ANALYSIS OF SUPPORT VECTOR MACHINE REGRESSION FOR BUILDING ENERGY USE PREDICTION

A Thesis

by

SHINWOO LEE

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Juan-Carlos Baltazar
Committee Members,	Jeff S. Haberl
	David E. Claridge
Head of Department,	Robert R Warden

August 2020

Major Subject: Architecture

ABSTRACT

There are many inverse modeling methods to model the whole building energy use. Multiple linear regression (MLR) and change-point liner regression (CPLR) have been some of the most common methods due to their direct interpretation concerning building energy modeling and their fair accuracy.

Recently, as machine-learning techniques have become more accessible, there have been many attempts to apply these techniques to building energy modeling. However, no studies have conducted an in-depth comparison with the conventional inverse model methods using large buildings sample size.

This study conducted a comprehensive comparative study based on Support Vector Machine (SVM), one of the most widely used machine-learning methods for flexibility and accuracy, with enough cases to draw a reasonable conclusion between models generated from conventional methods such as MLR and CPLR, and those from SVM. This work, besides the comparative analysis, included a thorough SVM performance analysis for building energy modeling. It described in detail its implementation, and showed its performance as a regression technique for building energy modeling under the influence of different variables.

The comparative study focused on modeling whole building chilled water use (CHW) and heating hot water use (HHW), and analyzed the influence of such variables as the outdoor dry-bulb temperature (OAT), the outdoor dew-point temperature (DPT), the

outdoor air enthalpy (OAE), and the operational effective enthalpy (OEE). The numerical experiments were based on a sample of 41 whole year daily and hourly building energy use datasets that were converted from hourly data.

According to the comparative analysis between SVM and MLR, based on CHW data, SVM consistently showed higher performances by an average of 6.8% on daily and 2.0% on monthly models, respectively. For the SVM and CPLR performance analysis, four pairs of dependent and independent variables were considered: CHW-OAT, CHW-OAE, CHW-OEE, and HHW-OAT. On daily modeling, SVM demonstrated consistently higher performance, although most of the cases resulted in a marginal advantage by less than 1% for all variables utilized. Despite such marginal gains in mean performance, SVM showed advantages by up to 3% for some datasets. On the monthly model, however, SVM did not exhibit better results for any dependent-independent variable pair.

ACKNOWLEDGEMENTS

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supervised by a thesis committee consisting of Professors Juan-Carlos Baltazar (Advisor) and Jeff Haberl of the Department of Architecture and Professor David Claridge of the Department of Mechanical Engineering.

All of the work conducted for the thesis was completed by the student independently.

Funding Sources

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER I INTRODUCTION

Energy consumption in buildings takes up approximately 40% of the world's energy use. [1] Thus, many governments promoted energy-efficient retrofits for all types of buildings: residential, commercial, institutional and industrial one. Unfortunately, most building owners do not support energy efficient retrofits just for the good of mankind when it does not make sense financially. Therefore, no matter how many social-environmental benefits, most buildings owners will decline the implementation of retrofits unless they can make sure that the cost savings outweigh the implementation cost. In other words, unless the amount of potential energy savings can be quantified, the financial benefits also will not be able to be calculated, ending up with no motivation to carry out energy-saving retrofits. Once buildings retrofits have been performed, it is not possible to measure how much energy would have been consumed had it not been for retrofit implementation using before-after whole building data. The only available records after retrofit are post-retrofit data. In such cases, a baseline model that simulates pre-retrofit energy consumption can be used. Regarding this issue, ASHRAE Guideline 14 [2] provides a thorough guidance about how to measure energy savings in a reliable manner, covering different approaches and instrumentation. There are two ways to create building energy models: forward and inverse models. [3] Forward modeling uses a comprehensive model of the whole-building energy consumption process that use engineering principles of heat transfer and thermodynamics. Although it is expected to be highly accurate, forward modeling requires a huge number of parameters that are usually not always easy to identify. On the other hand, when a certain period of energy

consumption data is available, inverse modeling is more convenient since it does not require as many inputs as forward modeling, and it can provide a highly accurate prediction over a longer period of time. This study focuses on the latter category, the inverse modeling.

