ICIC Express Letters
Part B: Applications
Volume **11**, Number **4**, April **2020**
ICIC International ©2020 ISSN 2185-2766
pp. 333–339

STOCK PRICE MODELING USING LOCALIZED MULTIPLE KERNEL LEARNING SUPPORT VECTOR MACHINE

Hasbi Yasin[1], Rezzy Eko Caraka[2], Abdul Hoyyi[1] and Sugito[1]

[1]Department of Statistics
Diponegoro University
Tembalang, Central Java 50275, Indonesia
Hasbiyasin@live.undip.ac.id

[2]College of Informatics
Chaoyang University of Technology
No. 168, Jifeng East Road, Wufeng District, Taichung 41349, Taiwan
rezzyekocaraka@gmail.com

Received October 2019; accepted January 2020

Abstract. *Effectively and efficiently learning an optimal kernel is of great importance to the success of kernel method. Along with this line of research, many pioneering kernel learning algorithms have been proposed, developed and combined in many ways. This paper aims to explain the application of Localized Multiple Kernel Learning Support Vector Machine (LMKL-SVM) to predict the daily stock price of PT.XL Axiata Tbk (EXCL), PT.Indofood SuksesMakmur Tbk (INDF) and PT.Unilever Indonesia Tbk (UNVR) from January 2014 to May 2016. It can be concluded that LMKL-SVM has good performance to predict daily stock price with Mean Absolute Percentage Error (MAPE) produced all less than 2%.*
Keywords: Stock price, Multi kernel, Localized SVM, Time series

1. **Introduction.** The area of data mining and machine learning deals with the design of methods that can learn rules from data [1], adapt to changes, and improve performance with experience [2]. Also, to be one of the initial dreams of robust, efficient and applicable methods and reached minimum error, machine learning has become crucial to solve increasingly complex problems and become more integrated into many applications, precisely time series [3,4]. Machine learning theory also has close connections to issues in Economics [5-7]. Machine learning methods can be used in the design of auctions and other pricing mechanisms with guarantees on their performance [8]. Risk is the result of probabilistic world where there are no certainties and complexities abound. People use statistics to mitigate risk in decision making. Reliable knowledge about future can help investor make the right decision [9] with lower levels of risk [10]. Figure 1 represented as a statistician we must have abilities to capture data and make visualization to catch and present the insight with accurately reflecting the numbers and inappropriate visuals can create misinterpretation and misunderstanding. Using relevant methods with error minimum it could be the detection of pattern in the data to gain knowledge and make argument, interpretation and justification [11]. Support vector machines have become a subject of intensive study [12]. Many authors combine and enhance this method in time series [13], regression [14], classification and cluster. In this case, we can compute a string kernel from the sequence data and a Gaussian kernel upon the vectorial data, and learn the relative significance of the two kernels via the setting of multiple kernel learning. By multiple kernel learning [15], the relative importance of the kernels can be

