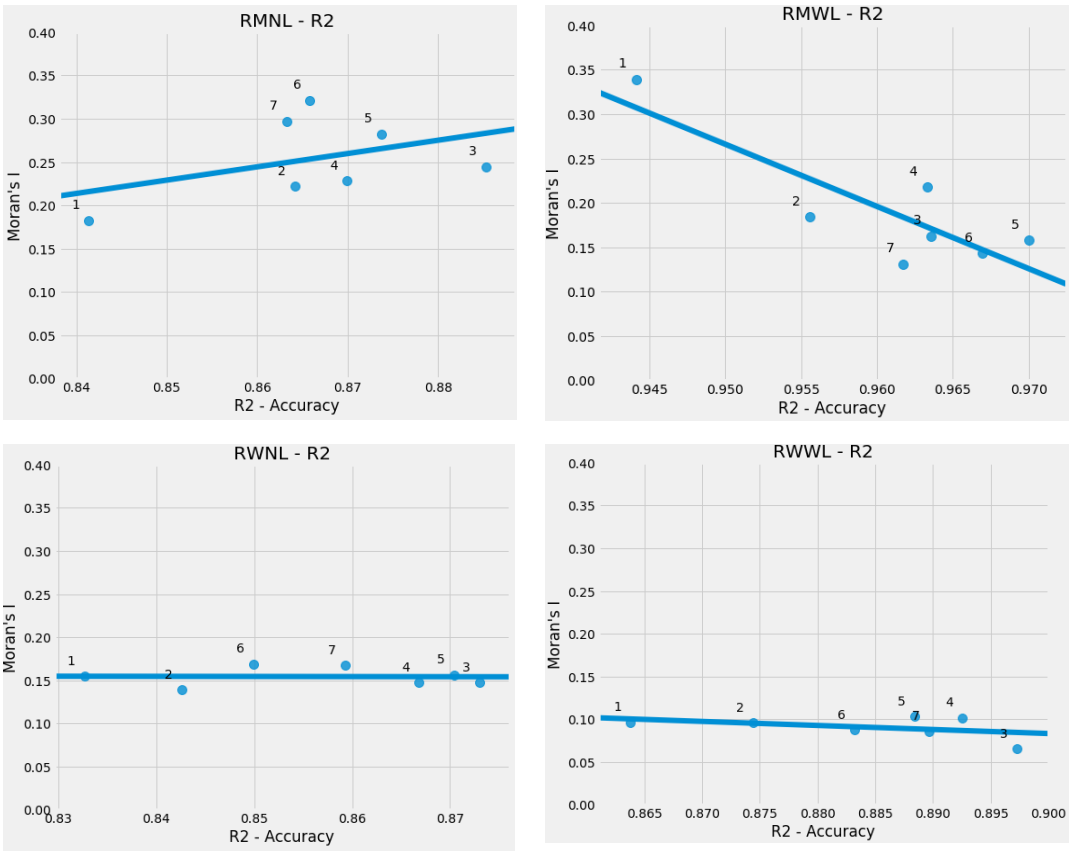


Figure 26. R-squared of RF experiments are compared. The blue line is a regression line to estimate relationship between r-squared and SAC residuals. The number 1 until 7 is cross-validation split.

To evaluate the magnitude of model prediction errors, mean absolute errors (MAE) and median absolute deviation (MAD) were performed to the model. Model prediction errors measured using MAE are lower as the magnitude of SAC of regression residual becomes higher for model RMNL as shown in Figure 27. This result is coherent with r-squared evaluation. However, using MAD, as shown in Figure 28, the correlation between the magnitude of errors and the degree of SAC of the regression residuals for model RMWL becomes positive. As it can be seen in Figure 27, the values of errors measured using MAE in split six, seven, four and five are shifted to the left in MAD in Figure 28. This means the distribution of model errors in these splits are positively skewed.

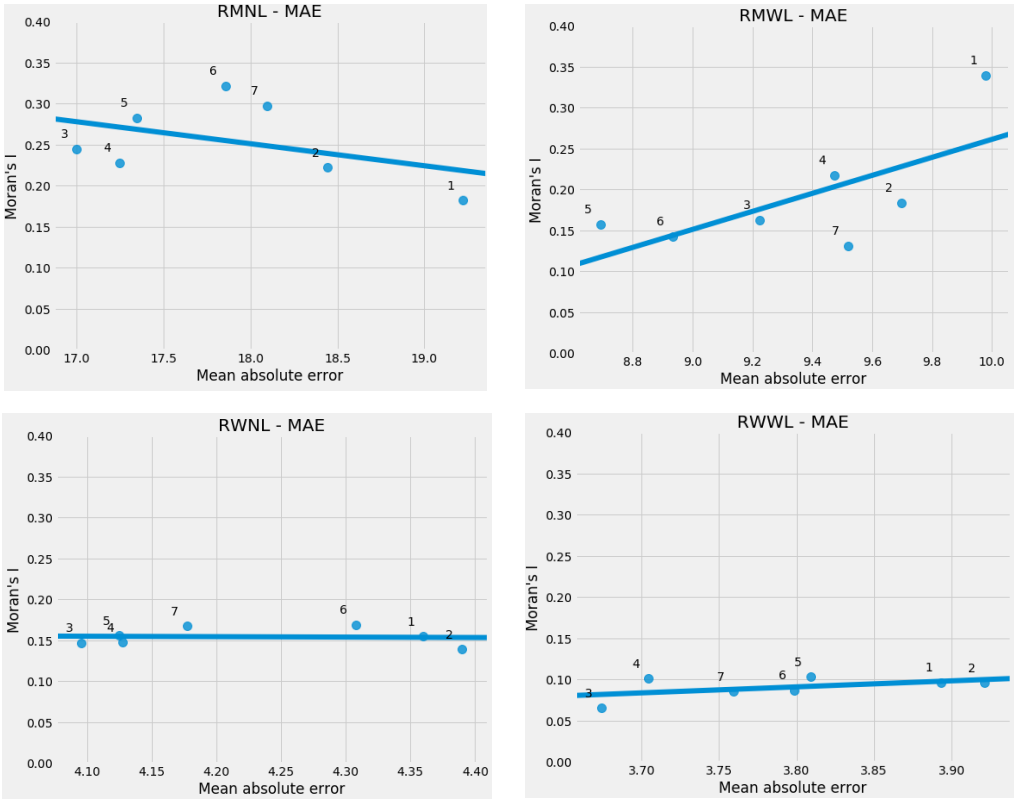


Figure 27. Equally weighted of average model errors of RF experiments. The blue line is a regression line to estimate the relationship between MAE and SAC residuals.

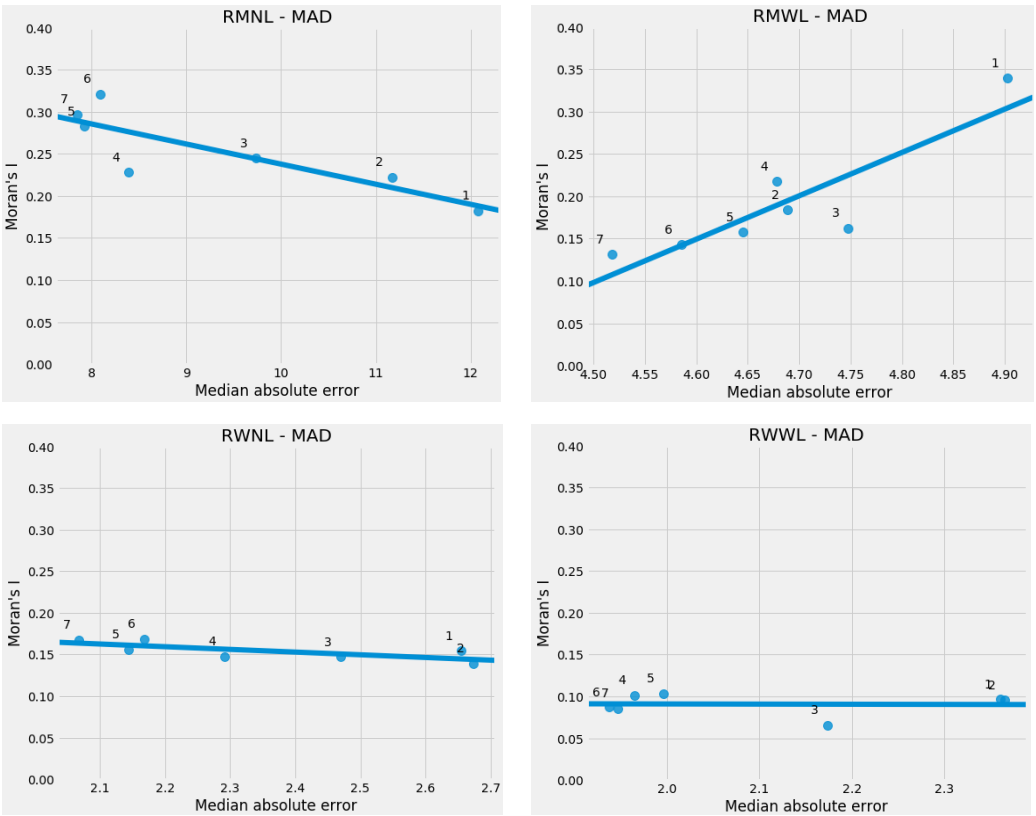


Figure 28. Comparison of prediction errors RF models using MAD. The blue line is a regression line to estimate the relationship between MAD and SAC residuals.

To summarize, from the various metrics evaluation, we can infer that the model trained using weekly data has lower SAC of the regression residuals level compared with their counterparts despite the lower prediction accuracy. The RF model trained using geographical features as geographical proximity in the predictors, in this case, zip code still prone to induce SAC in the residual regression. Therefore, consider these metrics evaluation, we opted to build the final model using lagged features and evaluate the final model on unseen data to both different scale datasets.

5.1.2. SUPPORT VECTOR REGRESSION

There are several approaches to evaluate predictive model performance. To begin with, we use scatterplots as shown in Figure 29 to give an overview of the model prediction performance. From these images, we observe that SMNL and SMWL model have better predictive performance than their weekly counterparts. We can see that most of the prediction points on both models are close to the best line fit.

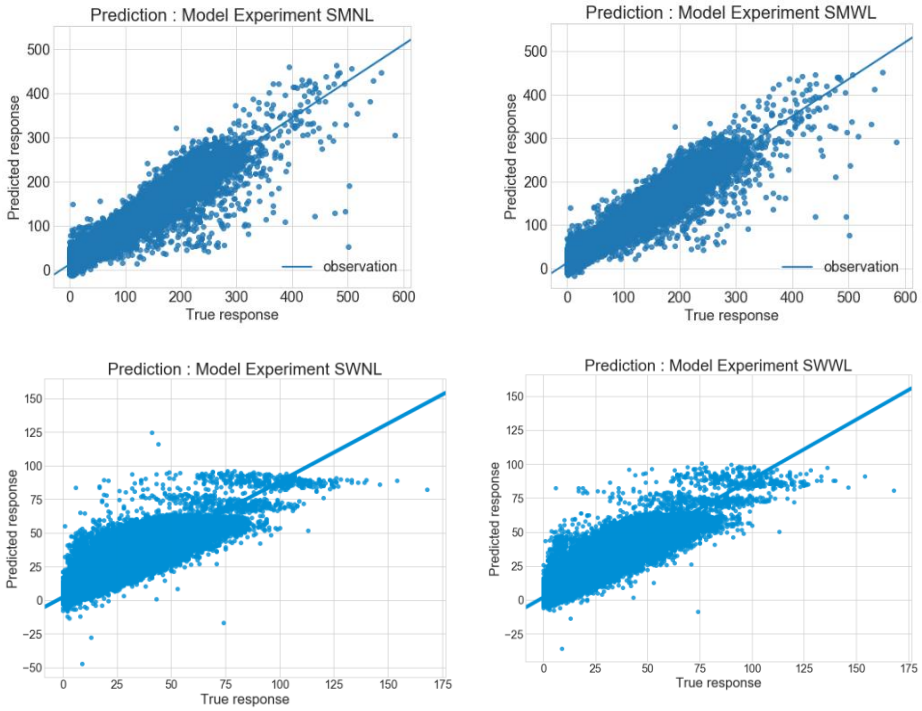


Figure 29. Scatterplots the predicted value and true response of vanilla SVR experiments.

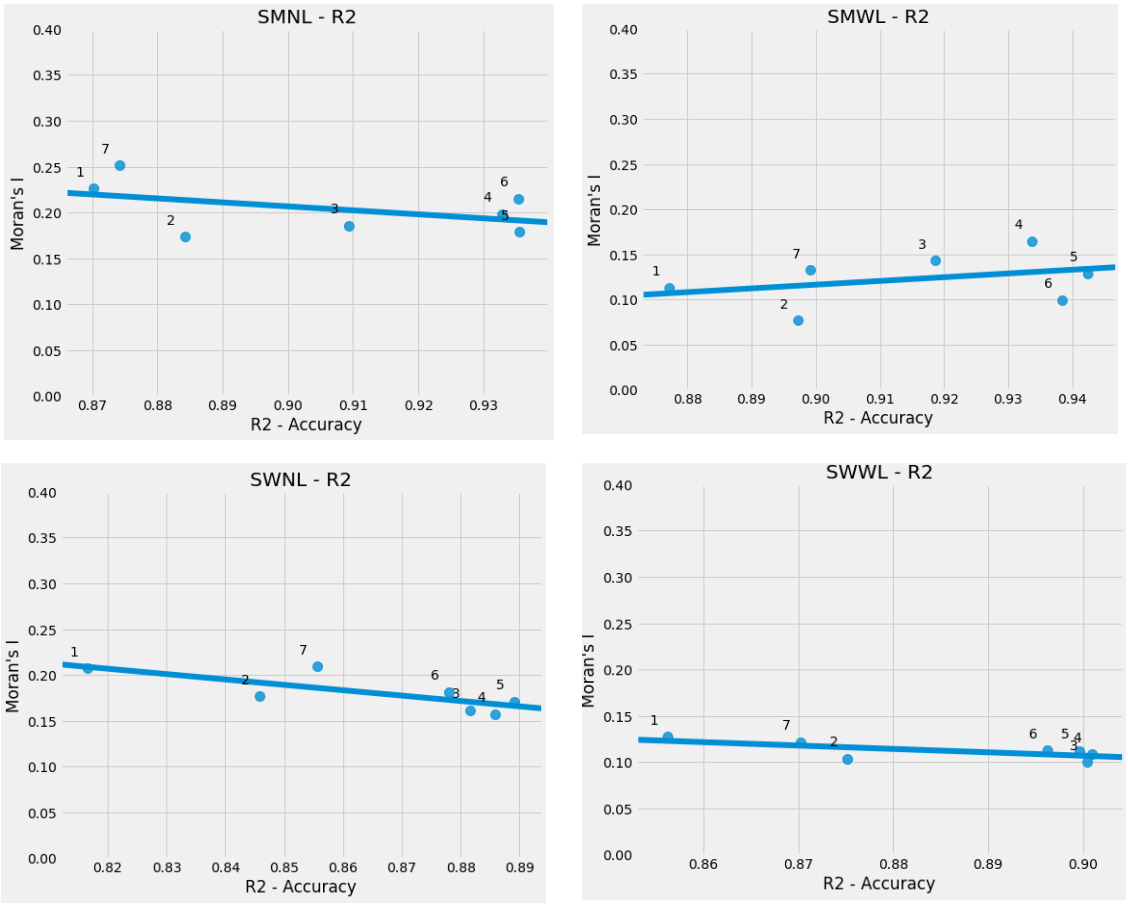
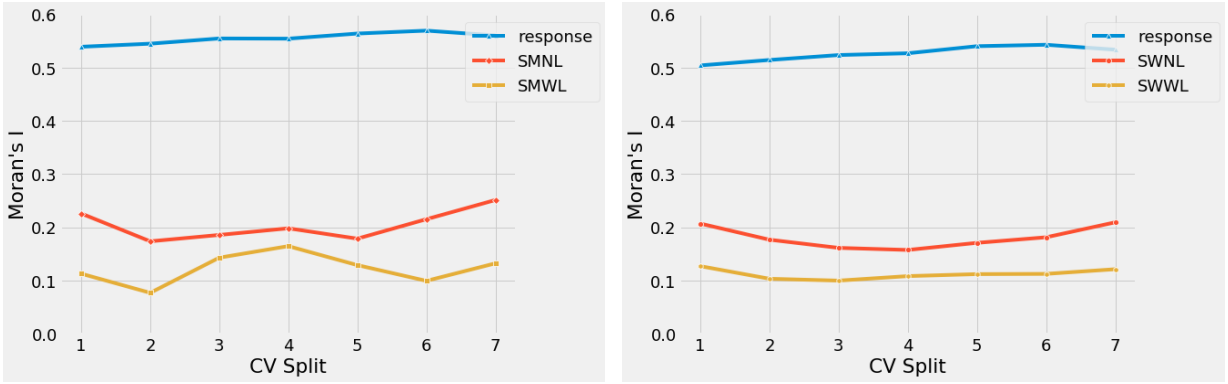


Figure 30. Prediction accuracy measured using r-squared to SVR models and compared. The blue line is a regression line to estimate the relationship between r-squared and SAC residuals.