Inverse modeling is a broad term that encompasses linear regression, time-series analysis, and a variety of machine-learning methods. Among these, multiple linear regression and change-point regression have become the best practices in building energy measurement and verification, thus they are suggested in ASHRAE Guideline-14 [2]. Change-point regression is a type of piecewise linear regression that consists of two or three pieces of line connected at the points called changed points. Change-point regression can provide the physical interpretation about how a building consumes energy. Multiple linear regression is mathematically an extension of simple linear regression. Unlike a simple linear regression, the multiple linear regression uses more than one independent variables. The coefficients of the multiple linear regression provide the insight as to which independent variable is relatively more influential than other independent variables.

Recently, as more complicated techniques or machine-learning algorithms have become increasingly user-friendly, many researchers have attempted to apply the new techniques to their area of research interest. In the building energy inverse model research, Artificial Neural Networks (ANNs) and Support Vector Machine (SVM) are the two most popular machine-learning methods [4],[5]. Nonetheless, few studies of this kind conducted an in-

depth comparison of the conventional inverse model methods with a large number of samples.

To fill this gap, this study selected one of the two machine-learning methods mentioned above and conducted an in-depth comparison between the conventional methods and the new method. Between the two machine-learning candidates, SVM was selected for two reasons. First, the predictive ability of SVM outperformed the Artificial Neural Networks (ANNs) model, according to the studies of Li et al. [6], Zhao and Magoulès [7], and Massana et al. [8]. Second, it is easier to implement than ANNs as they have fewer parameters to tune. After SVM was chosen as a representative machine-learning method, a comparative study was conducted with Change-point regression and multiple linear regression with enough samples to draw a reasonable conclusion. Although the idea of a comparative study initiated this research, this study also covers the contents to better understand the SVM itself from fundamentals, implementation, observation of resultant SVM regression models, and the influence of different independent variables on the performance of SVM.

1.1 Objectives

SVM is a machine learning algorithm that can capture diverse patterns with high accuracy due to its flexibility. The main objective of this study is to investigate the performance of SVM as a whole-building energy inverse modeler in comparison to other

conventional methods in terms of accuracy and feasibility. The study also includes other types of experiments to better understand the intrinsic part of SVM.

Specifically, the following tasks were addressed to achieve the objective:

Task 1. Investigate the characteristics of SVM with varying conditions, such as different kernel functions and different patterns of input data.

Task 2. Determine the appropriate SVM implementation methodology.

Task 3 Implement a SVM with different independent variables and observe the influence of the use of different independent variables.

Task 4 Perform a comparison of the SVM regression model against multiple linear regression and change-point linear regression. Analyze the case where SVM can be a good alternative to conventional statistical methods.

1.2 Study Limitations

This research has the following limitations.

1) The study is based on commercial buildings, specifically buildings on the Texas A&M University campus. These buildings are located in a hot-humid climate.

2) The study was performed on building chilled water and heating hot water energy use data sets.

3) The detailed information needed to explain the actual consumption patterns of buildings was not considered, such as HVAC operation strategies, detailed operational schedules, and events that could change the energy consumption patterns of buildings.

Additionally, given the broad spectrum of options of the SVM hyper-parameter tuning methods, only the Exhaustive Grid Search (EGS) and another approach on One Dimensional Grid Search with recommended initial values (ODGS) were considered.

CHAPTER II BACKGROUND

To analyze how buildings consume energy, quantitative building energy simulation models are used to calculate the hourly energy use over the year. The accuracy of such a model is directly related to the reliability of the building energy models used. In general, there are two different approaches to create models: One uses building characteristics information and engineering principles to simulate the building annual energy use. The other uses the measured building energy consumption data to establish an empirical model. The former is called forward modeling and the latter is called inverse modeling.

2.1 Forward Modeling

This approach can be used to predict peak loads and annual energy consumption of buildings by establishing an energy model. It does not necessarily require an actual building, but can be used with building design information. To ensure accuracy, a forward model requires as many buildings characteristics as possible such as the *U*-values of each material, internal loads, and operating schedule. It also requires multiple models that describe the performance of the Heating Ventilating and Air Conditioning (HVAC) equipment such as air handling units, fans and pumps, and main plant chillers and boilers. Such parameters and equipment models make possible the generation of a complete whole-building energy model. Major simulation software used for this approach includes DOE-2, EnergyPlus, ESP-r, and TRNSYS [3]. The details of simulation methodologies vary software by software. However, all of them can be

explained with similar categories: space loads, system components, and plant energy use. Space loads indicate cooling, heating, or ventilation loads that need to be met to provide the zonal set point temperatures, humidity, and air quality. Such systems serve to satisfy such space loads either mechanically or passively. In general, such a system connects the load to the heat sources or cooling sources. This is carried out by means of ducts, dampers, heating coils, cooling coils, and control devices. Through the heat transfer between heating/cooling coils, return air, and outdoor air, air is conditioned as needed to be supplied to the space in demand. How systems are arranged affects the thermal comfort and energy consumption of a building. HVAC plants serve as a heat source or cooling source for an individual building or multiple buildings. The typical major equipment in a plant are boilers and chillers. Although various sources of energy can be used as an input to these devices, natural gas is often used for boilers and electricity is used for chillers. As previously mentioned, this model can be applied to buildings that exist, or to new buildings during the design phase. [3]