DOI: 10.24507/icicelb.11.04.333

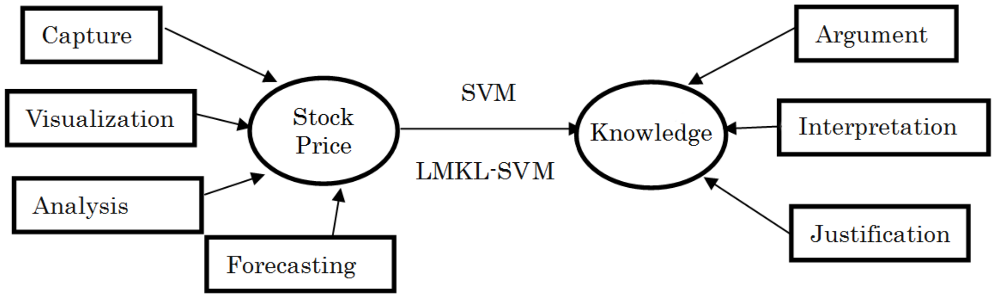

Figure 1. An illustration of stock price analysis using LMKL-SVM

evaluated together with the solution of the Support Vector Regression (SVR) [16]. Recently, multiple kernel learning has been automated for Support Vector Machine (SVM) classification using Semi Definite Programming (SDP) in optimization theory. However, the problem of multiple kernel learning on SV regression has not yet been examined. In this work, we formulate the SV regression problem as SMO [17], Quadratic Programming (QP) [18], and MOSEK optimization problems [19], so that kernel selection can be performed automatically for SV regression. In this paper we use support vector machines in the field of time series prediction. Brief introduction to Support Vector Regression (SVR) and adaptive machine learning algorithms can be viewed as a model for how individuals can or should adjust to change environment [20]. Moreover, the development of especially fast-adapting algorithms sheds light on how approximate equilibrium states might quickly be reached in a system, even when each individual has a large number of different possible choices. In the other direction, economic issues arise in machine learning when not only is the computer algorithm adapting to its environment, but it also is affecting its environment and the behavior of other individuals in it [21]. Connections between these two areas have become increasingly strong in recent years as both communities aim to develop tools for modeling. The remainder of the paper is organized as follows. Section 2 provides a review of the material and methods. Section 3 presents dicussion. Finally, conclusions and future research directions are indicated in Section 4.

2. **Methods.** Support Vector Regression (SVR) is part of Support Vector Machine (SVM) for regression case. SVR is also a method that can overcome the overfitting, so it will produce a good performance. Suppose there are l training data, $(\boldsymbol{x}_i, y_i)$ $i = 1, \ldots, l$ of the input data $\boldsymbol{x} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_l\} \subseteq \Re^{\mathrm{N}}$ and $\boldsymbol{y} = \{y_1, \ldots, y_l\} \subseteq \Re$. SVR is obtained by the method of regression function as follows:

$$f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}}\varphi(\boldsymbol{x}) + b \tag{1}$$

with $\boldsymbol{w}$ = vector of weight coefficient, $\varphi(\boldsymbol{x})$ = feature space, b = bias.

In order to obtain good generalization for the regression function, it can be done by minimizing the norm of $\boldsymbol{w}$. Hence there is need for the completion of the following optimization problem:

$$\min \left\{ \frac{1}{2} ||\boldsymbol{w}||^2 \right\} \tag{2}$$

with the provision of:

$$y_i - \boldsymbol{w}^{\mathrm{T}}\varphi(\boldsymbol{x}_i) - b \leq \varepsilon \tag{3}$$

$$\boldsymbol{w}^{\mathrm{T}}\varphi(\boldsymbol{x}_i) - y_i + b \leq \varepsilon, \quad i = 1, 2, \ldots, l$$

The loss function shows the relationship between an error and the subject penalties. Differences loss SVR function will produce different formulations [22]. There are two types of loss function used in this study, the ε-insensitive and quadratic loss function. Here is a mathematical formulation for the ε-insensitive loss function:

$$L(\boldsymbol{y}, f(\boldsymbol{x})) = \begin{cases} 0, & \text{for } |f(\boldsymbol{x}) - \boldsymbol{y}| < \varepsilon \\ |f(\boldsymbol{x}) - \boldsymbol{y}| - \varepsilon, & \text{otherwise} \end{cases} \quad (4)$$

The solution provided is:

$$\max \left\{ -\frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \left(\alpha_i - \alpha_i^*\right) \left(\alpha_j - \alpha_j^*\right) < \boldsymbol{x}_i, \boldsymbol{x}_j > + \sum_{i=1}^{l} \alpha_i (y_i - \varepsilon) - \alpha_i^* (y_i + \varepsilon) \right\} \quad (5)$$

or can be simplified into:

$$\max \left\{ -\frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \beta_i \beta_j < \boldsymbol{x}_i, \boldsymbol{x}_j > - \sum_{i=1}^{l} \beta_i y_i \right\} \quad (6)$$

with the provision of:

$$-C \le \beta_i \le C, \quad i = 1, \ldots, l \quad (7)$$

$$\sum_{i=1}^{l} \beta_i = 0$$

where $\beta_i = \alpha_i - \alpha_i^*$, $\beta_j = \alpha_j - \alpha_j^*$, $j = 1, 2, \ldots, l$, and C is a parameter which gives a tradeoff between model complexity and training error.