The model prediction accuracy evaluated using r-squared as shown in Figure 30, reinforce the previous observation that the model SMNL and SMWL have relatively good prediction accuracy around 0.91 and 0.92 respectively. However, the SAC residual on SMWL experiment is slightly increased as the prediction accuracy getting higher. As for the model trained using weekly dataset, SWNL and SWWL, the SAC residuals are slightly lower than their counterparts as the baseline of SAC response also slightly lower. Still, the model trained using lagged spatial features has lower SAC residuals.

Overall, the model performance trained using lagged spatial features has relatively lower SAC residuals as shown in Figure 31. Moreover, there is a similar pattern of SAC regression residuals level to the models were trained with lagged spatial features that lower the magnitude of SAC residuals regression.



(A) (B)

Figure 31. Comparison of SAC of regression residual on SVR experiments (A) trained with the monthly set (B) trained with the weekly set.

The model prediction errors were evaluated using RMSE and show that the model trained using lagged spatial features has a close gap and overlap with the model trained without lagged spatial features as shown in Figure 32. Based on this finding, training the model using SVR with lagged spatial features does not significantly improve the model prediction performance directly, but it may indirectly improve to lower the magnitude of SAC residuals.

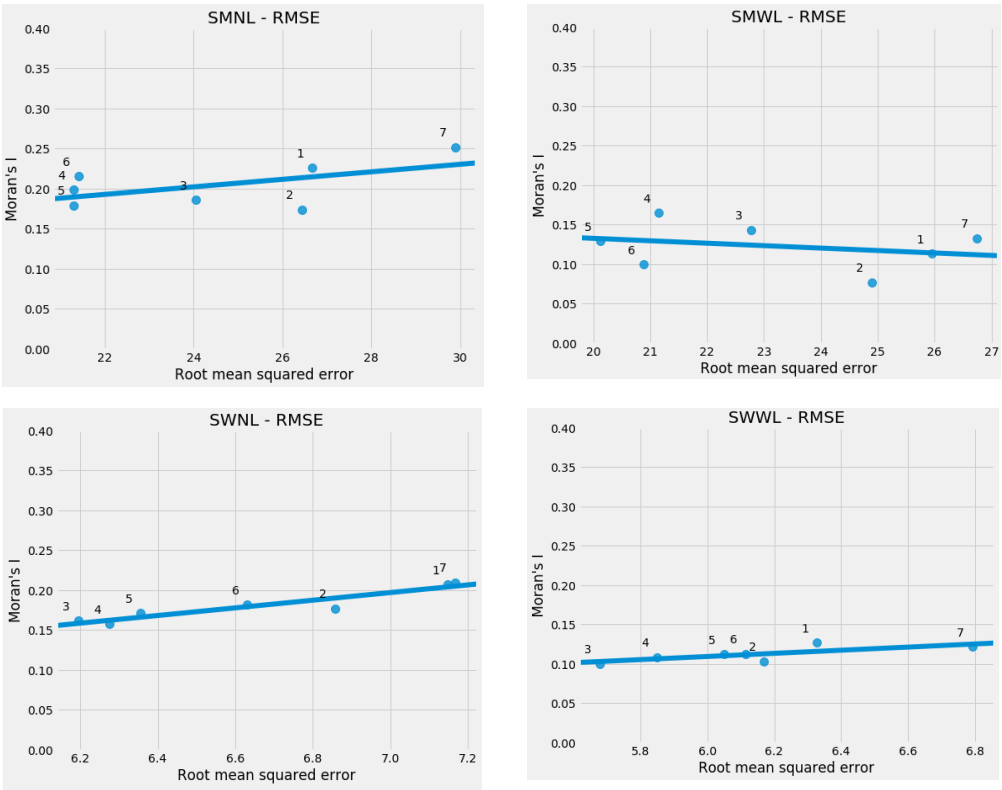


Figure 32. Prediction error of all vanilla SVR experiments is presented and compared.

5.1.3. MIXED EFFECTS RANDOM FOREST

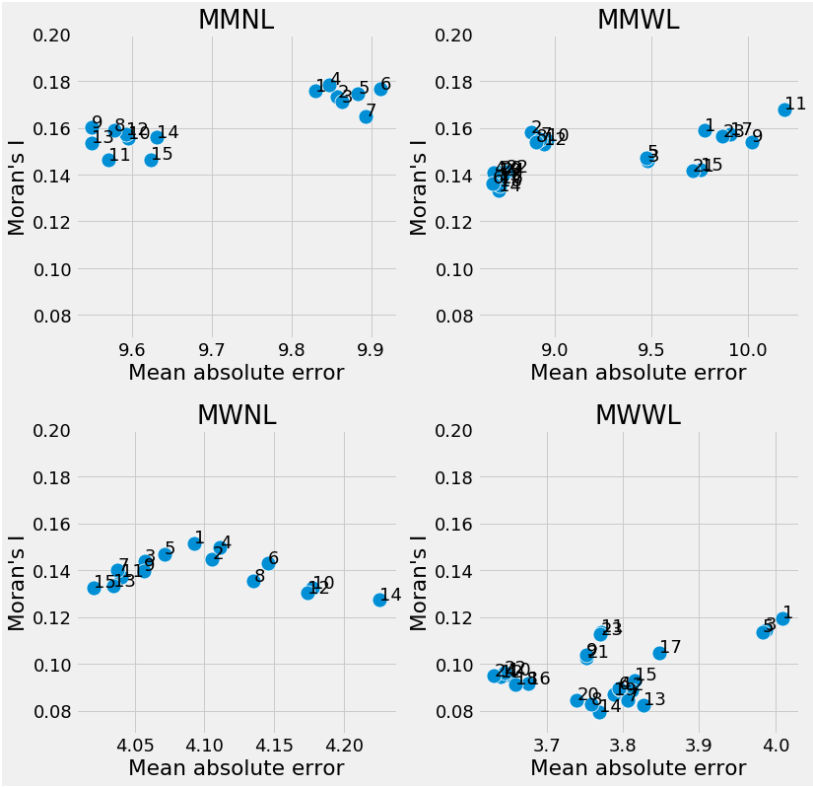


Figure 33. The degree of prediction errors evaluated using MAE is compared to all MERF models.

The aim of the cross-validation work is to select the best model configuration. MERF model evaluation was performed using MAE, MAD and R2 metrics. These metrics were compared to the level of SAC residuals. The results in the cross-validation split were averaged and plotted to the chart. The detail model performance of the best model in each split is also presented.

To begin with, MAE was performed to evaluate errors in the prediction performance along with the degree of SAC residuals. As it can be seen in Figure 33, the prediction errors of the model trained without lagged spatial features and with lagged features measured using MAE are close to each other. However, we can see from Figure 33; there are two cluster points that have different prediction errors of each domain (lagged and non-lagged). For model trained with lagged spatial features, experiment number 1, 2, 3, 5, 9, 11, 15, 17, 23 have slightly larger errors. While the other experiments have lower errors and lower SAC residuals. As for the models were trained using non-lagged spatial features, experiment number 1, 2, 6, 8, 10, 12 and 14 have slightly more errors. The interval prediction errors of the model between non-lagged and lagged are relatively small by 0.05 – 0.9.

The model performance in term SAC residuals, the model trained using monthly dataset have SAC residuals ranging from 0.13 – 0.18. As for model trained using weekly set, the SAC residuals ranging from 0.08 – 0.16. We observed experiments number 14, 13 and 20 for lagged models across time scale dataset have lower the SAC residuals. As for non-lagged models, experiment number 11, 13 and 15 have lower SAC residuals and lower errors.

We can infer that experiment configuration number 15 for non-lagged, and 14 for lagged have a relatively consistent result. These configurations have the lowest SAC and low prediction errors.

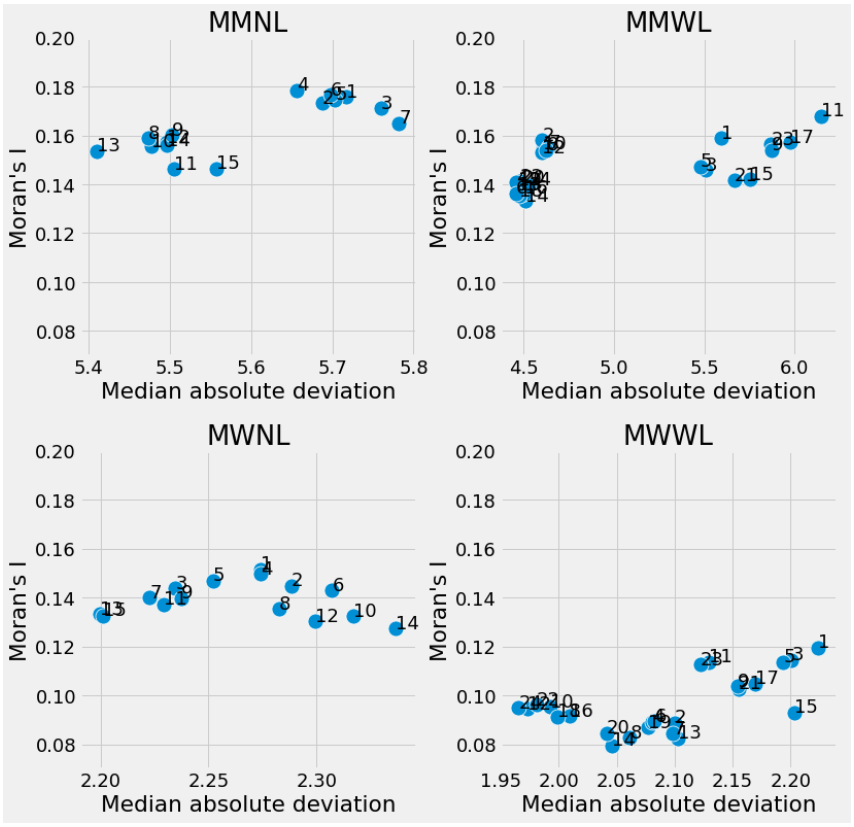


Figure 34. Evaluation of model predictive errors using MAD to all models is compared.

The model performance is also evaluated using MAD. The results are coherent with the MAE metric. As it can be seen in Figure 34, model number 15 for non-lagged and 14 for lagged spatial features are still consistent with low error and SAC residuals.

As for the model prediction accuracy, it can be seen in Figure 35, showing the gap are very close between non-lagged and lagged. Nevertheless, the level of SAC residuals between them is different. Random features significantly affect the magnitude of SAC residual.

The model trained with lagged features has higher accuracy than other models. However, the gap is very close around 0.03 points. Also, the SAC residuals levels are significantly different by 10 – 20%. This result is coherent with the vanilla RF.

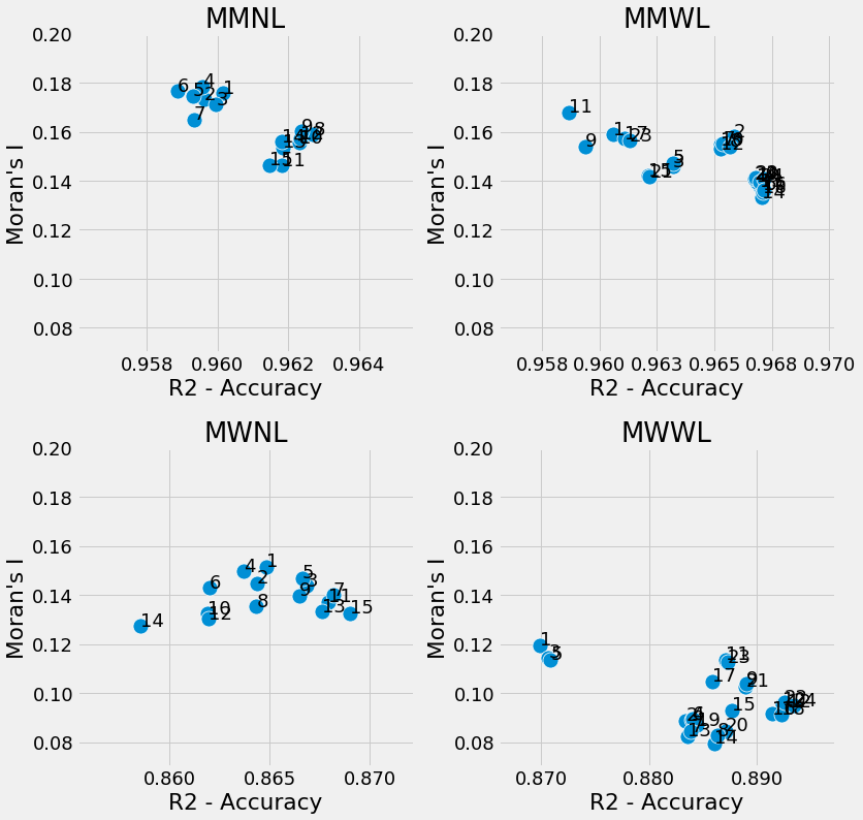


Figure 35. Evaluation of MERF model prediction accuracies measured using r-squared.

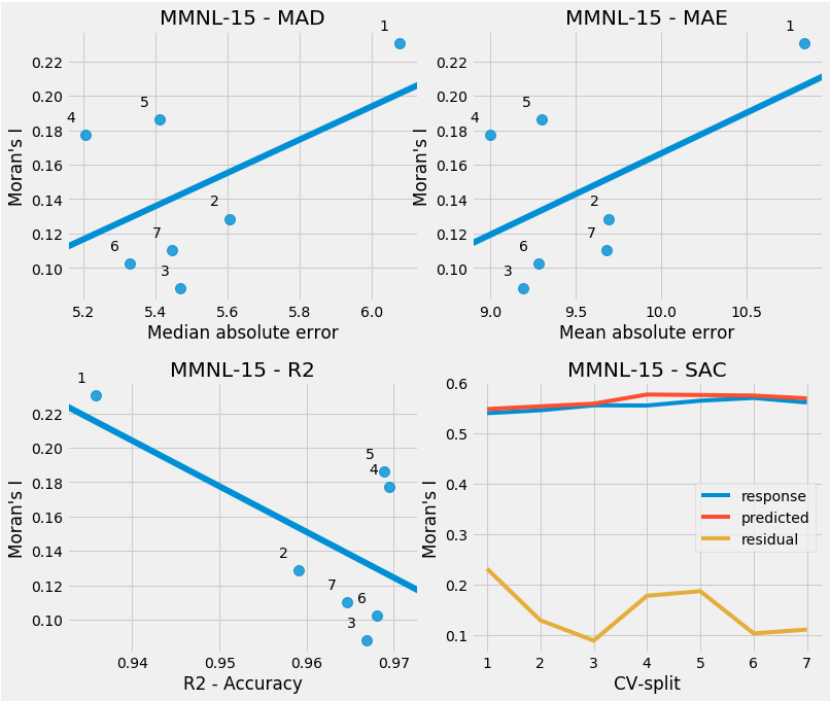


Figure 36. Snapshot the detail performance model MMNL-15 in the cross-validation.

Based on the model metrics evaluation were performed to the model, the model configuration number 15 for non-lagged and 14 for lagged domains were selected to build the final model to predict the unseen data.

The detailed performance of model MMNL-15 is shown in Figure 36. The model prediction errors were assessed with MAD and MAE metrics and have low rate error while keeping the SAC residuals low. As for the model prediction accuracy evaluated using r-squared are getting higher when the SAC residuals are lower. Furthermore, the model can capture the spatial pattern. The detail for the rest of the selected models can be accessed in the appendix.

5.1.4. MIXED EFFECTS SUPPORT VECTOR REGRESSION

MESVR models were trained using all the training dataset except the model trained with the weekly set. The fixed effects best estimator parameters were obtained using randomized search.

The model performances were evaluated using RMSE metrics to measure the magnitude of errors and also r-squared to check the model generalization. To start with, model evaluation using RMSE, as shown in Figure 37, we can observe that the model trained using lagged spatial features, experiment number 14 has a low level on both RMSE and SAC residuals. Although, it also appears that experiment number 15 and 18 have close performance with number 14. As for the model trained using non-lagged spatial features, experiment number 14 and 15 give low prediction error while keeping the SAC residuals low. From Figure 37, also shows the model trained only complaints features give higher error and SAC in the residual regression as in experiment number 1 and two on both lagged and non-lagged across the two different time scale datasets.

