Kernel-Based Machine Learning Methods for Modeling Daily Truck Volume at Seaport Terminals

Yuanchang Xie*, Ph.D.
Assistant Professor
Civil and Mechanical Engineering Technology
South Carolina State University
Orangeburg, SC 29117
Tel: (803) 536-8321
Fax: (803) 516-4693
E-mail: yxie@scsu.edu

Nathan Huynh, Ph.D.
Assistant Professor
Department of Civil and Environmental Engineering
University of South Carolina
Columbia, SC 29208
Tel: (803) 777-8947
Fax: (803) 777-0670
Email: huynhn@cec.sc.edu

* Corresponding author

Paper submitted for consideration of presentation at the 51st Transportation Research Forum

Word count: 4,797 + 2 Tables + 7 Figures = 7,047 Words

Submission Date
October 1, 2009

Kernel-Based Machine Learning Methods for Modeling Daily Truck Volume at Seaport Terminals

Yuanchang Xie and Nathan Huynh

ABSTRACT

The heavy truck traffic generated by major seaports can have huge impacts on local and regional transportation networks. Transportation agencies, port authorities, and terminal operators have a need to know in advance the truck traffic in order to accommodate them accordingly. Several previous studies have developed models for predicting the daily truck traffic at seaport terminals using terminal operations data. In this study, two kernel-based supervised machine learning methods are introduced for the same purpose: Gaussian Processes (GP) and ε-Support Vector Machines (ε-SVMs). They are compared against the Multilayer Feed-forward Neural Networks (MLFNNN) model, which was used in past studies, to provide a comparison of their relative performance. The model development is done using the data from Bayport and Barbours Cut (BCT) container terminals at the Port of Houston. Truck trips generated by import and export activities at the two terminals are investigated separately, generating four sets of data for model testing and comparison. For all test datasets, the GP and ε-SVMs models perform equally well and their prediction performance compares favorably to that of the MLFNN model. On a practical note, the GP and ε-SVMs models require less effort in model fitting compared to the MLFNN model. The strong performance of the GP and ε-SVMs models relative to the commonly used MLFNN model suggest that they can be considered as alternative approaches to the MLFNN in other predictive applications.

KEY WORDS: Ports, Freight, Drayage, Support Vector Machines, Gaussian Processes, Neural Networks.

1. INTRODUCTION

Ports are one of the major truck trip generators. The heavy truck traffic generated by ports presents considerable challenges to the various stakeholders (i.e. transportation planning agencies, port authorities, terminal operators) in their efforts to reduce emissions, mitigate congestion, and increase productivity so that growing cargo flows can coexist with port and terminal area communities. Recognizing the need to have better forecasting models of the daily truck trips that move through a terminal, several studies have developed truck trip models to assist transportation planning agencies, port authorities, and terminal operators. The goal of these studies is to provide models that are capable of accurately predicting future truck traffic, so that traffic engineers can better design traffic control strategies to mitigate the adverse impact of truck traffic on road networks and terminal operators can better manage handling equipment, terminal gate operations, work scheduling, and staffing requirements. With the availability of predicted truck traffic, simulation studies can be done to identify potential congested areas that need to be improved (*1*).

Past studies on this subject include the work of Al-Deek et al. who developed linear regression models to predict truck traffic for the Port of Miami (*2*). In their study, two separate linear regression models were developed, one for inbound traffic (export drop-off) and one for outbound traffic (import pickup). Daily inbound and outbound truck volumes were used as the dependent variables in the two models. Only one independent variable was considered in each of the mentioned regression model, which was the total number of loaded/unloaded freight units for one or several days. For the Port of Miami, truck traffic occurs only from Monday to Friday, while vessel loading and unloading activities occur everyday. The 24/7 vessel operations and Monday-Friday gate operations are typical of many ports in the U.S (*3*). To solve this discrepancy between input and output dimensions, Al-Deek et al. (*2*) used different strategies to combine the raw data such that each week produces three data points for the regression analysis.

To further explore the potential nonlinear relationship between truck trips and port operations activities, Al-Deek (*4*) introduced a neural network model and compared it with a linear regression model developed in (*2*). In this study, Al-Deek (*4*) employed a Multilayer Feed-forward Neural Network (MLFNN) model with two hidden layers and multiple output nodes in the output layer. The output nodes (can be considered as dependent variables) were used to represent total daily inbound/outbound freight traffic and transportation modes (e.g., rail, truck). In addition to the loaded and unloaded container data for the same day, loading and unloading activities for adjacent days were also considered as inputs to the neural network model. Different from the combination strategy used in the previous study (*2*), Al-Deek (*4*) used multiple inputs from several days to predict the daily inbound/outbound truck traffic. The developed neural network model was applied to the same data used in (*2*). Al-Deek concluded that better results were obtained from the neural network model. Neural network models were later applied to several other ports in Florida, including the Port of Tampa, Port Canaveral, and Port Everglades (*5,6*).

In addition to MLFNNs, other types of models have also been experimented with for port truck traffic forecasting. Sideris et al. (*7*) developed a container movement (drop-offs and pickups) model, which used historical terminal operations data to fit two empirical probability density functions (PDFs) for import and export container dwell times, respectively. The empirical PDFs were then used in conjunction with the vessel arrival and departure information to predict container movements. Sarvareddy et al. (*8*) developed a Fully Recurrent Neural Network (FRNN) model and compared it with MLFNNs. The results showed that the MLFNN model produced an accuracy of 84.6% and the FRNN model's accuracy was 71.85%. However, the authors did not specify how the accuracy data were calculated. Also, the MLFNN and FRNN models were not compared based on the same training and testing datasets. In another study, a time series model was integrated into MLFNNs to predict future truck trip generations by Al-Deek (*9*). He concluded in this study that for both inbound and outbound truck traffic, a neural network model without hidden layers and with a linear function in the output layer performed the best. Research has also been done to investigate truck trip generations from the planning perspective. For instance, Holguin-Veras et al. (*10*) used a linear regression model to establish relationships between daily truck traffic (dependent variable) and underlying factors (explanatory variables): total number of twenty-feet equivalent units (TEU) per year, number of container handled by trucks per year, area of the terminal in acres, and number of container berths.

This paper presents the development and evaluation of two kernel-based machine learning methods for predicting truck trips at seaports, Support Vector Machines (SVM) and Gaussian Processes (GP) models, using port operations data. These two models can be readily adapted for predicting truck traffic for planning applications using data such as sizes of storage areas and number of berths as inputs. The performance of the two new models will be compared with that of the MLFNN model. The rest of this paper is organized as follows. Section 2 introduces the SVM and GP models. Section 3 describes the data collection and descriptive analysis. Model testing is discussed in Section 4. Section 5 presents the analysis of results. Section 6 summarizes and discusses the findings of this research. Future work is described in Section 7.