While the quadratic loss function is:

$$L(\boldsymbol{y}, f(\boldsymbol{x})) = (f(\boldsymbol{x}) - \boldsymbol{y})^2 \quad (8)$$

Produce a solution:

$$\max \left\{ -\frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \left(\alpha_i - \alpha_i^*\right) \left(\alpha_j - \alpha_j^*\right) < \boldsymbol{x}_i, \boldsymbol{x}_j > + \sum_{i=1}^{l} \left(\alpha_i - \alpha_i^*\right) y_i - \frac{1}{2C} \sum_{i=1}^{l} \left(\alpha_i^2 + \alpha_i^{*2}\right) \right\} \quad (9)$$

or can be simplified into:

$$\max \left\{ -\frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \beta_i \beta_j < \boldsymbol{x}_i, \boldsymbol{x}_j > + \sum_{i=1}^{l} \beta_i y_i - \frac{1}{2C} \sum_{i=1}^{l} \beta_i^2 \right\} \quad (10)$$

with the provision of:

$$\sum_{i=1}^{l} \beta_i = 0 \quad (11)$$

One method that can be used to optimize the hyperplane, which is to solve the quadratic programming with constraints set is Sequential Minimal Optimization (SMO). Sequential Minimal Optimization (SMO) algorithm is to solve the problem Quadratic Programming (QP) that arises during training on support vector machine. SMO is a simple algorithm that can solve the problem QP quickly on SVM. SMO algorithm is to solve the problems in the SVM-QP without using optimization measures QP numerically [23]. Instead, SMO chooses to resolve the smallest possible optimization problem involving two elements α_i the need to meet the limiting linear equations. Many decision problems facing individuals and companies can be cast as an optimization problem, i.e., making an optimal decision given some constraints specifying the possible decisions. As an example consider the problem of determining an optimal production plan. This can be formulated as maximizing a profit function given a set of constraints specifying the possible production plans. The advantages of MOSEK: (i) solve linear optimization problems using either an interior-point or a simplex optimizer, (ii) solve conic quadratic and semi definite optimization problems, (iii) solve convex quadratic optimization problems, (iv) handle convex quadratic constraints, (v) solve mixed-integer optimization problems, including linear, convex quadratic and conic quadratic problems, and (vi) solve linear least squares problems with linear constraints.

3. **Discussion.** In this research, we use the SVR with some of kernel functions simultaneously as called Multiple Kernel Learning Support Vector Regresion (MKL-SVR). MKL-SVR is one model that can capture the nonlinear pattern of financial time series data including data on stock returns. Research on MKL-SVR has developed rapidly. The model is based on modeling SVR by Cortes and Vapnik [24] using some kernel functions simultaneously. MKL-SVR modeling is the development of models SVR which involves some kernel functions, both with the same type or different. Specifically, Gönen and Alpaydın [25] developed a method Localized Multiple Kernel Learning Support Vector Regression (LMKL-SVR) as a part of Localized Multiple Kernel Learning Support Vector Machine (LMKL-SVM) for regression case. Two reasons for using a kernel [26]:

1) Turn a linear learner into a non-linear learner (e.g., RBF, polynomial, and sigmoid)
2) Make non-vectorial data accessible to learner (e.g., string kernels for sequences)

Localized Kernel Regression (LKR) is a multidimensional extension of kernel regression that uses a matrix of bandwidth parameters optimally selected for each input dimension instead of a single bandwidth parameter [27]. The problem of automatically selecting a set of optimal bandwidth parameters is an area of active research and one for which there is no clear solution. The main advantage of LMKL over canonical multiple kernel machines is the inherent regularization effect of the gating model [28]. Canonical methods learn sparse models as a result of regularization on the weight vector but the underlying complexity of the kernel function is the main factor for determining the model complexity. MKL can combine only different kernel functions and more complex kernels are favored over the simpler ones in order to get better performance. However, LMKL can also combine multiple copies of the same kernel

$$f_R(\boldsymbol{x}) = \sum_{m=1}^{p} \eta_m(\boldsymbol{x}|\boldsymbol{V})\langle \boldsymbol{\omega}_m, \phi_m(\boldsymbol{x})\rangle + b \tag{12}$$