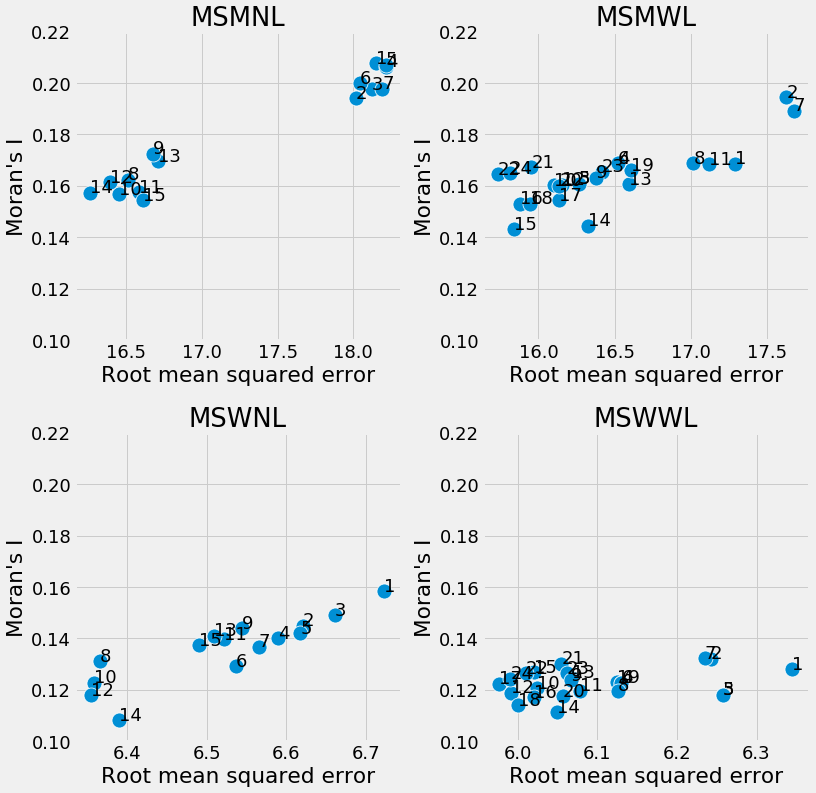


Figure 37. Prediction error of MESVR model is evaluated using RMSE and compared.

The model prediction accuracy assessed using r-squared metrics as shown in Figure 38, the accuracy for the models trained using weekly dataset, MSWNL and MSWWL, give lower prediction accuracy than the model trained using monthly dataset. Similarly, to the results of MERF models, these models have lower SAC residuals

From RMSE and r-squared evaluation results, we conclude that experiment MSMNL-15, MSWNL-14, MSMWL-14 and MSWWL-14 have good prediction accuracy while keeping the SAC residuals low. Hence, these models were selected to build the final model to predict unseen data. These results are quite similar to the best MERF model features configuration. Moreover, the model able to capture spatial pattern. The detail for the rest of the models selected can be accessed in the appendix.

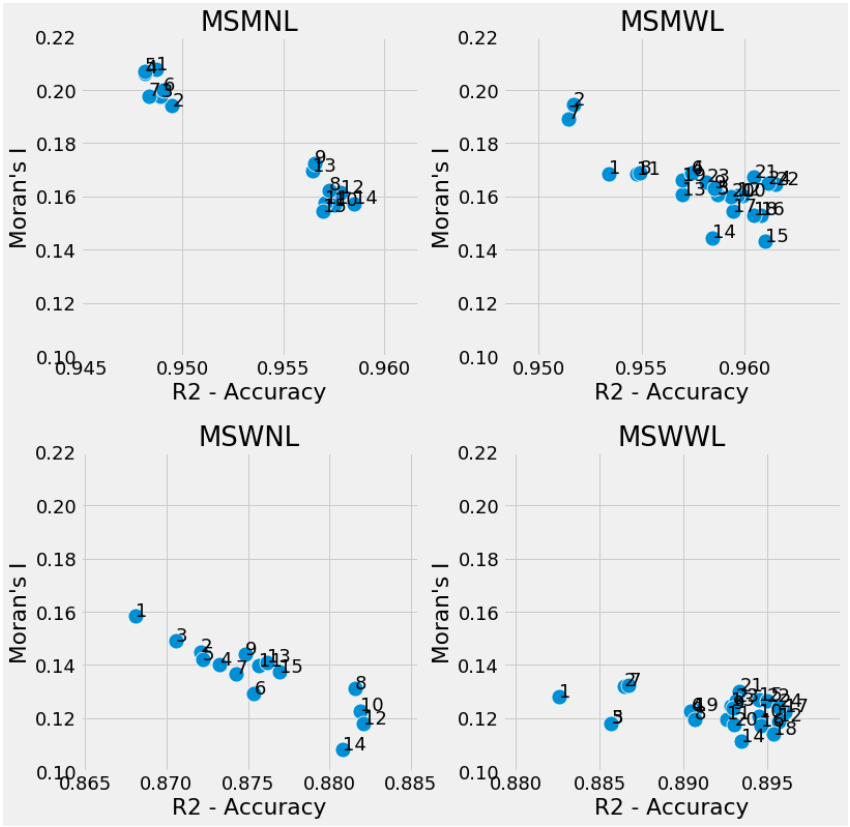


Figure 38. Evaluation of the prediction accuracy of MESVR models using r-squared.

The detailed performance of model MSMNL-15 is shown in Figure 39. The model performance on each split was evaluated using RMSE metrics showing a positive trend. The MSMNL-15 model trained using lagged spatial features as random features and excluded both of them from fixed features. As for the model prediction accuracy evaluated using r-squared are getting higher when the SAC residuals are lower.

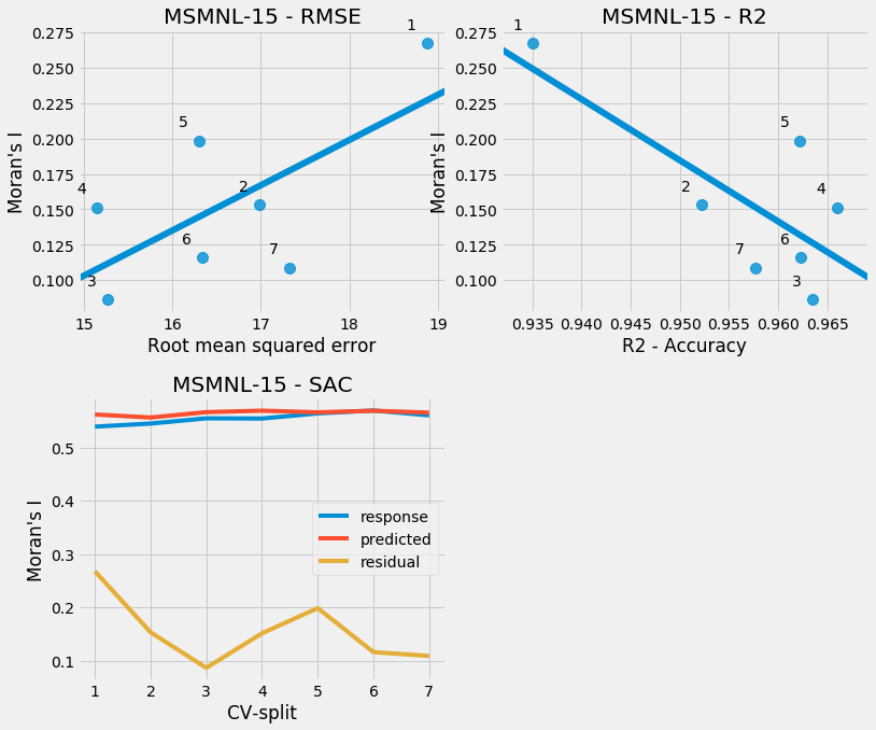


Figure 39. Evaluation of prediction errors on MSMNL-15 model using RMSE

5.2. MODEL PERFORMANCE

In the previous subsections, we performed parameter tuning and cross-validation model evaluation using various metrics to obtain the best model parameter configuration. These configurations were used to retrain the model using the whole training set to evaluate the generalization of the model. The model was evaluated using hold out test data. The aim is to detect and investigate the model generalization to predict and capture spatial pattern from complete a new dataset.

In this subsection, we explain the final model performance comparison to each vanilla RF and SVR and their mixed model counterparts in terms of prediction error, SAC residuals and ability to capture the pattern.

5.2.1. RANDOM FOREST AND MIXED EFFECTS RANDOM FOREST

The MERF model performs better. The predictive performance measured using r-squared is shown in Figure 40. The model that was trained using non-lagged spatial features, MMNL-15 outperforms vanilla RF in terms of accuracy by 10%. Additionally, the models that were trained using lagged spatial features have higher accuracy and lower SAC residuals compared to other models. These results are coherent with cross-validation stage results.

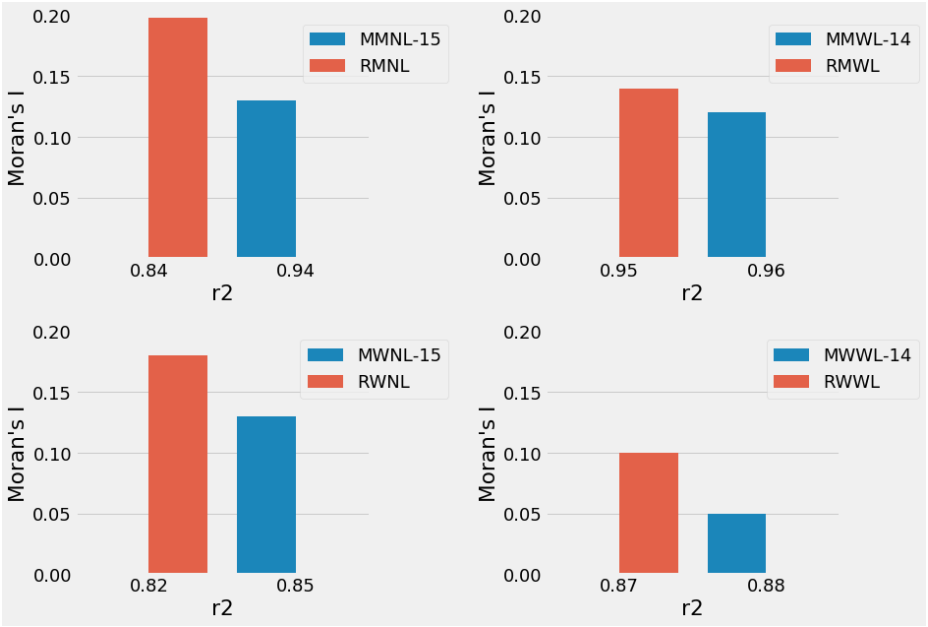


Figure 40. Side by side model prediction accuracy comparison between vanilla RF and MERF models

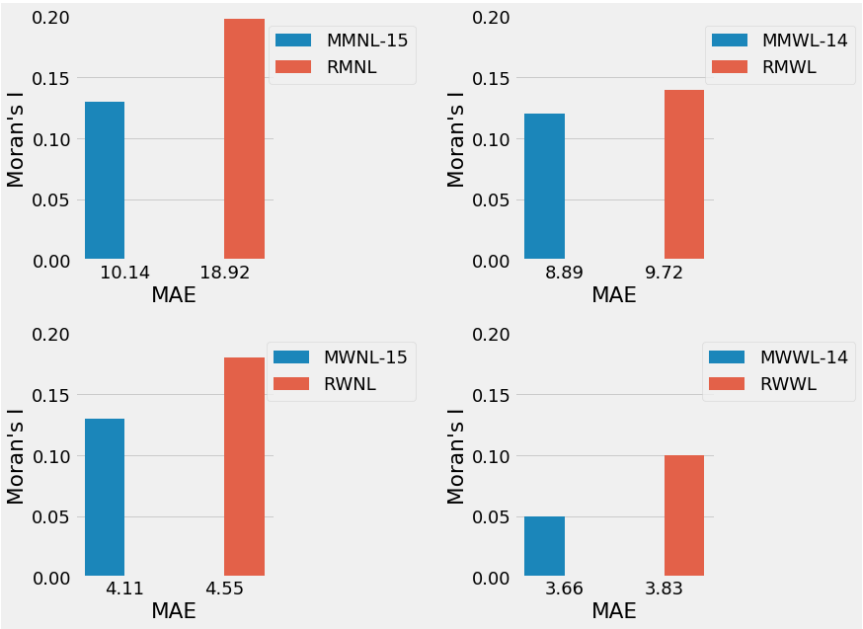


Figure 41. The prediction errors evaluation using MAE shows that MERF models have fewer prediction errors compared with vanilla RF models.

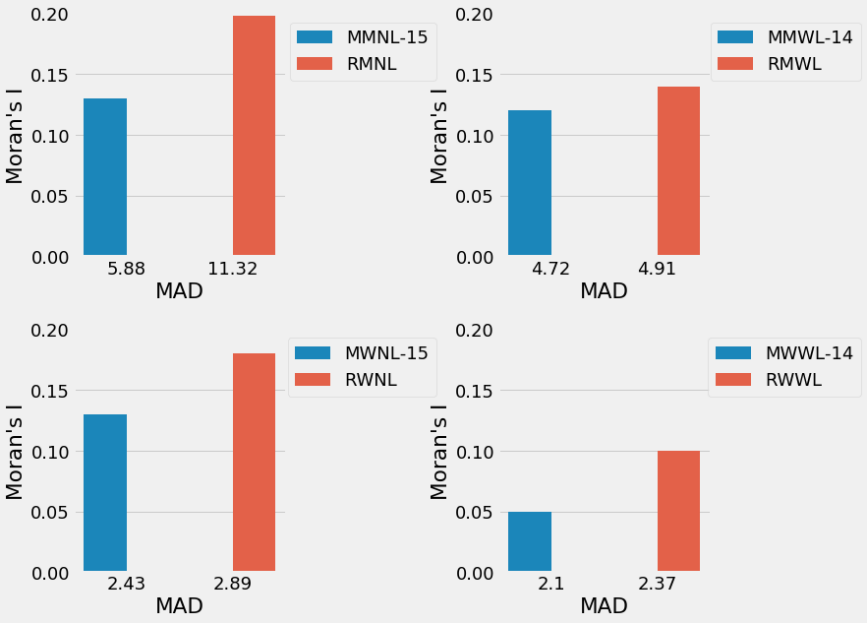


Figure 42. Prediction errors evaluation using MAD metrics on vanilla SVR and MERF

The prediction errors of MERF models are lower than vanilla RF. As it can be seen in Figure 41 and Figure 42, the model prediction deviation was evaluated using MAE and MAD showing the MERF models have significant improvement over vanilla RF. Overall, the MERF models have higher predictive accuracy, lower errors and lower SAC residuals compared vanilla RF. The MWWL-14 has the lowest error and SAC compared with the others.

The MERF models have low SAC residuals. Instead of fitting only using fixed effects features as in RF, MERF model also considers random effects features to capture the correlation between zip code. In other words, random effects features allow for non-independence control as shown in Figure 43.

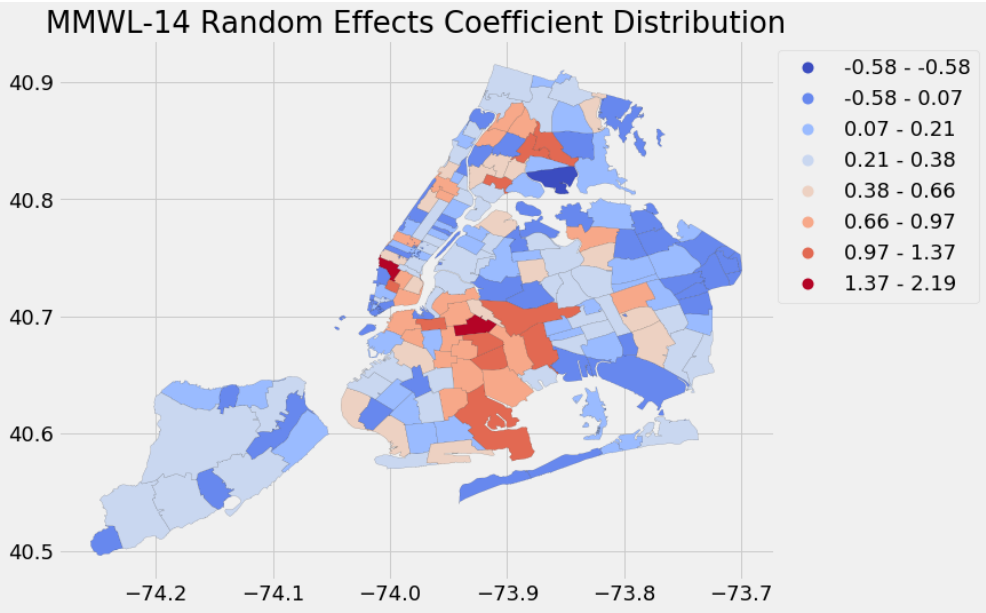


Figure 43. The map shows random effects coefficient distributions for model MMWL-14. These values are varying to all zip code.

By fitting the model only fixed effects features using zip codes as a cluster in RF, the regression assumes that each zip code is independent to each other and share residual variances. When random effects features in the linear mixed effects model are considered, the slope and intercept of the model on each zip code are varied. Thus, putting features such as temporal features, noise vehicle and dirty condition that have a similar trend with the response in the random effect features reduce the SAC residuals.

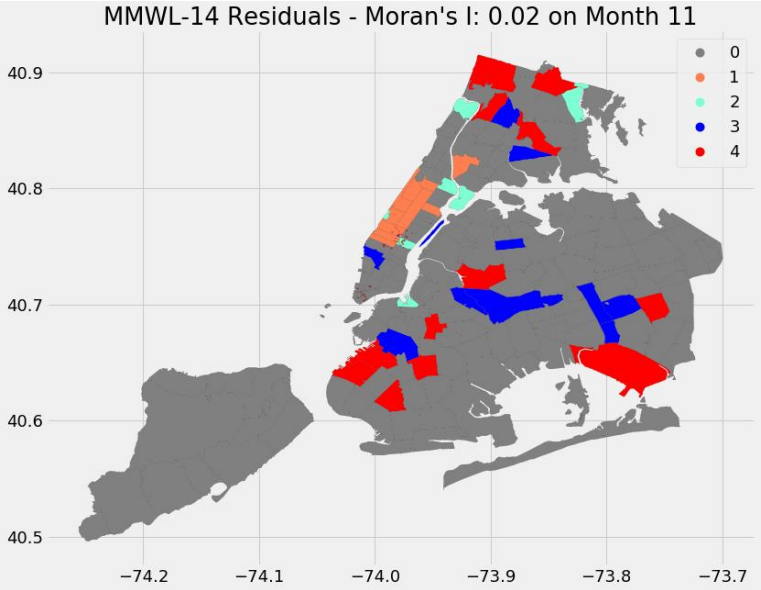


Figure 44. Plotting SAC residuals of MMWL-14 model to the map.