2. METHODOLOGY

Among the existing port truck traffic forecasting methods, neural networks are the most widely used (*4,5,6,8,9*). Neural networks have been extensively researched for other transportation applications such as traffic flow modeling and forecasting (*11*). One major reason for the popularity of neural networks is that they have very strong function approximation ability (*12*) and can better model the potential nonlinear relationship between port operations data and inbound/outbound truck traffic volumes. Also, applying neural networks does not require specifying an explicit model formulation as is required by many other methods. Although neural networks have many attractive features, applying neural network models is not an easy task. Many challenging decisions have to be made -properly in regard to model training and selection, such as network architectures, type of transfer (activation) functions, learning rate, and number of hidden neurons (*13*) in order to obtain a valid model. Furthermore, cautions must be taken during the training of neural networks to prevent overfitting the training data and to avoid the solution from reaching local mimima.

To address the problems with neural networks implementations, SVM models have been introduced (*14,15*). Similar to neural networks, SVM models also have superior function approximation ability and do not require the specification of explicit model formulations. In addition, SVM models are developed based on the structural risk minimization (SRM) principle (*16*), as opposed to the empirical risk minimization (ERM) principle used in conventional neural networks. Hence, SVM models theoretically can better solve the overfitting problem and have better generalization ability than the conventional neural networks. Another important feature of SVM models is that they can guarantee a globally optimal solution for given training datasets (*14*,*15*).

Another method introduced in this paper to estimate truck trips is Gaussian Process (GP). GP models recently have attracted considerable attention in the machine learning community. They have been extensively used for regression and classification applications due to their strong function approximation ability (*17*,*18*). GP's formulation is based on a full Bayesian framework, which provides GP models with excellent generalization ability. This full Bayesian framework also enables GP models to generate statistically interpretable predictions. Compared to MLFNNs and SVM, fitting GP models is relatively easier. A brief introduction to SVM and GP models is presented Section 2.

2.1. Support Vector Machines (SVM)

In this study, an ε-Support Vector Machines (ε-SVM) model is adopted. Assume for a port truck trip forecasting problem with N inputs $\{x(i)\}_{i=1}^{N}$ and outputs $\{y(i)\}_{i=1}^{N}$, where $x(i) \in R^{In}$ and $y(i) \in R^1$, the ε-SVM model first maps the inputs from a *In*-dimensional space into a higher *h*-dimensional space using a function $\Phi(x(i))$ such that the potential nonlinear relationship between $x(i)$ and $y(i)$ can be linearized. In this new and higher input dimension, the estimation function of output $y(i)$ is

$$\hat{y}(i) = f(x(i)) = w^T \Phi(x(i)) + b \tag{1}$$

where $w \in R^h$ and $b \in R^1$ are coefficients to be estimated by solving the following optimization problem (*16*,*19*)

$$\text{Min} \quad R = \frac{1}{2} w^T w + \frac{C}{N} \sum_{i=1}^{N} \left(\xi_i + \xi_i^* \right)$$

$$\text{Subject to} \begin{cases} w^T \Phi(x(i)) + b - y(i) \le \varepsilon + \xi_i \\ y(i) - w^T \Phi(x(i)) - b \le \varepsilon + \xi_i^* \\ \xi_i^*, \xi_i \ge 0, \qquad i = 1, \dots, N \end{cases} \tag{2}$$

where ξ_i and ξ_i^* are slack variables; C is a regularization parameter; the superscript T in Eq. (2) means transpose; and ε is a soft margin loss parameter. As shown in Figure 1, ξ_i or ξ_i^* can be greater than zero only when the difference between $\hat{y}(i)$ and $y(i)$ is larger than ε. In other words, the minor differences ($\le \varepsilon$) between the observed and predicted values are not penalized in the objective function of Eq. (2). The optimization problem in Eq. (2) can be solved more efficiently in its dual form shown in Eq. (3).

$$\text{Min} \quad R = \frac{1}{2}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)K(x(i), x(j)) + \varepsilon \sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i(\alpha_i - \alpha_i^*)$$

$$\text{Subject to} \begin{cases} \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0 \\ \alpha_i, \alpha_i^* \in [0, C], i = 1, \dots, N \end{cases} \tag{3}$$

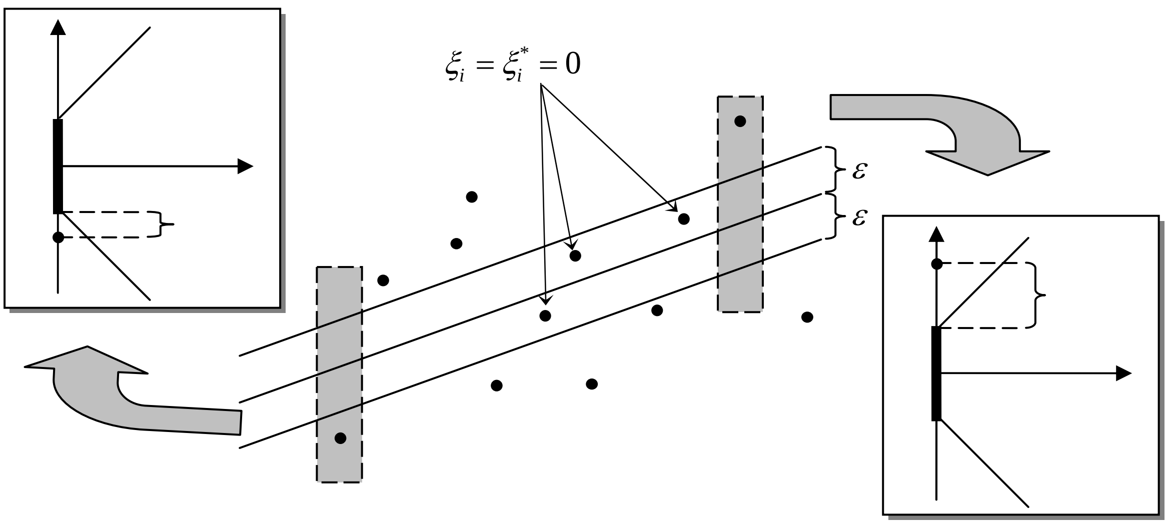

FIGURE 1 Soft margin loss parameter in ε-SVM.

Once the dual problem is solved, the prediction function can be written as (*19*)

$$\begin{aligned} \hat{y} = f(x) &= \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)\Phi(x(i))^T \Phi(x) + b \\ &= \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)K(x(i), x) + b \end{aligned} \tag{4}$$

where $K(x(i), x(j)) = \Phi(x(i))^T \Phi(x(j))$ is a kernel function to be specified. There are several types of kernel functions, including linear, polynomial, radial basis, sigmoid, and Automatic Relevance Determination (ARD) kernel functions. In this study, the sigmoid kernel function is selected for the ε-SVM model due to its better performance.