And the optimization of the equation

$$\begin{aligned} \min \quad & \frac{1}{2}\sum_{m=1}^{p} ||\boldsymbol{\omega}_m||_2^2 + C\sum_{t=1}^{N}\left(\xi_i^+ + \xi_i^-\right) \\ \text{w.r.t.} \quad & \boldsymbol{\omega}_m \xi_i^+, \quad \xi_i^-, \quad \boldsymbol{V} \\ \text{s.t.} \quad & \epsilon + \xi_i^+ \geq y_i - f_R(\boldsymbol{x}_i) \quad \forall i \\ & \epsilon + \xi_i^- \geq f_R(\boldsymbol{x}_i) \quad \forall i \\ & \xi_i^+ \geq 0 \quad \forall i; \quad \xi_i^- \geq 0 \quad \forall i \end{aligned} \tag{13}$$

where C is a regularization parameter with ξ_i^+, ξ_i^- is the vector of slack variable and ϵ is the tube width. Optimization of the slack variable is not convex and nonlinear. By adding $\boldsymbol{V}$ we will get a convex optimization, and we can get a dual formulation:

$$\begin{aligned} \max \quad J(\boldsymbol{V}) = & \sum_{t=1}^{N} y_i\left(\alpha_i^+ - \alpha_i^-\right) - \epsilon\sum_{i=1}^{N}\left(\alpha_i^+ - \alpha_i^-\right) \\ & - \frac{1}{2}\sum_{t=1}^{N}\sum_{j=1}^{N}\left(\alpha_i^+ - \alpha_i^-\right)\left(\alpha_j^+ - \alpha_j^-\right) k_\eta(\boldsymbol{x}_i, \boldsymbol{x}_j) \end{aligned} \tag{14}$$

$$\text{w.r.t.} \quad \alpha_j^+, \quad \alpha_j^-$$

$$\text{s.t.} \quad \sum_{t=1}^{N} \left(\alpha_i^+ - \alpha_i^-\right) = 0$$
$$C \geq \alpha_i^+ \geq 0 \quad \forall i; \text{ and } C \geq \alpha_i^- \geq 0 \quad \forall i$$

Locally combined kernel function can be defined as

$$k_\eta(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{m=1}^{p} \eta_m(\boldsymbol{x}_i|\boldsymbol{V}) k_m(\boldsymbol{x}_i, \boldsymbol{x}_j) \eta_m(\boldsymbol{x}_j|\boldsymbol{V}) \tag{15}$$

in order to get the function as follows

$$f_R(\boldsymbol{x}) = \sum_{t=1}^{N} \left(\alpha_i^+ - \alpha_i^-\right) k_\eta(\boldsymbol{x}_i, \boldsymbol{x}_j) + b$$

In this paper we used the optimization algorithm using MOSEK, SMO, QP. On data stock prices of PT.EXCL, PT.INDF and PT.UNVR. The data used are the daily stock price from January 2014 to May 2016. We use locally linear kernel and quadratic kernel. Then, we used parameter $C = 10$, 25, 50, 75, 100 also tube width (ϵ) 0.5. We can see the plot of prediction vs actual in Figure 2 illustrates a good to fit (indicated by actual (–) and predicted (o)). The size of the error used in this study is the value of Mean Absolute