Reducing the SAC residuals is important in the regression analysis as the presence of SAC residuals may cause erroneous in the results interpretation. However, using local Moran's I as shown in Figure 44, there is a small cluster occurrence in the residuals despite the magnitude of global SAC is 0.02 using pseudo p-value <0.05 due to random permutation value used to compute Local Moran's I. Map legend; namely 0 for not significant, 1 for high-high spatial clusters, 2 for low-high spatial outliers, 3 for low – low spatial cluster, 4 for high – low spatial outliers

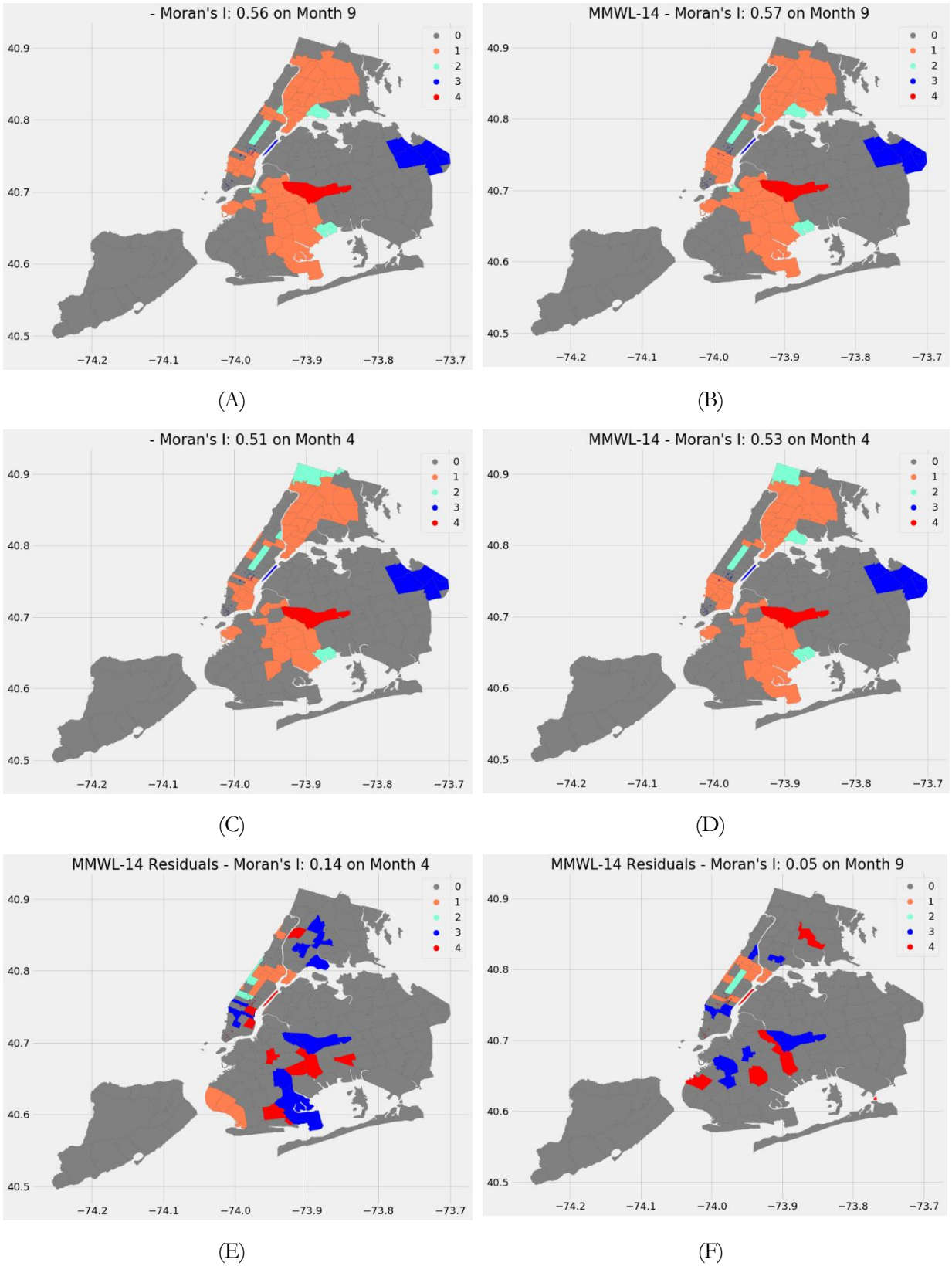
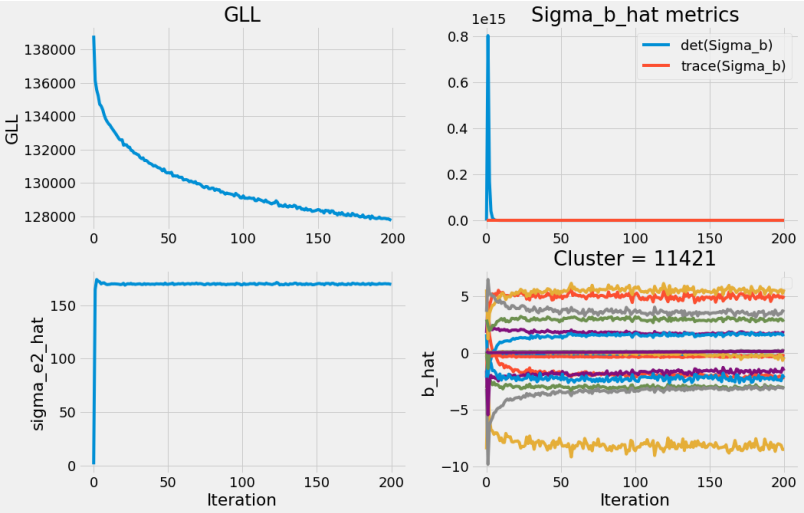


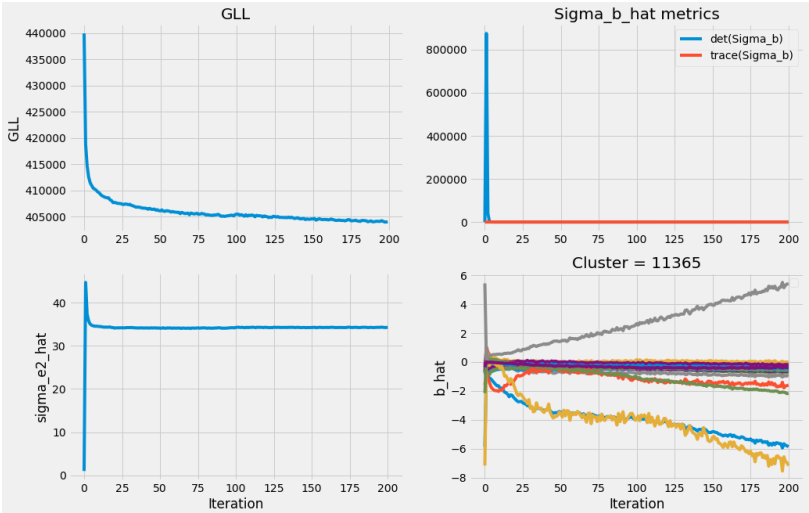
Figure 45. Spatial pattern of crime in New York City on particular month, SAC of each zip code measured using Local Moran's I, while SAC to entire area is measured using Global Moran's I. (A) The spatial pattern of the response variable which has the highest of SAC in 2017 (B) The corresponding predicted SAC pattern, on month 9 (C) The spatial pattern of response variable which has the lowest SAC in 2017, (D) The corresponding predicted SAC pattern on month 4 (E) SAC residuals MMWL-14 on month 4 (F) SAC residuals MMWL-14 on month 9.

MERF models trained using lagged spatial and non-spatial features on monthly data have strong predictive power and ability to capture spatial pattern. As can be seen in Figure 45, the MERF model trained using spatial features was able to capture most of the crime patterns in New York City. The MERF model with non-lagged features has a similar ability to the model trained using spatial features. This means the model can capture the correlation of response induced between zip code through random effects features. The result can be seen in the appendix B. The same applies to the RF model trained using spatial features.

Nevertheless, the MERF model trained using weekly data have lower prediction accuracy; the gap ranges from 6% to 10%. This result can be explained as in the weekly data; the estimated errors variances are less significant than in the monthly data. As it can be seen in Figure 46, sigma_e2_hat or $\hat{\sigma}^2_{(r)}$ MMWL-15 is higher than MWWL-14. These errors variances are originated from fixed noise. Hence, the higher errors variance of fixed effects, the more predictable the response from known clusters in this case zip codes (Hajjem et al., 2014). From this figure, we can also infer that the GLL convergences at 200 iterations. The random effects coefficients for MMWL-15 model is also flat. Conversely, several random effects coefficient for MWWL-14 diverge.



(A)



(B)

Figure 46. Training history on (A) MMWL-15 model and (B) MWWL-14 model

5.2.2. SUPPORT VECTOR REGRESSION AND MIXED EFFECTS SUPPORT VECTOR REGRESSION

The final MESVR models performance trained using monthly outperform vanilla SVR. These models have smaller errors compared with vanilla SVR models by 35% – 43%. These results are measured using RMSE metrics as shown in Figure 47. However, MESVR models using weekly dataset have a relatively good generalization as were trained using subsample dataset from 2012 to 2014. These models have a prediction error that is higher but close to vanilla SVR models, which were trained using the whole training. The error difference is between 0.72 and 0.51.

Similarly, to the analytical results of the random forest models, RMSE & SAC residuals are lower for three out four spatial lagged models compared to their non-spatial lagged counterparts. These models are SMWL SWWL and MSWWL.

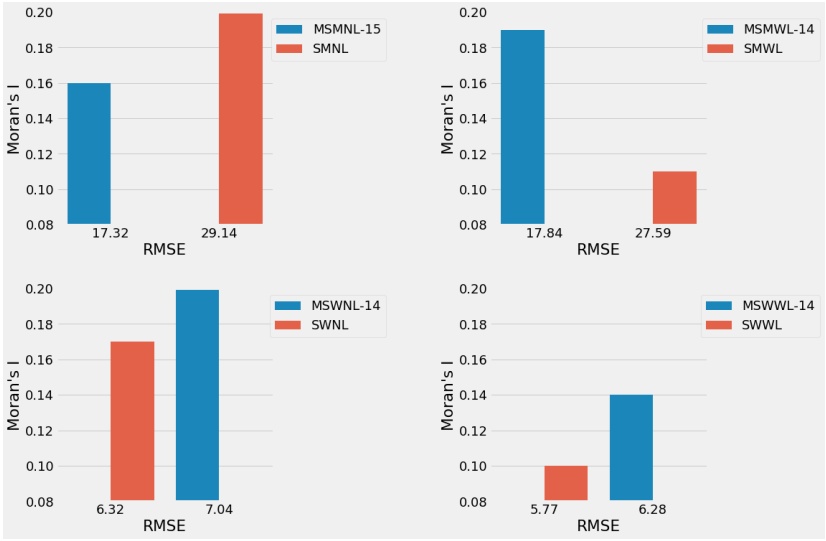


Figure 47. The MESVR and SVR model prediction errors were evaluated using RMSE and compared.

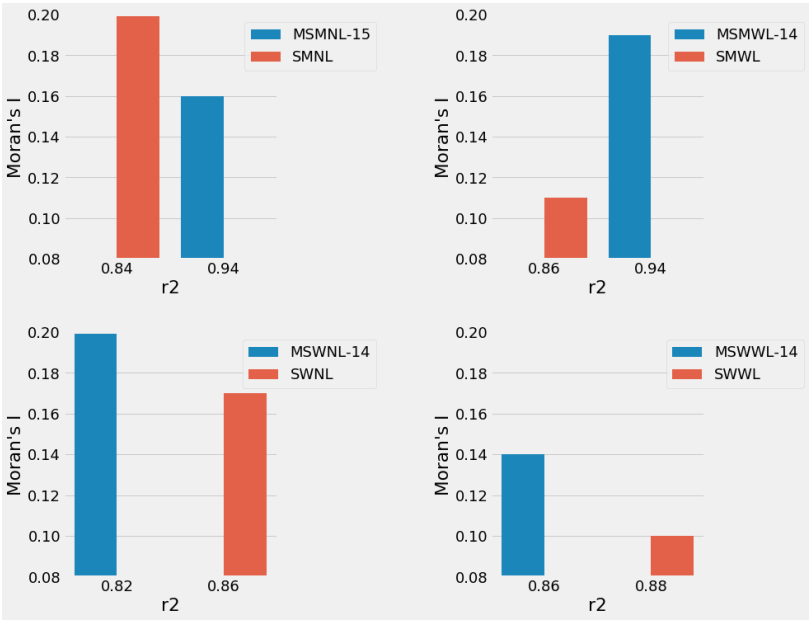
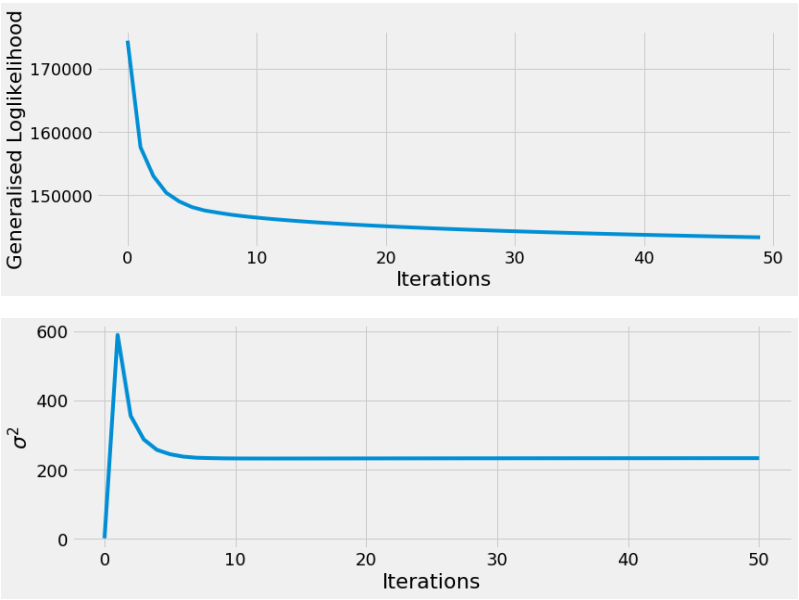


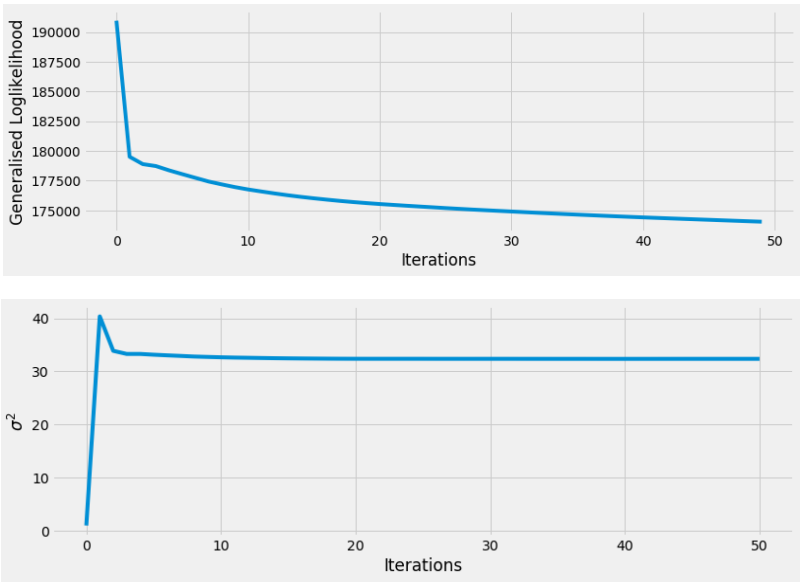
Figure 48. The final model generalization performance of MESVR and SVR are measured using r-squared and compared.

MSMNL-15 and MSMWL-14 outperform vanilla SVR models with 10% in term of accuracy as shown in Figure 48. MSWNL-14 and MSWWL-14 models that were trained using a subsample dataset have good generalization with 82% and 86% respectively. Although SMWL has significantly lower accuracy than MSMWL but the SAC residual is significantly lower.

It is likely the model trained using weekly dataset have similar error structure as MERF model. As it can be seen in Figure 49, GLL flattens at 50 iterations, but the MSWWL-14 model has difficulty in predicting the response from the new dataset due to fixed effects noise. The variance error ($\hat{\sigma}^2_{(r)}$) of MSMWL-14 is higher than MSWWL-14. Hence, the model that was trained with monthly dataset has better generalization.



(A)



(B)

Figure 49. Training statistics of (A) MSMWL-14 and (B) MSWWL-14 are compared. GLL for both models flattens and convergences for 50 iterations.