2.2. Gaussian Processes (GP)

GP models can be considered as extensions of the Bayesian linear regression models. Similar to ε-SVMs, GP models also require a mapping from the original input space into a new space of higher dimension. The new regression function is shown in Eq. (5).

$$\hat{y} = f(x) = w^T \Phi(x) + \varepsilon \tag{5}$$

where w is the regression parameters and $\Phi(\cdot)$ is the mapping function. ε in Eq. (5) is a noise term which follows an independent and identically distributed Gaussian distribution $N(0,\sigma^2)$. It is assumed that w also follows a Gaussian distribution with a mean of zero and covariance matrix Σ (i.e. $N(0,\Sigma)$). Given the assumed prior distributions and the observed input (*X*) and output (*y*), the posterior distribution of the regression parameters w can be calculated by the Bayes' rule in Eq. (6).

$$p(w \mid X, y) = \frac{p(y \mid X, w)\, p(w)}{p(y \mid X)} \tag{6}$$

For a test input x^*, the predicted result $f^* = f(x^*)$ will be the average output over all possible regression parameters. The distribution of f^* is shown in Eq. (7).

$$p(f^* \mid x^*, X, y) = \int p(f^* \mid x^*, w)\, p(w \mid X, y)\, dw \tag{7}$$

The mean and variance of f^* can be derived analytically as shown in Eqs. (8) and (9).

$$\mu(f^*) = K(x^*, X)[K(X, X) + \sigma^2 I]^{-1} y \tag{8}$$

$$\mathrm{var}(f^*) = K(x^*, x^*) - K(x^*, X)[K(X, X) + \sigma^2 I]^{-1} K(X, x^*) \tag{9}$$

Similar to ε-SVMs, kernel functions are introduced into the prediction function of GP models (e.g., Eqs. 8 and 9). For the GP model used in this study, an ARD kernel function is used.

3. DATA DESCRIPTION

To test the proposed models, data from two container terminals at the Port of Houston, Bayport and Barbours Cut (BCT), are used. The datasets include detailed daily operational data spanning seven months, from June 1, 2008 to January 30, 2009. Microsoft Access is used to aggregate the raw data into daily totals, including count of discharged containers, count of loaded containers, number of truck drop-offs, and number of truck pickups. In addition, the dwell time distributions for import and export containers are extrapolated from the containers' in and out times (see Figures 2 and 3). It can be seen that few containers are picked up/dropped off on the same day as they are discharged/loaded. On average the discharged containers are stored at the terminals for 5.3 days before they are picked up, and the export containers are shipped to the terminals around 6 days before they are loaded onto the vessels. Most import and export containers stay at the two terminals for less than 12 days.

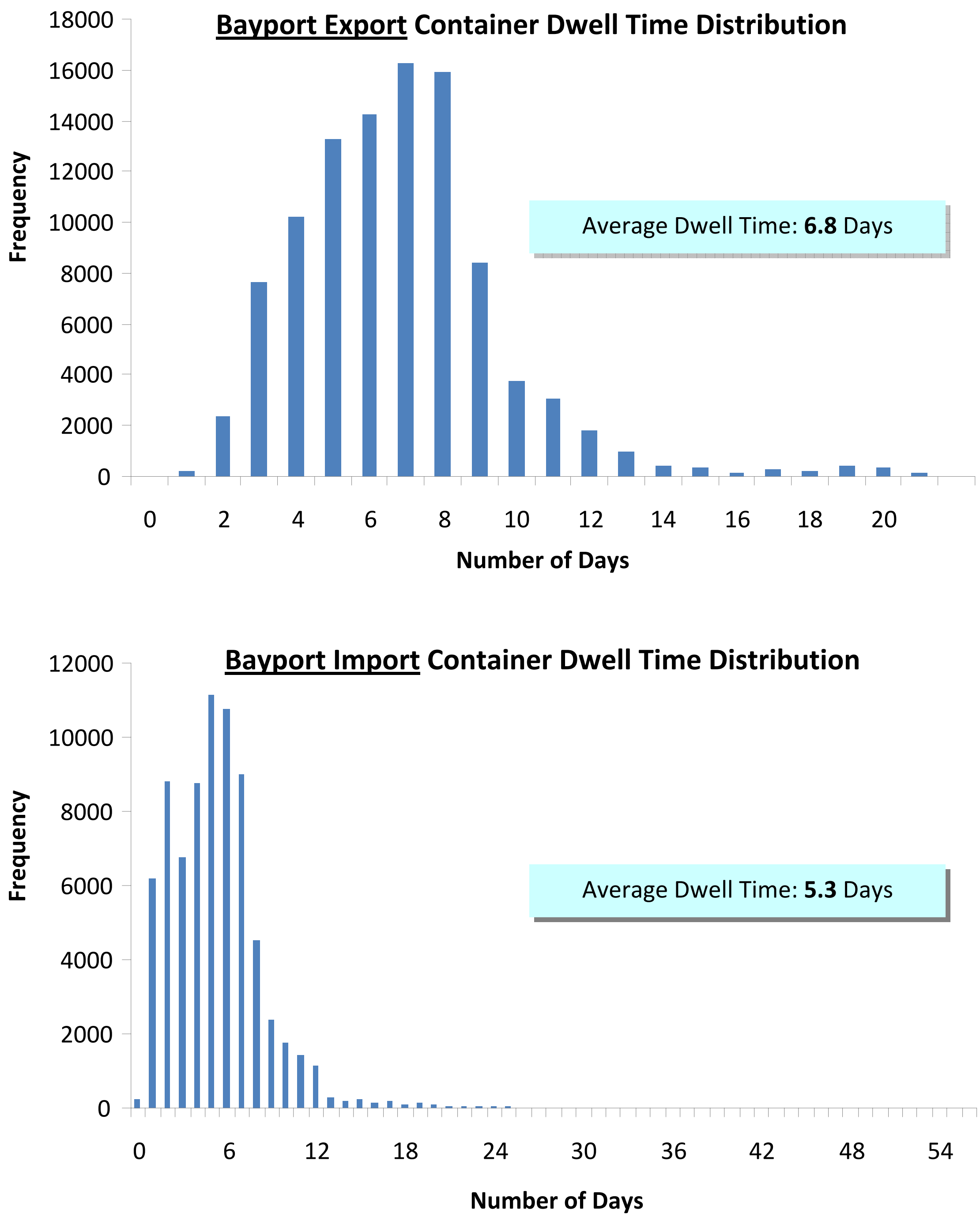

FIGURE 2 Dwell time distributions for Bayport.