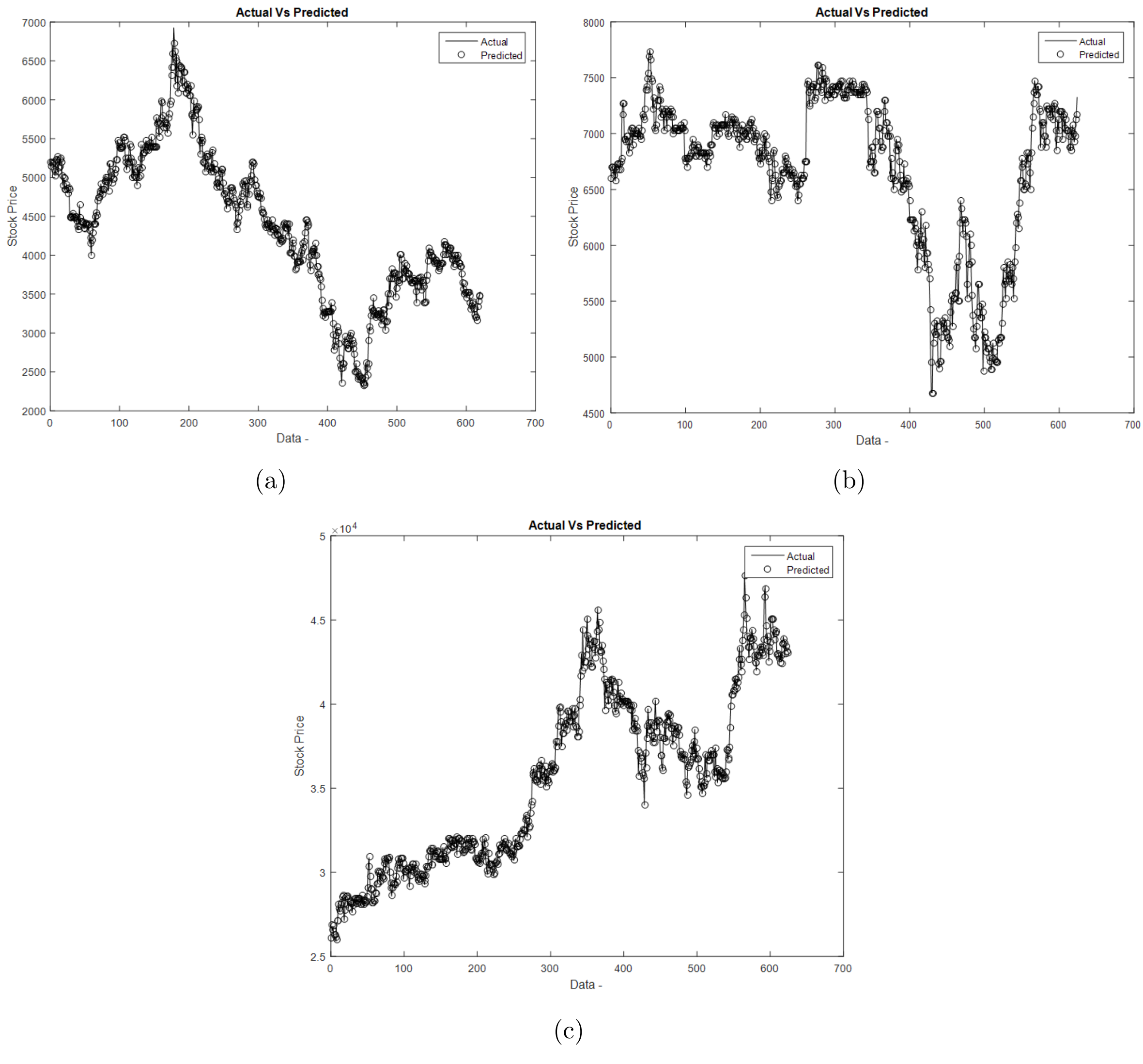

Figure 2. Plot prediction vs actual EXCL (a), INDF (b), UNVR (c)