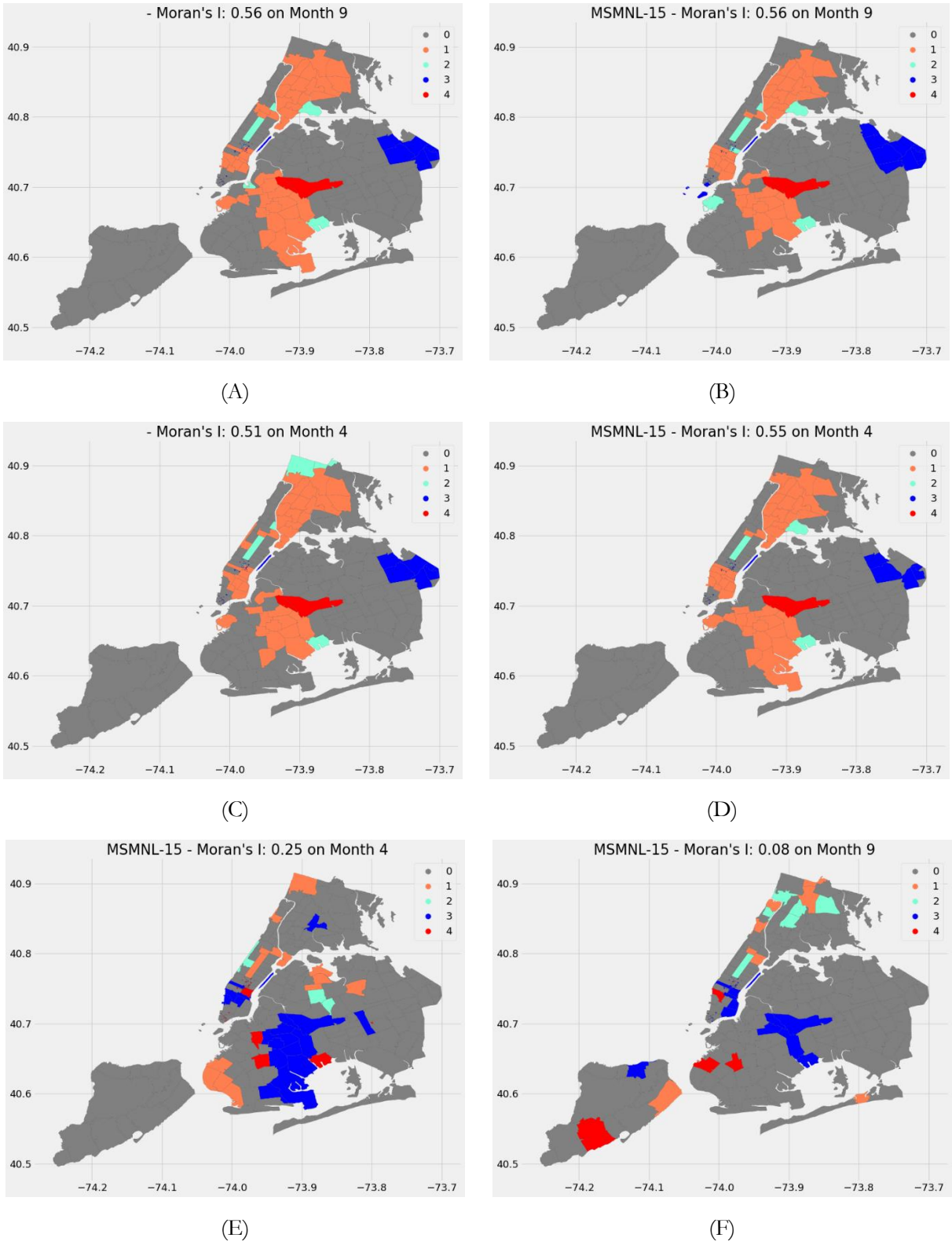


Figure 50. Spatial pattern of crime occurrences in New York City on particular months, (A) The spatial pattern of the response variable which has the highest SAC in 2017 (B) The predicted pattern of the response variable, which has the highest SAC in 2017, (C) The spatial pattern of the response variable, which has the lowest of SAC response in 2017, (D) The predicted pattern of the response variable, which has the lowest of SAC response in 2017 (E) and (F) are SAC residuals of the response variable of the lowest SAC and highest SAC respectively

The MESVR model's ability to capture spatial pattern as shown in Figure 50. The model can capture the spatial pattern when the SAC residuals are almost zero. However, when the SAC residuals are getting higher, the model's predicted pattern deviates slightly from the real spatial patterns.

5.2.3. MIXED EFFECTS RANDOM FOREST AND MIXED EFFECTS SUPPORT VECTOR REGRESSION

MERF model performs better than MESVR in terms of predictive accuracy and SAC residuals. As it can be seen in Table 5.1, the prediction errors showing that each MERF experiment has fewer errors compared with its counterparts. Moreover, MERF models also have lower SAC residuals ranging from 3% - 10%. Therefore, MERF model has better performance and ability to capture the spatial patterns.

Table 5.1. Model generalization of MERF and MESVR are evaluated using various metrics and compared

ME	Experiment	MAD	MAE	R^2	RMSE	MI Response	MI Residuals	MI Predicted
SVR	MNL-15	6.47	10.797	0.944	17.322	0.538	0.164	0.557
	MWL-14	5.772	10.551	0.941	17.835	0.538	0.191	0.545
	WNL-14	2.418	4.277	0.822	7.042	0.51	0.229	0.567
	WWL-14	2.272	3.967	0.858	6.278	0.51	0.138	0.536
RF	MNL-15	5.885	10.144	0.944	17.382	0.538	0.134	0.549
	MWL-14	4.724	8.888	0.959	14.947	0.538	0.122	0.552
	WNL-15	2.432	4.11	0.846	6.535	0.51	0.132	0.54
	WWL-14	2.097	3.662	0.879	5.795	0.51	0.052	0.52

5.2.4. COMPUTATIONAL TIME AND COMPLEXITY

Naturally, vanilla RF was the fastest and followed by SVR. Training time using RF in the hyperparameter tuning ranged from 5 – 40 minutes. However, the computational time becomes expensive when the density of the data and the features become bigger and larger, such as to train a model with cross-validation using a weekly dataset with more than 90.000 rows and hundred features.

The computation time required to train the model on a weekly set is nearly five times longer than monthly set as shown in Figure 51 below which also shows that using more features does not affect the computational time. Moreover, it turns out that the time required to train the model with additional features, which is the spatial and temporal lagged of the response variable is faster than train only with complaint features.

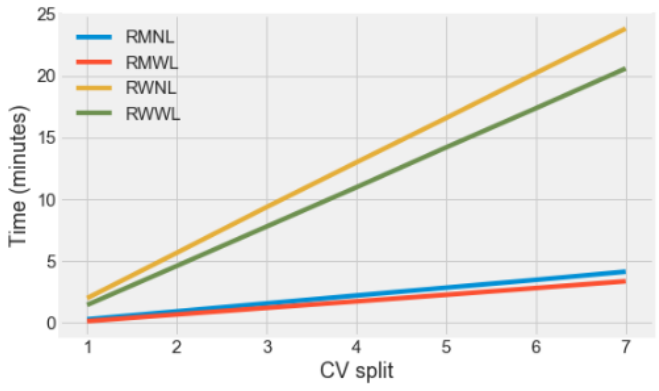


Figure 51. Computation time required to train the RF model in the cross-validation.

SVR training is very expensive since SVR solves an optimization problem of quadratic order. The training time on monthly scale dataset ranges from ten minutes to three hours in the hyperparameter tuning phase. Nevertheless, the time required to train the model increased linearly when the data are getting larger.

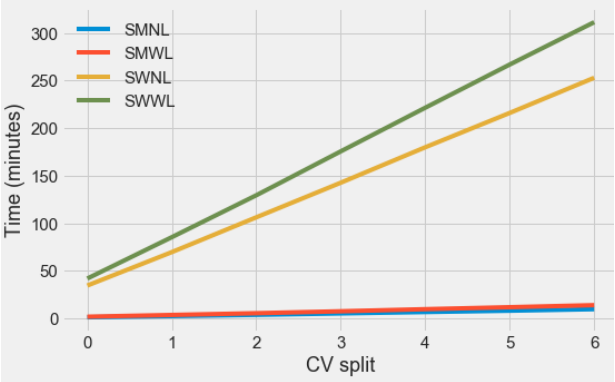


Figure 52. Computation time required to train the SVR model in the cross-validation.

It can be seen in Figure 52, SWWL model, using all features, needs approximately around one hour in each split to learn the data.

EM algorithm in the MERF is expensive. Training using the weekly dataset in the hyperparameter tuning need at least two to three hours to build one model using 200 estimators for 100 to 200 iterations. We have 15 and 24 combinations of non-spatial features and spatial features respectively. These combinations render 30 – 62 hours to complete cross-validation using seven folds split.

MESVR is the most expensive to train the model using the weekly dataset. This is why we use subsampled data to train the model. Using the whole training set, it needs at least three to four hours per fold, translating 21 hours using seven folds to complete one experiment. By reducing the size of the data and re-optimizing the kernel with loosening C and gamma, it needs around 30 – 40 minutes for 20 iterations per fold.

These algorithms are still depending on CPU cores to model. These results obtained using 16 cores CPU.

6. CONCLUSIONS AND RECOMMENDATIONS

6.1. CONCLUSIONS

In this MSc thesis, we investigated whether mixed effects machine learning regression model capture spatial pattern better than their vanilla counterparts. We reviewed RF and SVR and their mixed model counterparts to predict crime in New York City. We used seven years of data to train the model and one-year data for testing and also we worked with data at two temporal scales, monthly and weekly. Moreover, we used spatial features as both random and fixed effects in the model development. We used the best parameter configuration to each algorithm to train the model. The effect of using a various set of random and fixed effects features on the model prediction performance and its ability to capture spatial patterns also was investigated.

Our results show that the MERF and MESVR models trained using the complete training set outperform their vanilla counterparts. MERF models perform better than MESVR model in all performance metrics. We found that the optimal models of MERF and MESVR use a similar combination of random and fixed effects features. The model trained using lagged spatial features as in the experiment number 14 using the combination as follows: complaint and lagged spatial lag features as fixed effects, temporal features as random effects and lagged LISA's quadrant features as both fixed and random effects. As for the model trained using non-lagged spatial features as in experiment number 15, by placing temporal, and other two complaint features; namely noise_vehicle and dirty_condition as both fixed and random effects.

The model generalization to test set data also observed. Vanilla RF and SVR and their mixed effects counterparts that were trained using whole training set can generalize pretty well onto a hold out data, known as test set or unseen data. As for the MESVR models that were trained using subsampled weekly dataset relatively able to generalize well the test set despite having lower prediction accuracy than in the cross-validation, ranges by 0.02 – 0.05 measured using r-squared. Moreover, generally, the models that were trained using lagged spatial features have strong predictive power and lower SAC residuals than the others. The models that have low SAC residuals can capture spatial pattern pretty well.

The answer to the research question listed in subsection 1.2.1 is as follows:

Review the vanilla machine learning regression model; RF, SVR, and their mixed effects counterparts; MERF and MESVR and relate them to spatial data.

1. How do vanilla RF, SVR and their mixed effects counterparts approach work?

   In subsection 2.2 we discussed on theoretically behind the vanilla RF and SVR algorithms while in 2.3, we described a linear mixed model and mixed effects machine learning regression model. Mixed effects model works well with the data that has a cluster structured given any longitudinal or hierarchical datasets along with the response and several cluster inside. Recall equation 2.8 and 2.13, vanilla RF and SVR estimate the model only from fixed effects features. This means the fixed effects coefficients are estimated using variation of n-sample (subsample) dataset within each cluster and using the only variance between cluster. As for mixed effects models, recall equation 2.15 and 2.16, the model estimated using both fixed effects and random effects. Unlike their vanilla counterparts, the variation between cluster also contains random effects information. The architecture of each algorithm was elaborated in detail.

2. How can machine learning regression model approaches be used to model spatial data?

   In chapter 2, we adopted various approaches, Santibanez, Kloft, et al. (2015), we used zip code as spatial features to aggregate the occurrence of crime and complaints. Hengl et al. (2018) use spatial distance, in this thesis, to approximate the distance between each cluster we slightly modified it by using temporally and spatially lagged of the response variable and also the lagged LISA's value of response variable of each cluster. Rocha et al. (2018) to minimize overfitting of the model use ten folds cross-validation. However, as in subsection 4.1.1, in the cross-validation, we used group k-fold to split dataset using year as time slicer to retain spatial structure in the dataset. The number of folds to evaluate the model performance are varies and depends on the spatial structure in the data. There are two kinds of fold we used to validate the model, seven folds and three folds for the subsampled dataset. To reduce SAC residuals and improve predictive performance, in the cross-validation using mixed effects machine learning models we developed 15 models from different random and fixed features combinations of non-lagged features and 24 models from different random and fixed features combinations of lagged spatial features across two different temporal scales. As for their vanilla machine learning counterparts, we trained the model only with fixed effects using non-lagged and lagged variations.

Design, develop and evaluate vanilla and mixed effects machine learning regression models using spatiotemporal (crowdsourced) data from the crime domain

3. Can MESVR regression approach be developed and if so, how to apply regression?

   In subsection 2.3.2, we show how to use the framework of MERF and replace the non-linear function f(.) of RF with SVR algorithms. Out of bag prediction was replaced with a subsampled the training set in the cross-validation split. EM algorithm and GLL were not replaced. Slightly different from MERF, MESVR can use the already optimal model estimator which is SVR to train the model. The parameter of MERF was slightly modified; the number of estimators of RF was replaced with the model estimator.

4. How should the spatial features be applied to machine learning?

   In chapter 3, three kinds of spatial features are described that were used to train the model. To begin with, we use zip codes. Zip code as spatial features was used to train the model because it has geometry inside. Vanilla machine learning regression model used zip codes id as a feature to address the cluster. Therefore, we one hot encoded the zip code and generated another 248 features. As for their mixed effect counterparts, this process becomes easier, as we just put zip code as a cluster. Consequently, hyperparameter tuning of mixed effects machine learning model is effortless. We also use a spatial lag of response variable obtained using queen contiguity and weighted to each cluster neighbourhood. This spatial lag shifted to a minus one year and assigned to random or fixed effects or both of them. The last spatial feature is LISA's local moran. The lagged LISA's values are discrete. Thus, we one hot encoded the significance value of each cluster. These features to inform the model the correlation between cluster. Similar to spatial lag, it was obtained using queen contiguity and shifted to the next year.

5. How mixed effects machine learning approach deal with clustering in the data caused by geographical relationship?

   In chapter 4, in the mixed effects model, each cluster contains random effects information. Hence, there is a possibility of a correlation between cluster through random effects. Thus, we assigned zip code as a cluster and lagged spatial features; namely temporally lagged spatial lag to fixed effects and LISA's value to both random and fixed effects. The models that were trained using lagged spatial features, lagged spatial lag as fixed effects and LISA's value as both random and fixed effects give strong predictive performance and lower SAC residuals compared the others.

6. Which approaches perform better regarding predictive accuracy?

   In chapter 5, model results and performance are reported and discussed. MERF models have the best performance results using various metrics compared with other algorithms. This was achieved by experiment MMWL-14, MMNL-15, MWNL-15, MWWL-14.

7. What is the difference between mixed effects and general machine learning regarding the degree of SAC in the residuals?
   In chapter 5, MERF models have higher predictive accuracy and lower the SAC residuals compared with RF. Inversely, only one MESVR model has lower SAC residuals compared with SVR models. However, the prediction accuracy of MESVR model trained with monthly dataset outperforms vanilla SVR with a considerable margin.

6.2. RECOMMENDATIONS

For future research, I recommend:

i. To further investigate mixed effects models with clusters defined with k-means clustering instead of predefined cluster boundary as polygon or multipolygon.
ii. To further investigate the performance of MERF and MESVR using different domain and spatial resolution.
iii. To further investigate the performance of MERF and MESVR on spatial and also temporal autocorrelation using spatiotemporal dataset.
iv. To further investigate the use of lagged spatial features in the regression using different datasets that the response variables have random temporal patterns.
v. To develop and tune SVR and MESVR using GPGPU or splitting into several nodes using messages passing interface (MPI) to accelerate the training process. Since the model performance of MESVR and SVR using RBF kernel look promising.