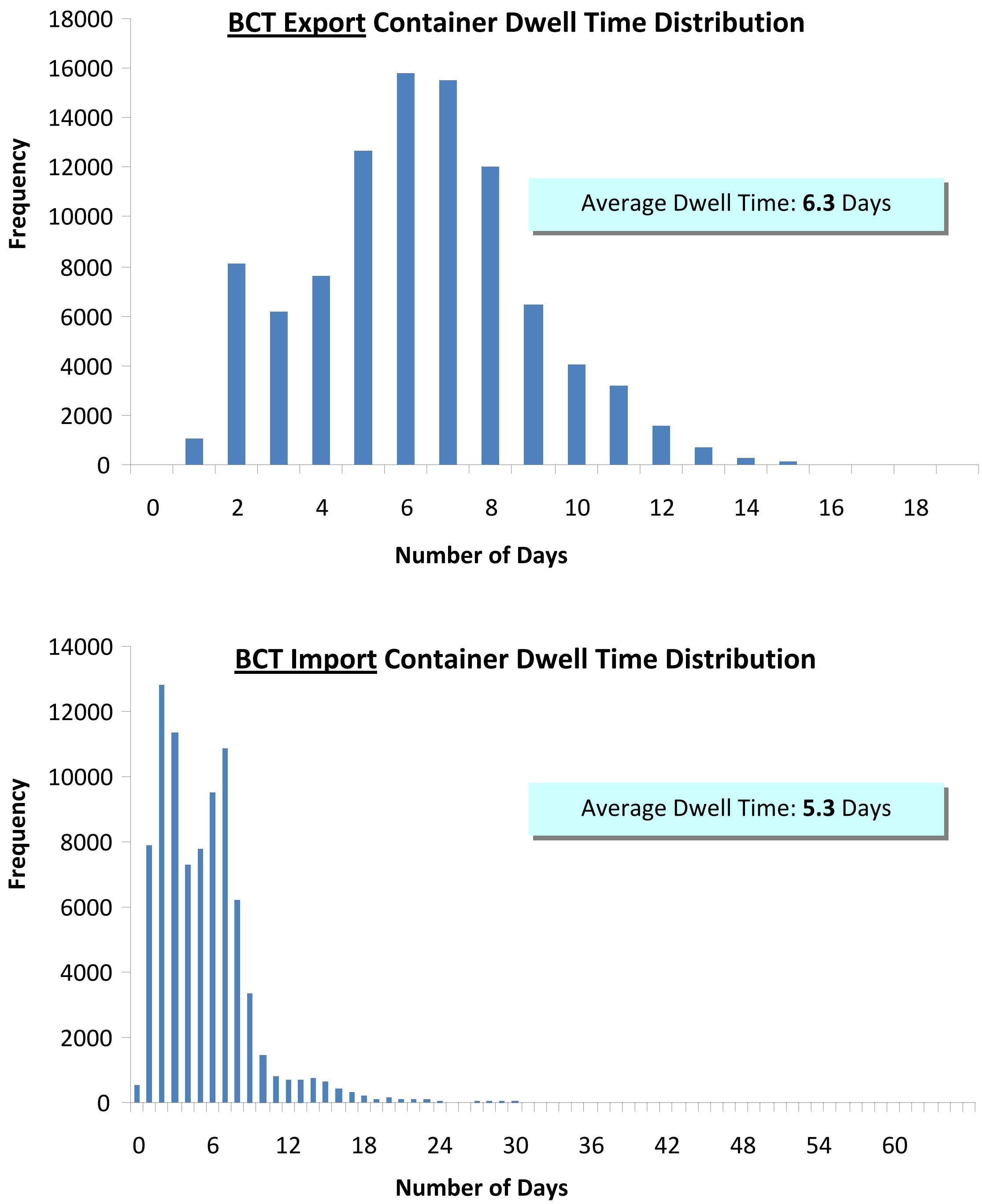

FIGURE 3 Dwell time distributions for BCT.

4. MODEL TESTING

The truck trips generated by import and export activities are modeled separately for each terminal. This results in four sets of data for model comparison, which are Bayport Export, Bayport Import, BCT Export, and BCT Import. For truck traffic generated by import activities, the dependent variable (model output) is the number of import pickups (I_i). Since most import containers stay at the terminals for less than 12 days, the corresponding independent variables (model input) are the numbers of discharged containers during each of the previous 12 days ($D_{-1},...,D_{-12}$). Similarly for truck traffic generated by export activities, the dependent variable is the number of export drop-offs (E_i). The corresponding independent variables are the numbers of loaded containers delivered to the terminal on each of the next 12 days ($L_1,...,L_{12}$). The same input and output data are used to train and test the MLFNN, ε-SVM, and GP models.

Missing and abnormal data points caused by holidays and unusual events such as Hurricane Ike are removed from the original data. This results in 150, 154, 139, and 148 days of data for the Bayport Export, Bayport Import, BCT Export, and BCT Import datasets, respectively. Each dataset is randomly separated into three subsets for training, validation, and testing. The training dataset is used to train the models. For the MLFNN and ε-SVM models, the validation dataset is used to find the best parameters such as C and to prevent overfitting the training data. For the GP model, the validation dataset is not required and it can actually be combined with the training dataset for training purpose. However, to provide a comparable basis for model comparison, the GP model is only trained using the training dataset.

To make the best use of the available data and also to further investigate the impact of different training, validation, and testing data sizes on model performance, two test scenarios are considered. The first scenario uses 70 data points for training, 30 data points for validation, and the rest data points for testing. The second scenario uses 80 data points for training, 40 data points for validation, and the rest data points for testing. For each test scenario, a total of 12 models are fitted and tested.

4.1. MLFNN Model

MLFNNs have been proven to be universal approximators, which means theoretically they can approximate any function with arbitrary accuracy (*12*). However, one byproduct of this superior function approximation ability is overfitting the training data. When overfitting happens, the fitted model can have very small prediction error on the training data. However, when applying the model to the testing dataset, the prediction error can be very large. Several strategies have been developed to address the overfitting problem. An early-stopping strategy is adopted in this study. As shown in Figure 4, at the initial training stage both the training and validation errors decrease drastically as the number of training iterations increases. After a certain number of training iterations, the training error keeps decreasing, while the validation error increases. In this study, the following criterion is used to stop the training process. Starting from any training iteration, if the validation error does not improve after 100 iterations, the training process is terminated and the trained network corresponding to the lowest validation error is chosen as the final training output. For all MLFNN models, the learning rate is set to be 0.01 and one hidden layer with six hidden neurons is considered.