TABLE 1. Simulation using LMKL-SVM

STOCK	N	C	ϵ	Optimization	MAPE (%)	RMSE
EXCL	621	100	0.5	SMO	1.87	112.6482
		100*	0.5*	QP*	1.87*	112.6312*
		75	0.5	SMO	1.88	113.1074
		50	0.5	MOSEK	1.89	114.8227
		10	0.5	SMO	1.91	113.7411
		25	0.5	SMO	1.87	112.7068
INDF	625	100	0.5	SMO	1.28	120.7054
		100	0.5	QP	1.28	120.7054
		75	0.5	SMO	1.28	120.7344
		50*	0.5*	MOSEK*	1.28*	120.6186*
		10	0.5	SMO	1.45	142.8360
		25	0.5	SMO	1.28	120.7866
UNVR	625	100*	0.5*	SMO*	1.25*	638.0822*
		100	0.5	QP	1.25	648.0822
		75	0.5	SMO	1.25	648.9264
		50	0.5	MOSEK	1.27	651.7972
		10	0.5	SMO	1.87	355504.2368
		25	0.5	SMO	1.37	693.7725

*BEST MODEL

Percentage Error (MAPE) and Root of Mean Squared Error (RMSE), so the formula of MAPE and RMSE can be expressed as: MAPE $= \left(\sum_{i=1}^{n} \frac{|y_i - \widehat{y_i}|}{y_i}\right) \times 100$ and RMSE $= \sqrt{\sum_{i=1}^{n} \frac{(y_i - \widehat{y_i})^2}{n}}$. Result of simulation LMKL-SVM can be seen in Table 1.

According to Table 1, we can see that the regularization parameters, tube width, and the type of optimization affect the learning process of LMKL-SVM. In this study, we only use two types of kernel locality, that is linear and quadratic. As a result of the simulation and analysis using LMKL-SVM, it can be concluded: the best optimization in EXCL stock price is QP with $C = 100$ and $\epsilon = 0.5$; then the best optimization in INDF stock prices is MOSEK with $C = 50$ and $\epsilon = 0.5$; also, the best optimization in UNVR stock price is SMO with $C = 100$, and $\epsilon = 0.5$.

4. **Conclusions.** Based on the analysis it concludes that LMKL-SVM had a very good performance in modeling with MAPE less than 2%. In this paper, we managed to make matlab GUI to simplify calculations and simulations using three different optimizations and also different kernel. Localized multiple kernel also known as Lazy Learning (LL) is a non-parametric. In a nutshell, localized multiple kernel SVR methods are more powerful than parametric methods if the assumptions for the parametric model cannot be met. The advantage of this method can be used to get a quick prediction with little error and easily implemented in the era of big data. Future studies will consider the feature selection and run the ensemble model with metaheuristic optimization.

Acknowledgment. This research fully supported by Institute for Research and Community Services (LPPM) Diponegoro University, under contract 329-44/UN7.P4.3/PP/2019.

REFERENCES

[1] E. Segel and J. Heer, Narrative visualization: Telling stories with data, *IEEE Trans. Vis. Comput. Graph.*, 2010.
[2] S. Sra, S. Nowozin and S. J. Wright, *Optimization for Machine Learning*, MIT Press, 2019.

[3] T. C. Fu, A review on time series data mining, *Engineering Applications of Artificial Intelligence*, vol.24, no.1, pp.164-181, 2011.

[4] R. E. Caraka, R. C. Chen, T. Toharudin, B. Pardamean, S. A. Bakar and H. Yasin, Ramadhan short-term electric load: A hybrid model of cycle spinning wavelet and group method data handling (CSW-GMDH), *IAENG Int. J. Comput. Sci.*, vol.46, no.4, pp.670-676, 2019.

[5] S. Rüping, SVM kernels for time series analysis, *Univ. Dortmund*, p.8, 2001.

[6] X. Wang, The plasticity and elasticity of stock price variations – Part 1: Theory and techniques, *International Journal of Innovative Computing, Information and Control*, vol.14, no.1, pp.261-277, 2018.

[7] X. Wang, The plasticity and elasticity of stock price variations – Part 2: Model analysis and application, *International Journal of Innovative Computing, Information and Control*, vol.14, no.2, pp.629-645, 2018.

[8] S. Ding, H. Zhao, Y. Zhang, X. Xu and R. Nie, Extreme learning machine: Algorithm, theory and applications, *Artif. Intell. Rev.*, 2015.

[9] Tarno, H. Yasin and B. Warsito, Constructing volatility model of portfolio return by using GARCH, *Glob. J. Pure Appl. Math.*, vol.12, no.2, pp.1201-1210, 2016.