LIST OF REFERENCES

Anselin, L. (1995). Local Indicators of Spatial Association—LISA. *Geographical Analysis*, *27*(2), 93–115. https://doi.org/10.1111/j.1538-4632.1995.tb00338.x

Ballabio, C., & Comolli, R. (2010). Mapping Heavy Metal Content in Soils with Multi-Kernel SVR and LiDAR Derived Data. *Digital Soil Mapping: Bridging Research, Environmental Application, and Operation*, *2*, 205–216. https://doi.org/10.1007/978-90-481-8863-5

Barron, J. (2018). New York City's Population Hits a Record 8.6 Million - The New York Times. Retrieved December 18, 2018, from https://www.nytimes.com/2018/03/22/nyregion/new-york-city-population.html

Baydaroğlu, Ö., Koçak, K., & Duran, K. (2018). River flow prediction using hybrid models of support vector regression with the wavelet transform, singular spectrum analysis and chaotic approach. *Meteorology and Atmospheric Physics*, *130*(3), 349–359. https://doi.org/10.1007/s00703-017-0518-9

Bergstra, James, & Bengio, Y. (2012). *Random Search for Hyper-Parameter Optimization*. *Journal of Machine Learning Research* (Vol. 13). https://doi.org/10.1162/153244303322533223

Bertazzon, S., Johnson, M., Eccles, K., & Kaplan, G. G. (2015). Accounting for spatial effects in land use regression for urban air pollution modeling. *Spatial and Spatio-Temporal Epidemiology*, *14–15*, 9–21. https://doi.org/10.1016/j.sste.2015.06.002

Bhattacharyya, I. (2018). Support Vector Regression Or SVR. Retrieved December 13, 2018, from https://medium.com/coinmonks/support-vector-regression-or-svr-8eb3acf6d0ff

Blood, E. A., Cabral, H., Heeren, T., & Cheng, D. M. (2010). Performance of mixed effects models in the analysis of mediated longitudinal data. *BMC Medical Research Methodology*, *10*(1), 16. https://doi.org/10.1186/1471-2288-10-16

Borman, S. (2004). *The Expectation Maximization Algorithm; A short tutorial*. Retrieved from https://www.cs.utah.edu/~piyush/teaching/EM_algorithm.pdf

Breiman, L. (2001). *Random Forests* (Vol. 45). Retrieved from https://link.springer.com/content/pdf/10.1023%2FA%3A1010933404324.pdf

Brownlee, J. (2013). How to Prepare Data For Machine Learning. Retrieved January 20, 2019, from https://machinelearningmastery.com/how-to-prepare-data-for-machine-learning/

Bzdok, D., Altman, N., & Krzywinski, M. (2018). Points of Significance: Statistics versus machine learning. *Nature Methods*, *15*(4), 233–234. https://doi.org/10.1038/nmeth.4642

Cawley, G. C., & Talbot, N. L. C. (2010). *On Over-fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation*. *Journal of Machine Learning Research* (Vol. 11). Retrieved from http://www.jmlr.org/papers/volume11/cawley10a/cawley10a.pdf

Cawley, G. C., Talbot, N. L. C., Guyon, I., & Saffari, A. (2007). *Preventing Over-Fitting during Model Selection via Bayesian Regularisation of the Hyper-Parameters*. *Journal of Machine Learning Research* (Vol. 8). Retrieved from http://www.modelselect.inf.ethz.ch/index.php.

Chen, C., Yan, C., Zhao, N., Guo, B., & Liu, G. (2017). A robust algorithm of support vector regression with a trimmed Huber loss function in the primal. *Soft Computing*, *21*(18), 5235–5243. https://doi.org/10.1007/s00500-016-2229-4

Chen, G., Knibbs, L. D., Zhang, W., Li, S., Cao, W., Guo, J., … Guo, Y. (2018). Estimating spatiotemporal distribution of PM1concentrations in China with satellite remote sensing, meteorology, and land use information. *Environmental Pollution*, *233*, 1086–1094. https://doi.org/10.1016/j.envpol.2017.10.011

Chen, G., Wang, Y., Li, S., Cao, W., Ren, H., Knibbs, L. D., … Guo, Y. (2018). Spatiotemporal patterns of PM10 concentrations over China during 2005–2016: A satellite-based estimation using the random forests approach. *Environmental Pollution*, *242*, 605–613. https://doi.org/10.1016/j.envpol.2018.07.012

Chen, Y. (2016). Spatial Autocorrelation Approaches to Testing Residuals from Least Squares Regression. *PloS One*, *11*(1), e0146865. https://doi.org/10.1371/journal.pone.0146865

Cherkassky, V., & Ma, Y. (2004). Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Networks*, *17*(1), 113–126. https://doi.org/10.1016/S0893-6080(03)00169-2

Cho, D. (2010). Mixed-effects LS-SVR for longitudinal data. *Journal of the Korean Data &*, *21*(2), 363–369. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1027.5219&rep=rep1&type=pdf

Chou, Y.-H. (1995). Spatial pattern and spatial autocorrelation. In A. U. Frank & W. Kuhn (Eds.), *Spatial Information Theory A Theoretical Basis for GIS* (pp. 365–376). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-60392-1_24

Czernecki, B., Nowosad, J., & Jabłońska, K. (2018). Machine learning modeling of plant phenology based on coupling satellite and gridded meteorological dataset. *International Journal of Biometeorology*, 1–13. https://doi.org/10.1007/s00484-018-1534-2

Deng, H. (2018). An Introduction to Random Forest – Towards Data Science. Retrieved December 13, 2018, from https://towardsdatascience.com/random-forest-3a55c3aca46d

Department of Information Technology & Telecommunications (DoITT). (2018). Zip Code Boundaries | NYC Open Data. Retrieved February 1, 2019, from https://data.cityofnewyork.us/Business/Zip-Code-Boundaries/i8iw-xf4u

Dormann, C. F., M. McPherson, J., B. Araújo, M., Bivand, R., Bolliger, J., Carl, G., … Wilson, R. (2007). Methods to account for spatial autocorrelation in the analysis of species distributional data: A review. *Ecography*, *30*(5), 609–628. https://doi.org/10.1111/j.2007.0906-7590.05171.x

Esri. (2013). Regression analysis basics. Retrieved August 14, 2018, from http://resources.esri.com/help/9.3/arcgisengine/java/GP_ToolRef/Spatial_Statistics_toolbox/regression_analysis_basics.htm

Feng, Y., Chen, L., & Chen, X. (2018). The impact of spatial scale on local Moran's I clustering of annual fishing effort for Dosidicus gigas offshore Peru. *Journal of Oceanology and Limnology*. https://doi.org/10.1007/s00343-019-7316-9

Fox, J. (2018). Why London Has More Crime Than New York. Retrieved December 18, 2018, from https://www.bloomberg.com/opinion/articles/2018-12-18/michael-flynn-sentencing-washington-owes-him-an-apology

Galbraith, S., Daniel, J. A., & Vissel, B. (2010). A Study of Clustered Data and Approaches to Its Analysis. https://doi.org/10.1523/JNEUROSCI.0362-10.2010

Guo, Y., Li, X., Bai, G., & Ma, J. (2012). Time series prediction method based on LS-SVR with modified Gaussian RBF. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 7664 LNCS, pp. 9–17). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-34481-7_2

Hagenauer, J., & Helbich, M. (2013). Hierarchical self-organizing maps for clustering spatiotemporal data. *International Journal of Geographical Information Science*, *27*(10), 2026–2042. https://doi.org/10.1080/13658816.2013.788249

Hajjem, A., Bellavance, F., & Larocque, D. (2014). Mixed-effects random forest for clustered data. *Journal of Statistical Computation and Simulation*, *84*(6), 1313–1328. https://doi.org/10.1080/00949655.2012.741599

Hastie, T. T. (2017). *The Elements of Statistical Learning Second Edition. Math. Intell.* (Vol. 27). https://doi.org/111

Hengl, T., Nussbaum, M., Wright, M. N., Heuvelink, G. B. M., & Gräler, B. (2018). Random forest as a generic framework for predictive modeling of spatial and spatio-temporal variables. *PeerJ*, *6*, e5518. https://doi.org/10.7717/peerj.5518

Hoef, J. M. V., London, J. M., & Boveng, P. L. (2010). Fast computing of some generalized linear mixed pseudo-models with temporal autocorrelation. *Computational Statistics*, *25*(1), 39–55. https://doi.org/10.1007/s00180-009-0160-1

Hua, W., Junfeng, Z., Fubao, Z., & Weiwei, Z. (2016). Analysis of spatial pattern of aerosol optical depth and affecting factors using spatial autocorrelation and spatial autoregressive model. *Environmental Earth Sciences*, *75*(9), 822. https://doi.org/10.1007/s12665-016-5656-8

Hulin, W., & Zhang, J.-T. (2006). Parametric Mixed-Effects Models. In *Nonparametric Regression Methods for Longitudinal Data Analysis: Mixed-Effects Modeling Approaches* (pp. 17–39). John Wiley & Sons, Ltd. https://doi.org/10.1002/0470009675.ch2

Jin, X., Sun, Z., Wang, H., Wang, F., & Yan, Q. (2013). Application of improved support vector machine regression analysis for medium- and long-term vibration trend prediction. *Journal of Vibroengineering*, *15*(2), 942–950. Retrieved from https://www.jvejournals.com/article/10083/pdf

Kleynhans, T., Montanaro, M., Gerace, A., & Kanan, C. (2017). Predicting top-of-atmosphere thermal radiance using MERRA-2 atmospheric data with deep learning. *Remote Sensing*, *9*(11), 1133. https://doi.org/10.3390/rs9111133

Kong, Q., Allen, R. M., Schreier, L., & Kwon, Y.-W. (2016). MyShake: A smartphone seismic network for earthquake early warning and beyond. *Science Advances*, *2*(2), e1501055–e1501055.

https://doi.org/10.1126/sciadv.1501055
Kwan, M. P., & Neutens, T. (2014). Space-time research in GIScience. *International Journal of Geographical Information Science*, *28*(5), 851–854. https://doi.org/10.1080/13658816.2014.889300
Lary, D. J., Zewdie, G. K., Liu, X., Wu, D., Levetin, E., Allee, R. J., … Aurin, D. (2018). Machine Learning Applications for Earth Observation. In *Earth Observation Open Science and Innovation* (pp. 165–218). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-65633-5_8
Legendre, P., Dale, M. R. T., Fortin, M. J., Gurevitch, J., Hohn, M., & Myers, D. (2002). The consequences of spatial structure for the design and analysis of ecological field surveys. *Ecography*, *25*(5), 601–615. https://doi.org/10.1034/j.1600-0587.2002.250508.x
Leung, Y., Mei, C.-L., & Zhang, W.-X. (2000). Testing for Spatial Autocorrelation among the Residuals of the Geographically Weighted Regression. *Environment and Planning A*, *32*(5), 871–890. https://doi.org/10.1068/a32117
Lichstein, J. W., Simons, T. R., Shriner, S. A., & Franzkreb, K. E. (2002). Spatial autocorrelation and autoregressive models in ecology. *Ecological Monographs*, *72*(3), 445–463. https://doi.org/10.1890/0012-9615(2002)072[0445:SAAAMI]2.0.CO;2
Luts, J., Molenberghs, G., Verbeke, G., Van Huffel, S., & Suykens, J. A. K. (2012). A mixed effects least squares support vector machine model for classification of longitudinal data. *Computational Statistics and Data Analysis*, *56*(3), 611–628. https://doi.org/10.1016/j.csda.2011.09.008
Malik, F. (2018). Processing Data To Improve Machine Learning Models Accuracy. Retrieved December 19, 2018, from https://medium.com/fintechexplained/processing-data-to-improve-machine-learning-models-accuracy-de17c655dc8e
Meng, S. X., Huang, S., Vanderschaaf, C. L., Yang, Y., & Trincado, G. (2012). Accounting for serial correlation and its impact on forecasting ability of a fixed- and mixed-effects basal area model: A case study. *European Journal of Forest Research*, *131*(3), 541–552. https://doi.org/10.1007/s10342-011-0527-z
New York City Police Department. (2018). Overall Crime Continues to Drop in New York City Through First Quarter of 2018. Retrieved December 18, 2018, from https://www1.nyc.gov/site/nypd/news/pr0404/overall-crime-continues-drop-new-york-city-first-quarter-2018#/0
Newsday. (2018). Major crime in New York City, 2009-2015. Retrieved January 20, 2019, from http://data.newsday.com/long-island/data/crime/new-york-city-crime-rate%5Cnhttp://data.newsday.com/long-island/data/crime/new-york-city-crime-rate/
NYC Information Technology & Telecommunications. (2019). NYC Open Data. Retrieved February 1, 2019, from https://opendata.cityofnewyork.us/
Peck, C. C., & Dhawan, A. P. (1995). *Genetic Algorithms as Global Random Search Methods: An Alternative Perspective*. *Evolutionary Computation* (Vol. 3). https://doi.org/Doi 10.1162/Evco.1995.3.1.39
Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Édouard, D. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830. https://doi.org/10.1007/s13398-014-0173-7.2
Probst, P., Wright, M., & Boulesteix, A.-L. (2018). *Hyperparameters and Tuning Strategies for Random Forest*. Retrieved from https://arxiv.org/pdf/1804.03515.pdf
Rocha, A., Groen, T., Skidmore, A., Darvishzadeh, R., Willemen, L., Rocha, A. D., … Willemen, L. (2018). Machine Learning Using Hyperspectral Data Inaccurately Predicts Plant Traits Under Spatial Dependency. *Remote Sensing*, *10*(8), 1263. https://doi.org/10.3390/rs10081263
Roy, M.-H., & Larocque, D. (2012). Robustness of random forests for regression. *Journal of Nonparametric Statistics*, *24*(4), 993–1006. https://doi.org/10.1080/10485252.2012.715161
Santibanez, S. F., Kloft, M., & Lakes, T. (2015). Performance Analysis of Machine Learning Algorithms for Regression of Spatial Variables . A Case Study in the Real Estate Industry, (Bork 2015), 292–297.
Santibanez, S. F., Lakes, T., & Kloft, M. (2015). Performance analysis of some machine learning algorithms for regression under varying spatial autocorrelation. In *18th AGILE International Conference on Geographic Information Science*. Retrieved from https://agile-online.org/conference/proceedings/proceedings-2015
Sawada, M. (2001). Global Spatial Autocorrelation indices–Moran's I, Geary's C and the General Cross-Product Statistic. Retrieved August 28, 2018, from http://www.lpc.uottawa.ca/publications/moransi/moran.htm
Sayad, S. (2010). Support Vector Regression. Retrieved December 14, 2018, from http://www.saedsayad.com/support_vector_machine_reg.htm

Schug, F., Okujeni, A., Hauer, J., Hostert, P., Nielsen, J., & van der Linden, S. (2018). Mapping patterns of urban development in Ouagadougou, Burkina Faso, using machine learning regression modeling with bi-seasonal Landsat time series. *Remote Sensing of Environment*, *210*, 217–228. https://doi.org/10.1016/j.rse.2018.03.022

Seok, K. H., Shim, J., Cho, D., Noh, G.-J., & Hwang, C. (2011). Semiparametric mixed-effect least squares support vector machine for analyzing pharmacokinetic and pharmacodynamic data. *Neurocomputing*, *74*(17), 3412–3419. https://doi.org/10.1016/J.NEUCOM.2011.05.012

Shekhar, A. (2018). What Is Feature Engineering for Machine Learning? Retrieved December 19, 2018, from https://medium.com/mindorks/what-is-feature-engineering-for-machine-learning-d8ba3158d97a

Smola, A. J., & Sc Olkopf, B. (2004). A tutorial on support vector regression *. *Statistics and Computing*, *14*, 199–222. Retrieved from https://alex.smola.org/papers/2004/SmoSch04.pdf

Smolyakov, V. (2017). Ensemble Learning to Improve Machine Learning Results. Retrieved December 13, 2018, from https://blog.statsbot.co/ensemble-learning-d1dcd548e936

Steinwart, I., & Thomann, P. (2017). *liquidSVM: A Fast and Versatile SVM package*. Retrieved from http://arxiv.org/abs/1702.06899

The PostGIS Development Group. (2018). PostGIS 2.5.2dev Manual. Retrieved December 19, 2018, from https://postgis.net/docs/index.html

Üstün, B., Melssen, W. J., & Buydens, L. M. C. (2006). Facilitating the application of Support Vector Regression by using a universal Pearson VII function based kernel. *Chemometrics and Intelligent Laboratory Systems*, *81*(1), 29–40. https://doi.org/10.1016/j.chemolab.2005.09.003

Valenzuela, O., Zhang, M., & Selpi, S. (2017). Combining Support Vector Regression with Scaling Methods for Highway Tollgates Travel Time and Volume Predictions. In *Proceedings of International Work-Conference on Time Series Analysis (ITISE 2017)* (Vol. 1, pp. 411–421). Granada. Retrieved from https://research.chalmers.se/en/publication/251312

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Berlin, Heidelberg: Springer-Verlag.