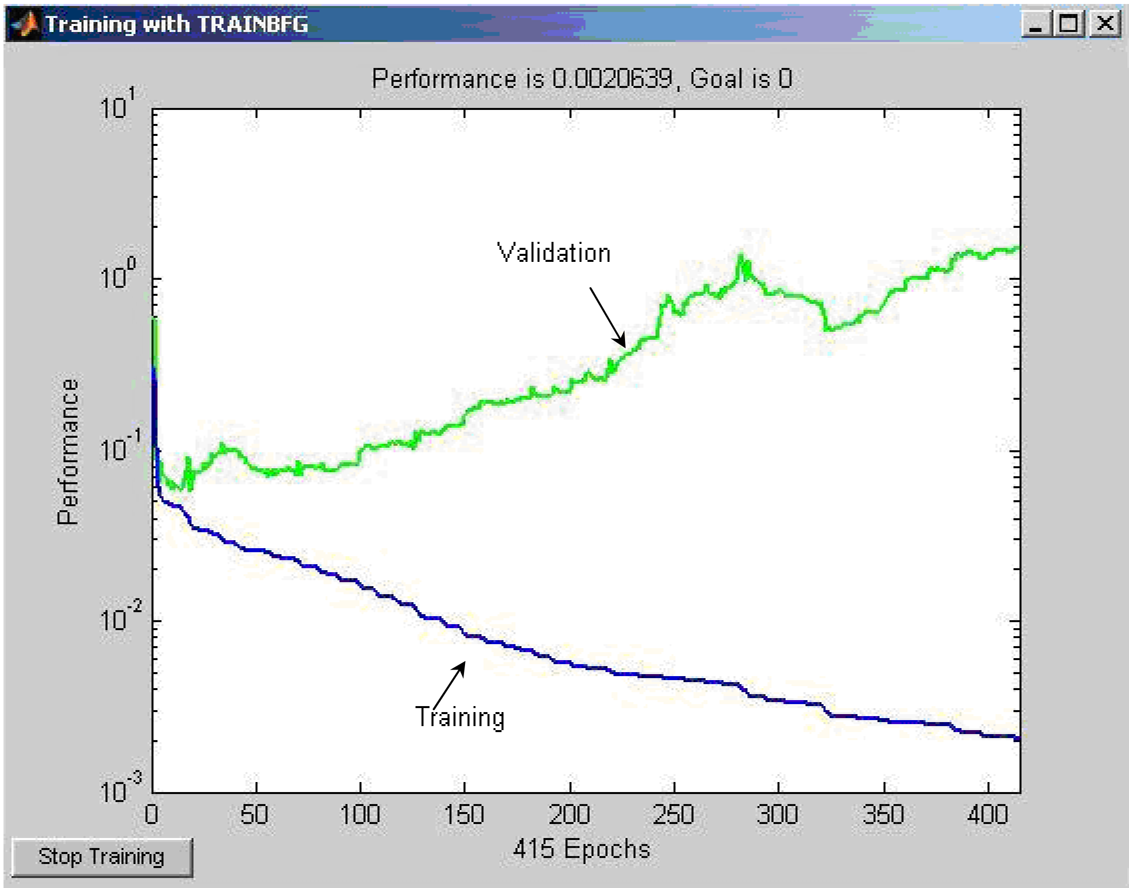

FIGURE 4 Training of MLFNN model.

Another problem with MLFNN models is the solution reaching local minima during the training process (*20*). That is, the model training can easily get trapped into some local minimum points such that a second run may end up with a very different solution; such occurrences are confirmed by tests on the Bayport and BCT datasets. In this study, a multiple-run strategy is used to address the local minima problem, which is to run the model training process twenty times and choose the trained network with the lowest validation error (not the training error).

4.2. ε-SVM Model

Since the ε-SVM model can guarantee a globally optimal solution, the training of the ε-SVM model is relatively easier compared to the MLFNN model. No multiple runs are required to overcome the local mimima problem. For the ε-SVM model used in this study, three parameters needed to be determined: *C*, ε, and a kernel function parameter (γ). Based on the value recommended in (*21*), an ε value of 0.05 is selected for all ε-SVM models. For each individual ε-SVM model, a grid search method is performed to find the best values for *C* and γ. Values between 1 and 100 with increment of 0.1 are tested for *C*, and values between 0.0001 and 0.01 are tested for γ with increment of 0.0005. In all, 991 *C* and 20 γ

values are evaluated for each ε-SVM model.

The ε-SVM models are fitted using the training datasets and evaluated on the validation datasets. For each ε-SVM model, *C* and γ values that result in the lowest validation error are selected. Figure 5 shows the parameter optimization result for the Bayport Import dataset under test scenario II.