[10] B. Warsito, R. Santoso, Suparti and H. Yasin, Cascade forward neural network for time series prediction, *Journal of Physics: Conference Series*, vol.1025, no.1, 2018.

[11] R. E. Caraka, R. C. Chen, H. Yasin, B. Pardamean, T. Toharudin and S. H. Wu, Prediction of status particulate matter 2.5 using state Markov chain stochastic process and HYBRID VAR-NN-PSO, *IEEE Access*, vol.7, pp.161654-161665, 2019.

[12] C. Chang and C. Lin, LIBSVM: A library for support vector machines, *ACM Trans. Intell. Syst. Technol.*, vol.2, pp.1-39, 2013.

[13] H. Yasin, R. E. Caraka, Tarno and A. Hoyyi, Prediction of crude oil prices using support vector regression (SVR) with grid search – Cross validation algorithm, *Glob. J. Pure Appl. Math.*, vol.12, no.4, 2016.

[14] Y. Han, K. Yang, Y. Yang and Y. Ma, Localized multiple kernel learning with dynamical clustering and matrix regularization, *IEEE Trans. Neural Networks Learn. Syst.*, vol.29, no.2, pp.486-499, 2018.

[15] A. Rakotomamonjy, F. R. Bach, S. Canu and Y. Grandvalet, Simple MKL, *J. Mach. Learn. Res.*, vol.9, pp.2491-2521, 2008.

[16] R. E. Caraka and S. A. Bakar, Evaluation performance of hybrid localized multi kernel SVR (LMKSVR) in electrical load data using 4 different optimizations, *J. Eng. Appl. Sci.*, vol.13, no.17, 2018.

[17] J. C. Platt, *Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines*, 1998.

[18] J. Nocedal and S. J. Wright, Quadratic programming, in *Numer. Optim.*, 2006.

[19] E. D. Andersen, *Markowitz Portfolio Optimization Using MOSEK*, 2009.

[20] D. D. Prastyo, F. S. Nabila, Suhartono, M. H. Lee, N. Suhermi and S. F. Fam, VAR and GSTAR-based feature selection in support vector regression for multivariate spatio-temporal forecasting, *Communications in Computer and Information Science*, pp.46-57, 2019.

[21] Suhartono, B. Maghfiroh and S. P. Rahayu, Hybrid VARX-SVR and GSTARX-SVR for forecasting spatio-temporal data, *Int. J. Innov. Technol. Explor. Eng.*, vol.8, no.4, pp.212-218, 2019.

[22] R. E. Caraka, S. A. Bakar and M. Tahmid, Rainfall forecasting multi kernel support vector regression seasonal autoregressive integrated moving average, *AIP Conf. Proc.*, vol.020014, no.6, 2019.

[23] A. Karatzoglou, A. Smola, K. Hornik, and A. Zeileis, Kernlab – An S4 package for kernel methods in R, *J. Stat. Softw.*, vol.11, no.9, pp.1-20, 2004.

[24] C. Cortes and V. Vapnik, Support-vector networks, *Mach. Learn.*, vol.20, no.3, pp.273-297, 1995.

[25] M. Gönen and E. Alpaydın, Multiple kernel learning algorithms, *J. Mach. Learn. Res.*, vol.12, pp.2211-2268, 2011.

[26] R. E. Caraka, S. A. Bakar, B. Pardamean and A. Budiarto, Hybrid support vector regression in electric power load during national holiday season, *Innovative and Creative Information Technology (ICITech)*, pp.1-6, 2017.

[27] R. E. Caraka, S. A. Bakar, M. Tahmid, H. Yasin and I. D. Kurniawan, Neurocomputing fundamental climate analysis, *Telkomnika*, vol.17, no.4, pp.1818-1827, 2019.

[28] F. Dinuzzo, M. Neve, G. De Nicolao and U. P. Gianazza, On the representer theorem and equivalent degrees of freedom of SVR, *J. Mach. Learn. Res.*, vol.8, pp.2467-2495, 2007.