Verbeke, G., Molenberghs, G., & Rizopoulos, D. (2010). Random effects models for longitudinal data. In *Longitudinal Research with Latent Variables* (pp. 37–96). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-11760-2_2

Wang, H., Guo, Y., Liu, Z., Liu, Y., & Hong, S. (2013). Using spatial autocorrelation and spatial autoregressive models to analyze the spatial pattern of aerosol optical depth and the affecting factors. In *Proceedings of the 12th International Conference on GeoComputation*. Retrieved from http://www.geocomputation.org/2013/papers/60.pdf

Wang, Y., Fan, Y., Bhatt, P., & Davatzikos, C. (2010). High-dimensional pattern regression using machine learning: From medical images to continuous clinical variables. *NeuroImage*, *50*(4), 1519–1535. https://doi.org/10.1016/J.NEUROIMAGE.2009.12.092

Warsito, B., Yasin, H., Ispriyanti, D., & Hoyyi, A. (2018). Robust geographically weighted regression of modeling the Air Polluter Standard Index (APSI). *Journal of Physics: Conference Series*, *1025*(1), 12096. https://doi.org/10.1088/1742-6596/1025/1/012096

Westfall, J. A. (2016). Strategies for the use of mixed-effects models in continuous forest inventories. *Environmental Monitoring and Assessment*, *188*(4). https://doi.org/10.1007/s10661-016-5252-0

Yang, H., Huang, K., Chan, L., King, I., & Lyu, M. R. (2004). *Outliers Treatment in Support Vector Regression for Financial Time Series Prediction*. https://doi.org/10.1007/978-3-540-30499-9_196

Yang, W., Deng, M., Xu, F., & Wang, H. (2018). Prediction of hourly PM2.5using a space-time support vector regression model. *Atmospheric Environment*, *181*(March), 12–19. https://doi.org/10.1016/j.atmosenv.2018.03.015

Zhang, D., Jie, ·, Sun, L., & Pieper, K. (2016). Bivariate Mixed Effects Analysis of Clustered Data with Large Cluster Sizes. *Statistics in Biosciences*, *8*, 220–233. https://doi.org/10.1007/s12561-015-9140-x

Zhao, J., Wang, Y., & Shi, W. (2018). Using Local Moran's I Statistics to Estimate Spatial Autocorrelation of Urban Economic Growth in Shandong Province, China (pp. 32–39). Springer, Singapore. https://doi.org/10.1007/978-981-13-0893-2_4

Zhigljavsky, A. A., & Pintér, J. (1991). Main Concepts and Approaches of Global Random Search. In *Theory of Global Random Search* (pp. 77–113). Dordrecht: Springer Netherlands. https://doi.org/10.1007/978-94-011-3436-1_3

Zuur, A. F., & Ieno, E. N. (2016). A protocol for conducting and presenting results of regression-type analyses. *Methods in Ecology and Evolution*, *7*(6), 636–645. https://doi.org/10.1111/2041-210X.12577

APPENDIX A

A 1. Detail experiments of non-lagged model

		Explanatory Variables				
Experiments Code	Random/ Fixed Variable (RV/FV)	Complaints	Temporal $(month)$	Complaints (dirty_ condition)	Complaints (noise_ vehicle)	Zip Code
V+MNL/WNL	FV	●		●	●	(OHE)
ME+MNL/WNL - 1	FV	●	●			C
	RV					
ME+MNL/WNL - 2	FV	●	●		●	C
	RV			●		
ME+MNL/WNL - 3	FV	●	●	●	●	C
	RV			●		
ME+MNL/WNL - 4	FV	●	●	●		C
	RV				●	
ME+MNL/WNL - 5	FV	●	●	●	●	C
	RV				●	
ME+MNL/WNL - 6	FV	●	●			C
	RV			●	●	
ME+MNL/WNL - 7	FV	●	●	●	●	C
	RV			●	●	
ME+MNL/WNL - 8	FV	●		●	●	C
	RV		●			
ME+MNL/WNL - 9	FV	●	●	●	●	C
	RV		●			

		Explanatory Variables				
Experiments Code	Random/ Fixed Variable (RV/FV)	Complaints	Temporal $(month)$	Complaints (dirty_ condition)	Complaints (noise_ vehicle)	Zip Code
ME+MNL/WNL -10	FV	●	●	●	●	C
	RV		●		●	
ME+MNL/WNL - 11	FV	●	●	●	●	C
	RV		●		●	
ME+MNL/WNL - 12	FV	●			●	C
	RV		●	●		
ME+MNL/WNL - 13	FV	●	●	●	●	C
	RV		●	●		
ME+MNL/WNL - 14	FV	●				C
	RV		●	●	●	
ME+MNL/WNL - 15	FV	●	●	●	●	C
	RV		●	●	●	

A 2. Detail experiments of lagged spatial features model

		Explanatory Variables				
Experiments Code	Random/ Fixed Variable (RV/FV)	Complaints	Temporal $(month)$	Spatial Lag $(t-1)$	LISA's Quadrant $(t-1)$	Zip Code
V+MWL/WWL	FV	●		●	●	(OHE)
ME+MWL/WWL - 1	FV	●	●		●	C
	RV					
ME+MWL/WWL - 2	FV	●	●	●		C
	RV					

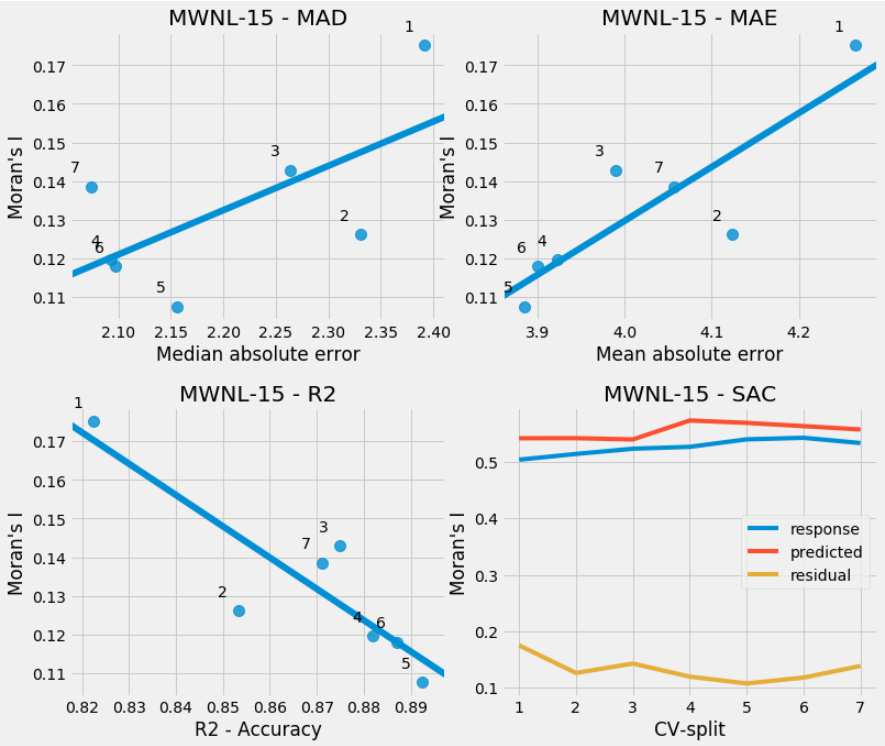
		Explanatory Variables				
Experiments Code	Random/ Fixed Variable (RV/FV)	Complaints	Temporal $(month)$	Spatial Lag $(t-1)$	LISA's Quadrant $(t-1)$	Zip Code
ME+MWL/WWL - 3	FV	●			●	C
	RV		●			
ME+MWL/WWL - 4	FV	●		●		C
	RV		●			
ME+MWL/WWL - 5	FV	●	●		●	C
	RV		●			
ME+MWL/WWL - 6	FV	●	●	●		C
	RV		●			
ME+MWL/WWL - 7	FV	●	●	●		C
	RV				●	
ME+MWL/WWL - 8	FV	●	●	●	●	C
	RV				●	
ME+MWL/WWL - 9	FV	●	●		●	C
	RV			●		
ME+MWL/WWL - 10	FV	●	●	●	●	C
	RV			●		
ME+MWL/WWL - 11	FV	●	●			C
	RV			●	●	
ME+MWL/WWL - 12	FV	●	●	●	●	C
	RV			●	●	
ME+MWL/WWL - 13	FV	●		●		C
	RV		●		●	

		Explanatory Variables				
Experiments Code	Random/ Fixed Variable (RV/FV)	Complaints	Temporal $(month)$	Spatial Lag $(t-1)$	LISA's Quadrant $(t-1)$	Zip Code
ME+MWL/WWL - 14	FV	●		●	●	C
	RV		●		●	
ME+MWL/WWL - 15	FV	●			●	C
	RV		●	●		
ME+MWL/WWL - 16	FV	●		●	●	C
	RV		●	●		
ME+MWL/WWL - 17	FV	●				C
	RV		●	●	●	
ME+MWL/WWL - 18	FV	●		●	●	C
	RV		●	●	●	
ME+MWL/WWL - 19	FV	●	●	●		C
	RV		●		●	
ME+MWL/WWL - 20	FV	●	●	●	●	C
	RV		●	●		
ME+MWL/WWL - 21	FV	●	●		●	C
	RV		●	●		
ME+MWL/WWL - 22	FV	●	●	●	●	C
	RV		●	●		
ME+MWL/WWL - 23	FV	●	●			C
	RV		●	●	●	
ME+MWL/WWL - 24	FV	●	●	●	●	C
	RV		●	●	●	