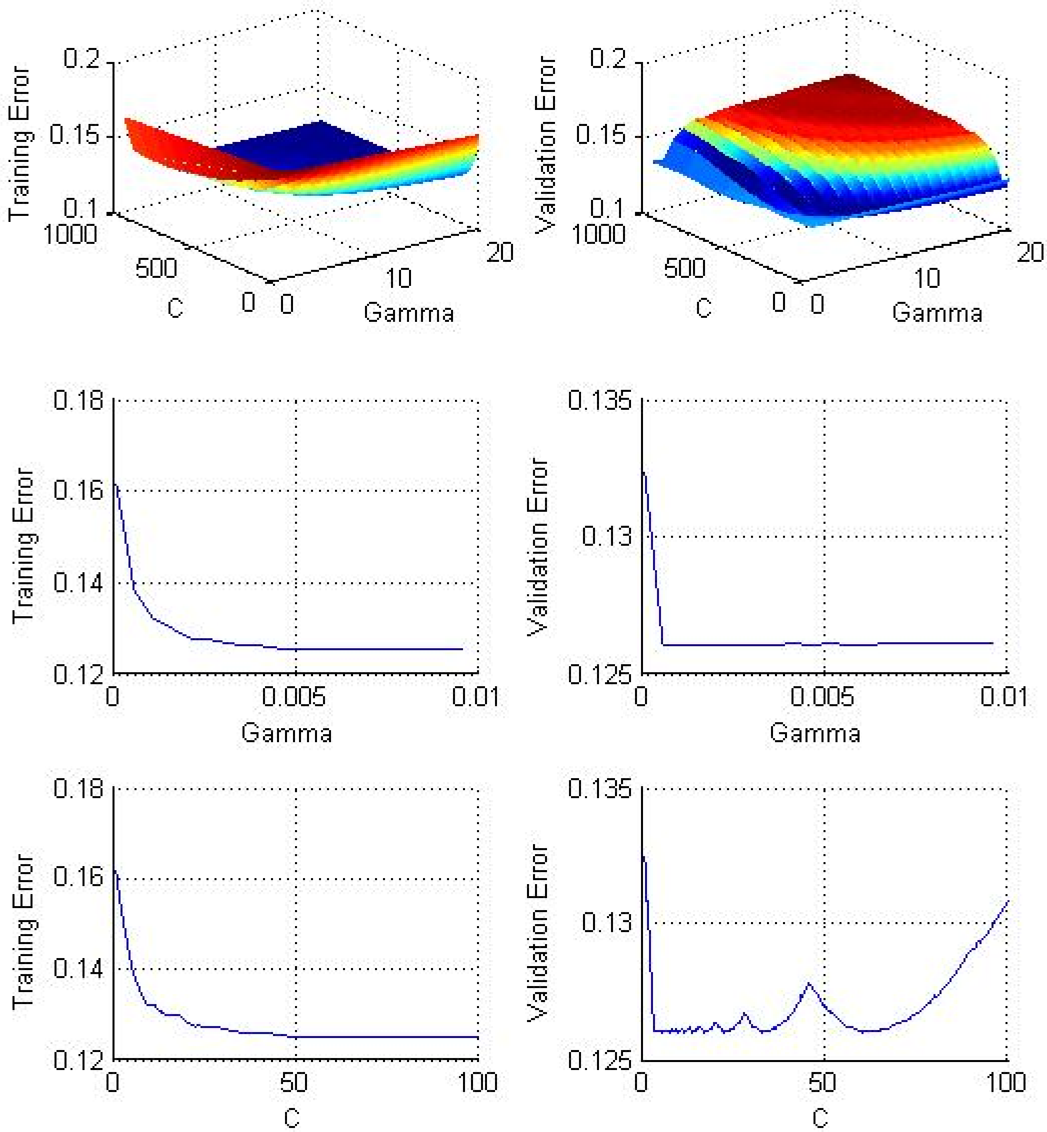

FIGURE 5 Training of ε-SVM model.

Note that the *x*-axis and *y*-axis values for the two 3-D plots in Figure 5 are not the actual *C* and γ values. They are the indexes for the *C* and γ values evaluated. The training and validation errors are also plotted against *C* and γ in Figure 5. It can be seen that for larger *C* and γ values, the training and validation errors initially become smaller. As the values of *C* and γ increase, the training and validation errors tend to stabilize and become less sensitive to the changes in *C* and γ. One exception is that when the *C* value is larger than a certain threshold, the validation error initially becomes unstable and then increases along with *C*. Based on the lowest validation error, the best *C* and γ are chosen to be 10 and 0.0036, respectively, for the Bayport Import dataset.

4.3. GP Model

Among the three models, the GP model requires the least effort to fit. There is no need to use a separate validation dataset for model selection. Fitting the GP model simply involves finding the best kernel parameters for the ARD kernel function using the training dataset. This is equivalent to maximizing a Type II maximum likelihood function (*22*) with respect to a vector of kernel parameters. Details of this training process can be found in (*13,22*) and will not be repeated here. Theoretically, this maximization process may also be trapped into local minimum points and generate different results from multiple training runs. However, tests on the Bayport and BCT data show that multiple training runs of the same GP model produce consistent results.

5. RESULT ANALYSIS

Two test scenarios are considered in this study for comparing the effectiveness between ε-SVM, GP, and MLFNN models for estimating truck trips. These three models are compared based on their Mean Absolute Percentage Error (MAPE) performance, which is defined in Eq. (10).

$$MAPE = \frac{1}{N}\sum_{k=1}^{N}\left|\frac{\hat{vol}(k) - vol(k)}{vol(k)}\right| \times 100\% \tag{10}$$

The MAPE results for the two test scenarios are shown in Tables 1 and 2, respectively. The data in Tables 1 and 2 suggest that for both test scenarios the GP and ε-SVM models perform better than the MLFNN model. The MAPEs from the GP and ε-SVM models are approximately the same. In several cases, these two methods have the same MAPEs. The overall performance of the GP and ε-SVM models is encouraging, suggesting that they are viable approaches for predicting truck volume at seaport terminals, intermodal facilities, and other transportation applications. The less than satisfactory performance of the MLFNN model does not mean that it should be excluded from further consideration. As discussed earlier, there are many different strategies to address the overfitting and local minima problems associated with the MLFNN model. It might be possible that by adopting the best strategy the MLFNN model can yield comparable results to that of the GP and ε-SVM models. However, it is beyond the scope of this work to compare all strategies and identify the best one. Also, even if the MLFNN model can perform as well as the GP and ε-SVM models, these two new models still have some advantages over the MLFNN due to their proven strong function approximation ability and ease of use.

TABLE 1 MAPE results for test scenario I

Dataset	**GP**	ε **-SVMs**	**MLFNN**
Bayport Export	13.4	13.4	16.5
Bayport Import	15.3	15.7	17.2
BCT Export	16.3	16.3	18.8
BCT Import	14.5	14.7	17.0

TABLE 2 MAPE results for test scenario II

Dataset	**GP**	ε **-SVMs**	**MLFNN**
Bayport Export	14.6	14.6	17.9
Bayport Import	14.8	14.1	16.3
BCT Export	12.8	11.9	15.9
BCT Import	14.9	14.9	20.9

A unique feature of the GP model which distinguishes it from the other two models is that it can estimate the standard deviations of the predicted truck volume data, which are shown in Figure 6. The top half of Figure 6 shows the observed and predicted truck volume data. The bottom half of it shows the corresponding estimated standard deviations. Such standard deviation information can be very useful for constructing confidence intervals for the predicted values. Since errors generally are inevitable in port truck traffic predictions, providing a confidence interval such as the one in Figure 7 would be much more informative and meaningful than simply presenting a single value. Figure 7 shows that most of the observed truck volume data fall into the predicted upper and lower boundaries. This result is very encouraging and confirms the usefulness of the estimated standard deviations.

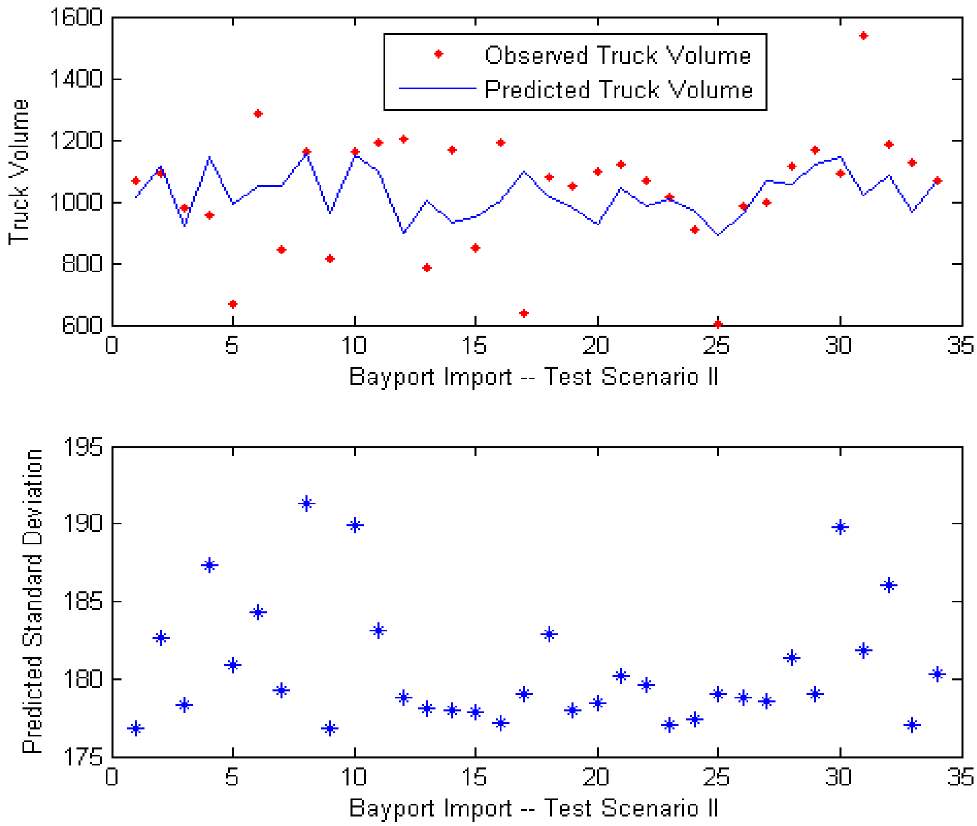

FIGURE 6 Prediction result for Bayport Import under test scenario II.

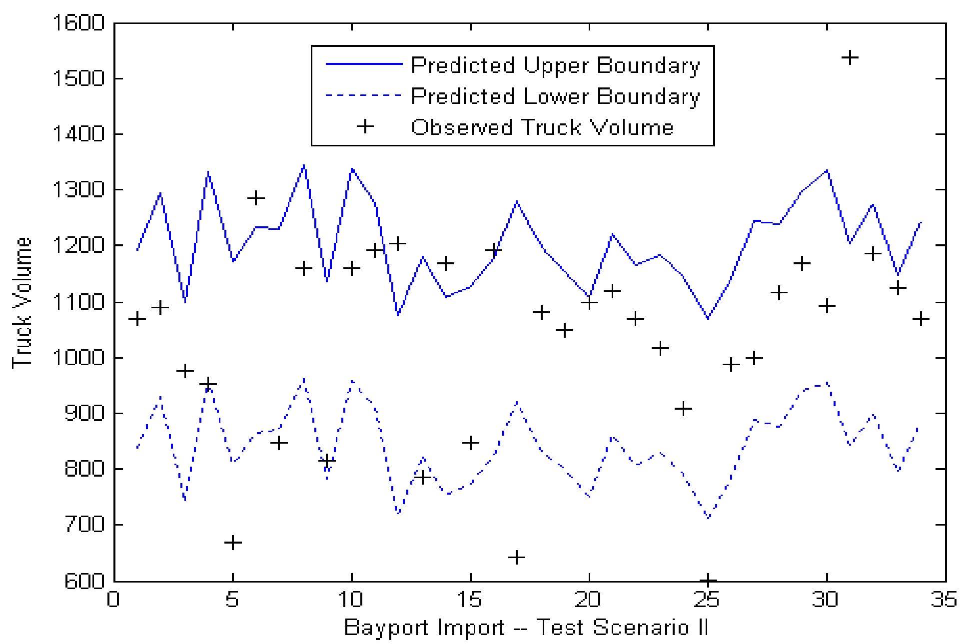

FIGURE 7 Predicted boundaries for Bayport Import under test scenario II.

6. CONCLUSIONS AND DISCUSSIONS

The heavy truck traffic generated by major seaports can pose significant negative impact on local and regional transportation network as well as the surrounding communities. A number of studies have developed models for predicting port truck traffic based on daily terminal operations data. Most of these are based on neural networks. In this study, two kernel-based machine learning methods are introduced for estimating truck trips at seaports. These two methods, Gaussian Processes (GP) and ε-Support Vector Machines (ε-SVM), are evaluated based on actual operations data from the Port of Houston's Bayport and Barbours Cut (BCT) terminals. The proposed two new models are also compared to the Multilayer Feed-forward Neural Networks (MLFNNN) model in terms of Mean Absolute Percentage Error (MAPE).

In this study, truck trips generated by import and export activities are investigated separately, generating four sets of data for model testing and comparison. Analysis of the summarized data suggests that most import and export containers stay no more than 12 days at the two terminals. Therefore, inbound truck trips are modeled based on the numbers of export containers in the next 12 days, while outbound truck trips are modeled based on the numbers of import containers during the previous 12 days. Each test dataset is divided into training, validation, and testing subsets. Two test scenarios with different training, validation, and testing subset sizes are considered for model evaluation and comparison. For both test scenarios, the GP and ε-SVM models yield similar prediction performance and in both test cases their prediction performance compares favorably to that of the MLFNN model.

All three models evaluated in this study have been proven to have superior function approximation ability. The key advantage of the proposed GP and ε-SVM models over the MLFNN model is that they require much less effort for model fitting. Fitting a MLFNN model needs to overcome the overfitting and local minima problems. What makes this task more challenging is that the solutions to these two problems are contradicting. On one hand, the model training process tries to minimize the training error and to stay away from any possible local minimum points. On the other hand, it is desirable to avoid overtraining the model and to stop the training process prematurely before the validation error goes up. Such contradiction is especially true for applications with limited training data sizes, as in this case the information contained in the samples is limited and it is easy for the MLFNN model to be overfitted. There is no established method in the literature to jointly solve the overfitting and local minima problems. In this study, an early-stopping method was adopted to stop the training process if the validation error keeps increasing for a certain number of iterations. The local minima problem was addressed by running the same training process multiple times, and the trained network with the lowest validation error was chosen for further comparison with the GP and ε-SVM models.

For fitting the ε-SVM model, there is no need to run the training process multiple times, as a single run can guarantee the globally optimal solution. This makes the training considerably easier and the dilemma associated with the MLFNN training can be avoided. In addition, a regularization term ($\frac{1}{2}w^T w$) is included in the formulation of the ε-SVM model (see Eq. 2). This regularization term provides the ε-SVM model with better generation ability such that the overfitting problem can be properly addressed. Training the ε-SVM model still requires a validation dataset, which is used to help find the best model parameters

C and γ. Without the issue of the local minima, finding the best ε-SVM model parameters can easily be done by a simple grid search method.