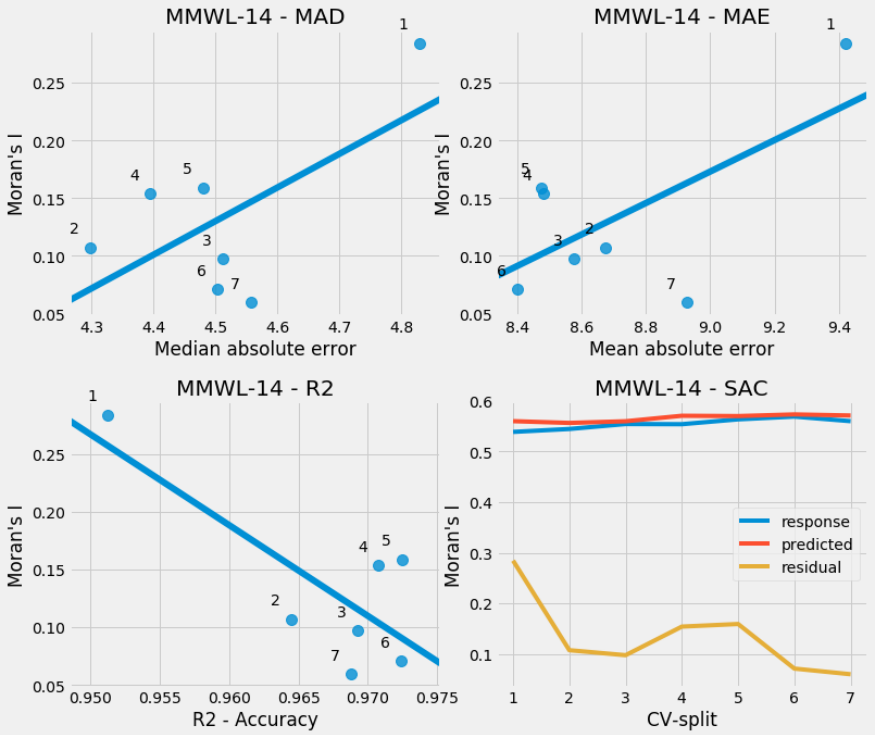
APPENDIX B

DETAIL CROSS-VALIDATION RESULTS

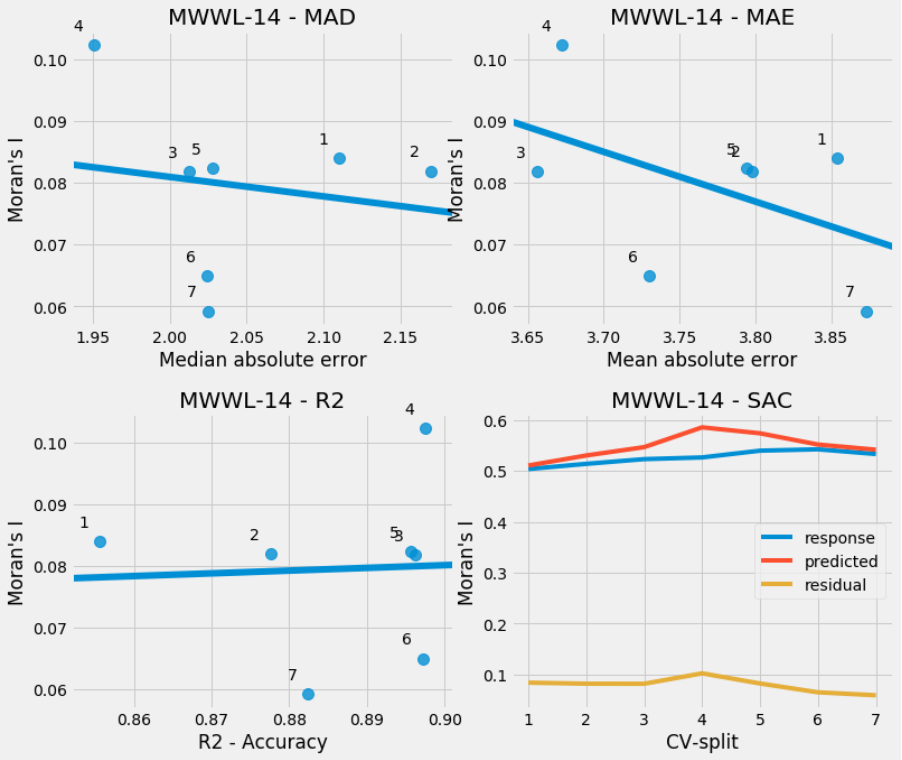
MIXED EFFECTS RANDOM FOREST



B 1. Detail performance of MWNL-15 model

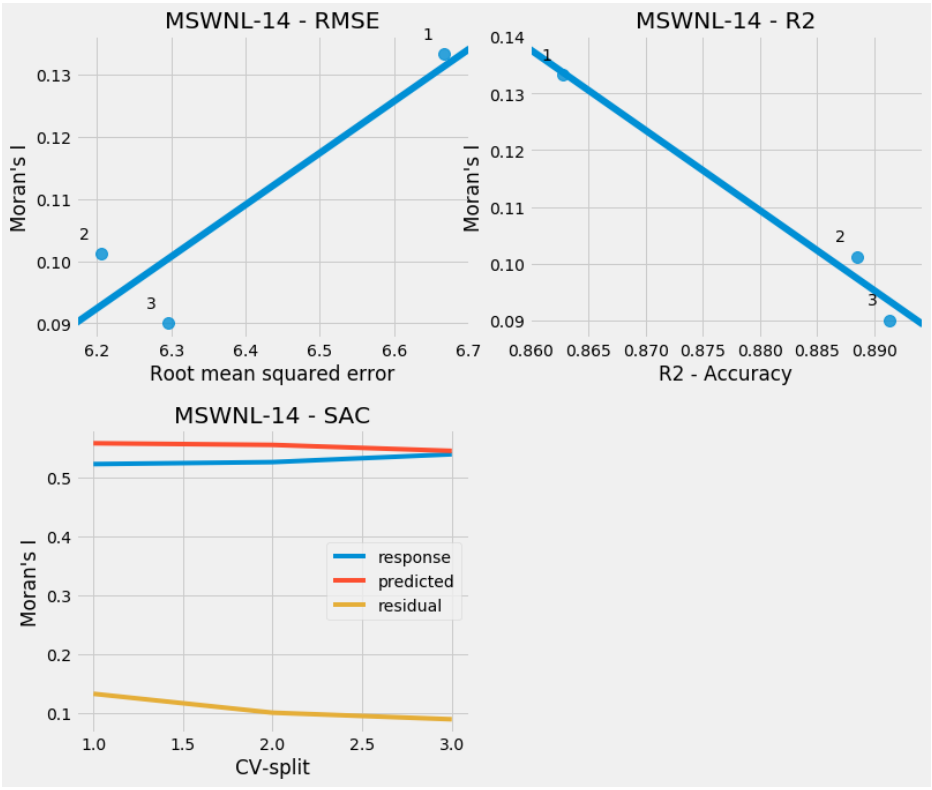


B 2. Detail performance of MMWL-14 model

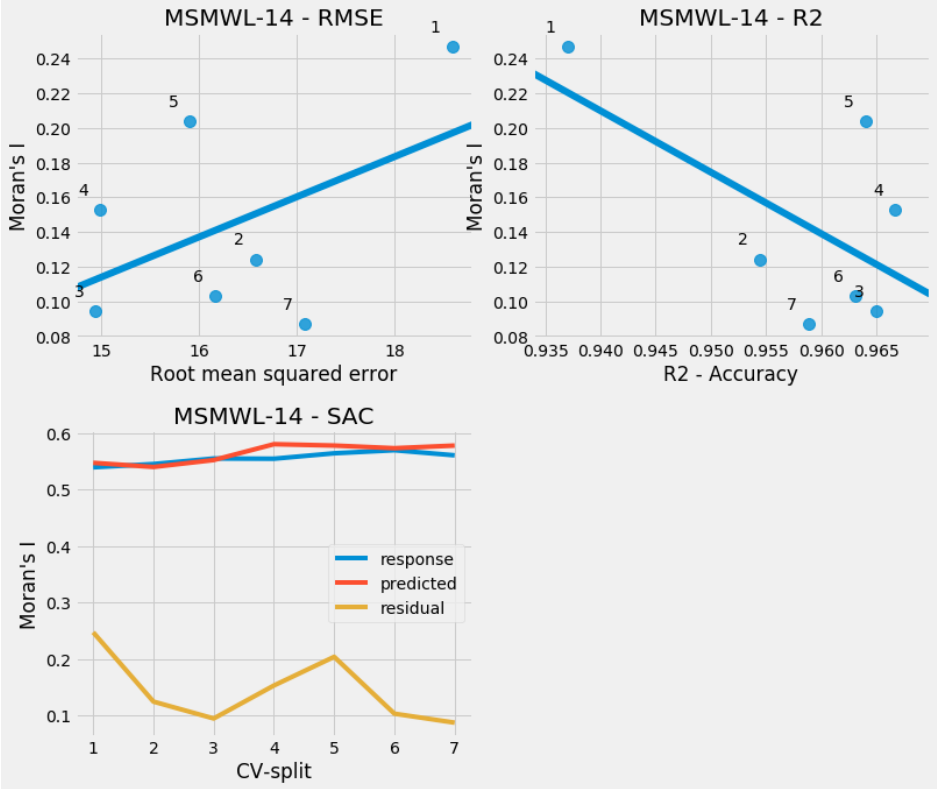


B 3. Detail performance of MWWL-14 model

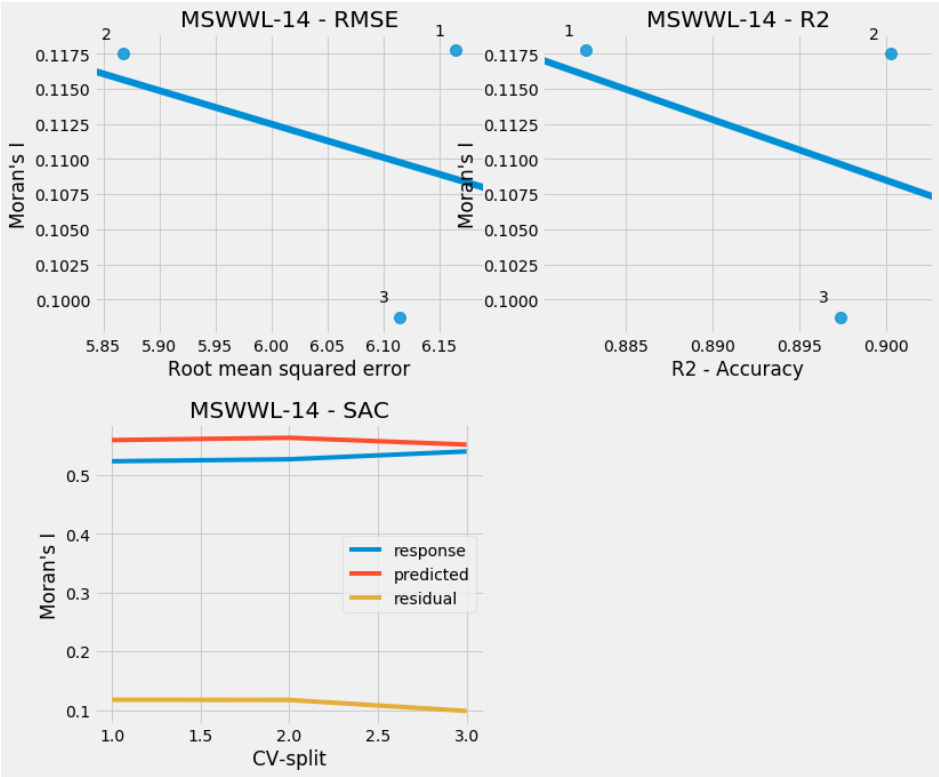
MIXED EFFECTS SUPPORT VECTOR REGRESSION



B 4. Detail performance of MSWNL-14 model



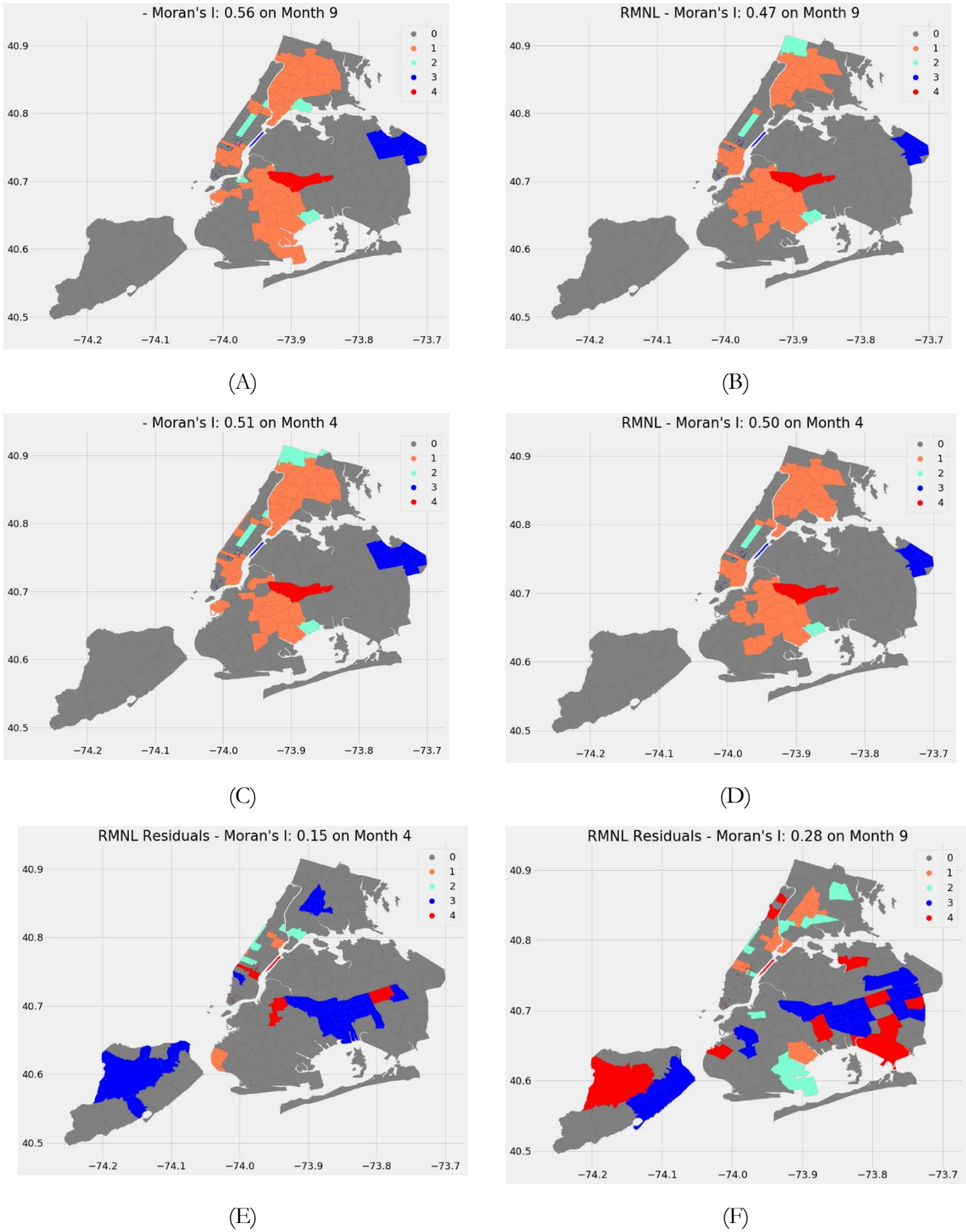
B 5. Detail performance of MSMWL-14 model



B 6. Detail performance of MSWWL-14 model

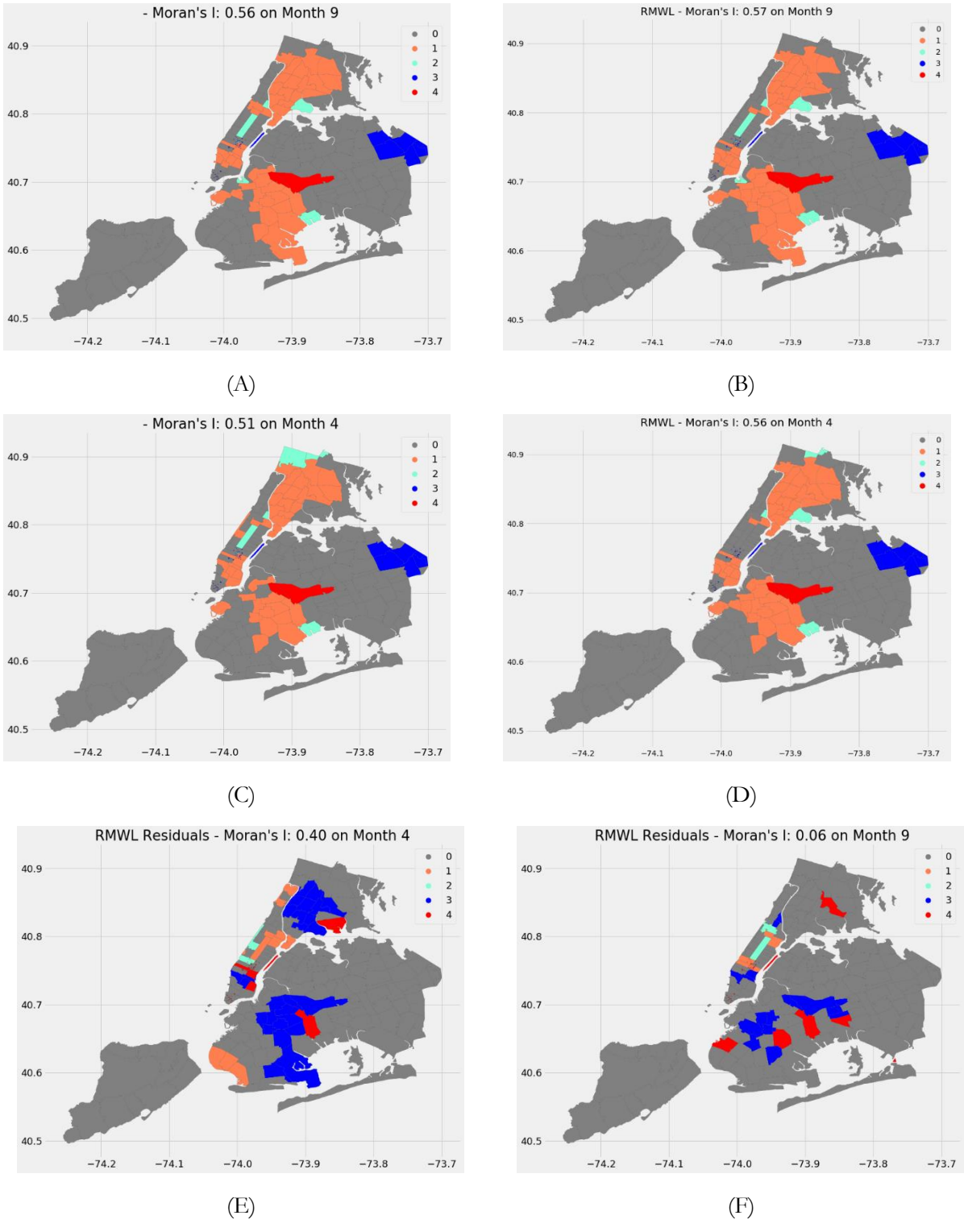
FINAL MODEL EVALUATION

RF – RMNL



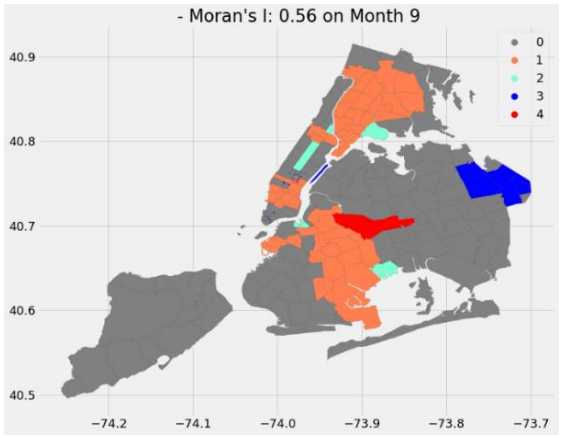
B 7. Spatial pattern of crime in New York City on particular month, SAC of each zip code measured using Local Moran's I, while SAC to entire area is measured using Global Moran's I. (A) The spatial pattern of the response variable which has the highest of SAC in 2017 (B) The corresponding predicted SAC pattern, on month 9 (C) The spatial pattern of response variable which has the lowest SAC in 2017, (D) The corresponding predicted SAC pattern on month 4 (E) SAC residuals RMNL on month 4 (F) SAC residuals RMNL on month 9.

RF - RMWL

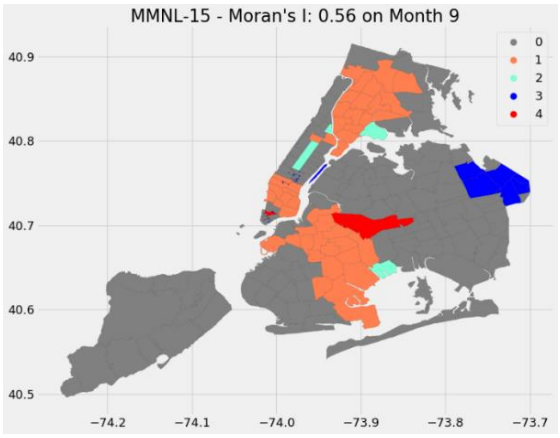


B 8. Spatial pattern of crime in New York City on particular month, SAC of each zip code measured using Local Moran's I, while SAC to entire area is measured using Global Moran's I. (A) The spatial pattern of the response variable which has the highest of SAC in 2017 (B) The corresponding predicted SAC pattern, on month 9 (C) The spatial pattern of response variable which has the lowest SAC in 2017, (D) The corresponding predicted SAC pattern on month 4 (E) SAC residuals RMWL on month 4 (F) SAC residuals RMWL on month 9.

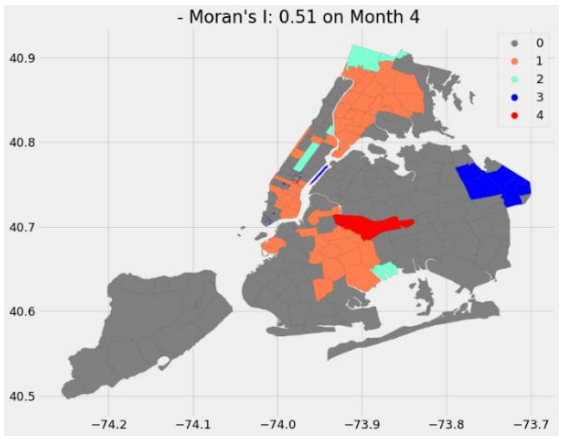
MERF – MMNL-15



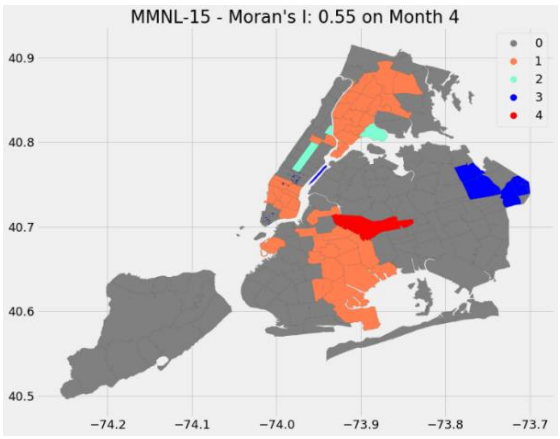
(A)



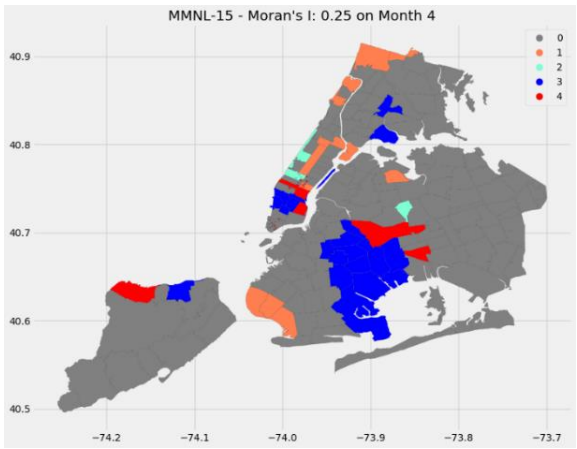
(B)



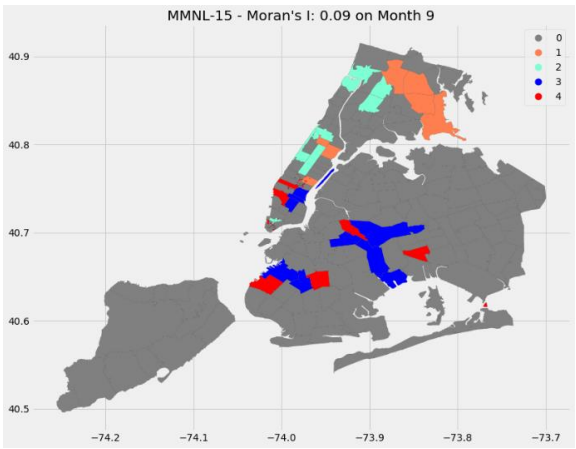
(C)



(D)



(E)



(F)

B 9. Spatial pattern of crime in New York City on particular month, SAC of each zip code measured using Local Moran's I, while SAC to entire area is measured using Global Moran's I. (A) The spatial pattern of the response variable which has the highest of SAC in 2017 (B) The corresponding predicted SAC pattern, on month 9 (C) The spatial pattern of response variable which has the lowest SAC in 2017, (D) The corresponding predicted SAC pattern on month 4 (E) SAC residuals MMNL-15 on month 4 (F) SAC residuals MMNL-15 on month 9.