Although the GP model requires the least effort for model training, it produces result comparable to that of the ε-SVM model. Unlike the other two models, the GP model does not need a validation dataset for choosing the model parameters, which makes its training even simpler. The training of the GP model is based on the Bayesian framework and is simply to maximize a Type II likelihood function with respect to a vector of kernel function parameters. Although it has not been proven that this maximization process can guarantee a global optimal solution, tests on the Port of Houston data suggest that multiple training runs always produce the same result.

In summary, the overall strong performance of the GP and ε-SVM models indicate that they are viable approaches for modeling truck trips at seaports. It might be possible that the prediction performance of the MLFNN model can be improved by using other methods to better address the overfitting and local minima problems. However, given that the GP and ε-SVMs models produce satisfactory prediction performance and their relatively straightforward application, they offer researchers good alternative approaches to the MLFNN model.

7. FUTURE WORK

Although results in favor of the GP and ε-SVM models were reported in this study, tests on additional datasets are needed to further confirm their advantages over the MLFNN model. Also, in the ε-SVM model formulation (Eq. 2), differences between the predicted and observed values are not penalized if they are less than or equal to ε. For differences larger than ε, their corresponding penalties in the objective function (Eq. 2) are linearly proportional to the magnitudes of the differences. Additional research is needed to investigate the impact of nonlinear relationships between the differences and penalties on the prediction accuracy.

8. ACKNOWLEDGMENTS

The authors would like to thank the James E. Clyburn University Transportation Center at South Carolina State University for supporting this research. This work benefited greatly from the assistance of Rob Sawyer who provided the Port of Houston terminal data. All contents, opinions, and results expressed in this paper are solely of the authors.

REFERENCE

1 Klodzinski, J., and H.M. Al-Deek. Methodology for Modeling a Road Network with High Truck Volumes Generated by Vessel Freight Activity from an Intermodal Facility. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1873*, TRB, National Research Council, Washington, D.C., 2004, pp. 35–44.

2 Al-Deek, H.M., G. Johnson, A. Mohamed, and A. El-Maghraby. Truck Trip Generation Models for Seaports with Container and Trailer Operation. In *Transportation Research*

Record: Journal of the Transportation Research Board, No. 1719, TRB, National Research Council, Washington, D.C., 2000, pp. 1–9.

3 Guan, C.Q., and R. Liu. Container Volume and Truck Trip Generations at Marine Container Terminals – A behavioral Analysis. CD-ROM. Transportation Research Board of the National Academies, Washington, D.C., 2008.

4 Al-Deek, H.M. Which Method is better for Developing Freight Planning Models at Seaports – Neural Networks or Multiple Regression? In *Transportation Research Record: Journal of the Transportation Research Board, No. 1763*, TRB, National Research Council, Washington, D.C., 2001, pp. 90–97.

5 Klodzinski, J., and H.M. Al-Deek. Transferability of an Intermodal Freight Transportation Forecasting Model to Major Florida Seaports. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1820*, TRB, National Research Council, Washington, D.C., 2003, pp. 36–45.

6 Klodzinski, J., A. Al-Daraiseh, M. Georgiopoulos, and H.M. Al-Deek. Development of a Java Applet for Generating Truck Trips from Freight Data. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1870*, TRB, National Research Council, Washington, D.C., 2004, pp. 10–17.

7 Sideris, A.C., M.P. Boile, and L.N. Spasovic. Using on-line information to estimate container movements for day-to-day marine terminal operations. Available online at http://transportation.njit.edu/NCTIP/publications/forecastTRB.pdf. Accessed on July 29, 2009.

8 Sarvareddy, P., H.M. Al-Deek, J. Klodzinski, and G. Anagnostopoulos. Evaluation of Two Modeling Methods for Generating Heavy-Truck Trips at an Intermodal Facility by Using Vessel Freight Data. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1906*, TRB, National Research Council, Washington, D.C., 2004, pp. 113–120.

9 Al-Deek, H.M. Use of Vessel Freight Data to Forecast Heavy Truck Movements at Seaports. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1804*, TRB, National Research Council, Washington, D.C., 2002, pp. 217–224.

10 Holguin-Veras, J., Y. Lopez-Genao, and A. Salam. Truck-Trip Generation at Container Terminals – Results from a Nationwide Survey. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1790*, TRB, National Research Council, Washington, D.C., 2002, pp. 89–96.

11 Xie, Y, and Y. Zhang. A Wavelet Network Model for Short-Term Traffic Volume Forecasting. *Journal of Intelligent Transportation Systems*, Vol. 10, No. 3, 2006, pp. 141-150.

12 Hornik, K., M. Stinchcombe, and H. White. Multilayer Feedforward Networks are Universal Approximators. *Neural Networks*, Vol. 2, No. 5, 1989, pp. 359-366.

13 Rasmussen, C.E., and C.K.I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Cambridge, MA, 2006.

14 Zhang. Y., and Y. Xie. Forecasting of Short-Term Freeway Volume with v-Support Vector Machines, In *Transportation Research Record: Journal of the Transportation Research Board, No. 2024*, TRB, National Research Council, Washington, D.C., 2008, pp. 92-99.

15 Wu, C.H., J.M. Ho, and D.T. Lee. Travel-Time Prediction with Support Vector Regression. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 5, No. 4, 2004, pp. 276-281.
16 Suykens, J.A.K., T.V. Gestel, J.D. Brabanter, B.D. Moor, and J. Vanderwalle. *Least Squares Support Vector Machines*. Singapore, World Scientific Publishing Co. Pte. Ltd., 2002.
17 Rasmussen, C.E., and C.K.I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Cambridge, MA, 2006.
18 Zhao, K., S.C. Popescu, and X. Zhang. Bayesian learning with Gaussian processes for supervised classification of hyperspectral data. *Photogrammetric Engineering & Remote Sensing*, Vol. 74, No. 10, 2008, pp. 1223-1234.
19 Smola, A.J., and B. Scholkopf. A tutorial on support vector regression. *Statistics and Computing*, Vol. 14, No. 3, 2004, pp. 199-222.
20 Zhang, Y., and Y. Xie. Application of Genetic Neural Networks to Real-Time Intersection Accident Detection Using Acoustic Signals. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1968*, TRB, National Research Council, Washington, D.C., 2006, pp. 75-82.
21 Chang, C.C., and C.J. Lin. LIBSVM: a library for support vector machines. 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.
22 Xie, Y., K, Zhao, Y. Sun, and D. Chen. Gaussian Process for Traffic Flow Modeling and Forecasting. Working Paper.