2.2 Inverse Modeling

When input and output variables are known, a functional or mathematical description of a system can be estimated with statistical approaches, this is called inverse modeling. In an inverse building energy model, the energy consumption patterns of buildings can be explained if coincidental weather data is available alongside the actual energy consumption data. Inverse modeling methods have different traits than forward

modeling. An inverse model can make relatively accurate prediction since it is based on data obtained from the actual building operation. In addition, inverse models are simple to establish since the method does not require buildings physical or descriptive parameters such as those required by the forward modeling. It is also possible to establish an inverse model using the output of building energy simulation. The validity of the inverse model can be assessed using statistical criteria such as the coefficient of determination (r^2), the standard error or the root mean square error (RMSE), and related indexes with statistical *t*-test. [9]. A few examples of famous inverse modeling methods are summarized in the following subsections.

2.2.1 Linear Regression

Linear regression is the simplest form of the regression methodologies and characterized by its simplicity and ease of automation. It relates a dependent variable to a linear combination of independent variables multiplied by each coefficient. When it comes to the application as building energy use forecast, single-variate piecewise linear regression (i.e. change-point linear regression) and multiple linear regression models are perhaps the most straightforward options.

2.2.1.1 Single-variate Piecewise Linear Regression (Single-variate Change-point Linear Regression)

Compared to simple linear regression, change-point regression can provide more accurate models. More importantly, with the addition of one or two parameters the physical interpretability noticeably increases. For building energy use prediction, as an independent variable, outdoor temperature is the most important one (see Figure 1) [3]. How a building consumes energy significantly changes when the outdoor temperature or other independent variables passes a change point. The numerical value of slopes implies information of the heating or cooling efficiency of air-conditioning equipment and the building thermal capacity, ventilation rate, infiltration rate of buildings. The best explained equations of change-point linear regression are expressed as below:

$$E = b_0 + b_1(T - b_2)^+ \tag{EQ. 1}$$

$$E = b_0 + b_1(b_2 - T)^+ \tag{EQ. 2}$$

$$E = b_0 + b_1(b_2 - T)^+ - b_3(T - b_2)^+ \tag{EQ. 3}$$

$$E = b_0 - b_1(b_2 - T)^+ + b_3(T - b_2)^+ \tag{EQ. 4}$$

$$E = b_0 + b_1(b_2 - T)^+ + b_3(T - b_4)^+ \tag{EQ. 5}$$

Where E is energy, T is outside temperature, b_0, b_1, b_2, b_3, and b_4 are parameters to specify the form of a regression model. Three-parameter and five-parameter models are based on Princeton scorekeeping methods (PRISM), which Fels [10] derived from the concept of variable-base degree day. Three parameters (3-P) models, which is expressed by Eq.1 and Eq.2, generally fit well for buildings having a clear distinction between

weather-dependent and weather independent outdoor temperature ranges. (See Figure 1 (a)) Five parameters (5-P) models fit well for buildings which use electricity for both heating and cooling (See Figure 1 (c)). Ruch and Claridge [11] developed a four-parameter (4-P) model for commercial buildings (See Figure 1 (b)). The four parameter models are described by Eq.3 and Eq.4. These models usually fit well for buildings that have complex HVAC features. Commercial buildings generally have a high internal load, thus the weather factor alone is usually not sufficient to explain the building energy consumption [12]. The slope of energy use begins to change in some cases when the total airflow reaches a minimum airflow set- point or when there is a change in the hot deck temperature (i.e. heating coil). In addition, different load characteristics between internal and external zones can make a change point more distinct.

Although the change-point regression models were introduced around 1980s and early 1990s, they are still very popular and a subject of research. For example, Kim and Haberl [13] developed a methodology for building energy simulation calibration using a sensitivity analysis of data from a case study house on a case model with a change-point regression model. In the study, they showed the parameters from the change-point model helped them to develop an easy-to-use home energy audit procedure. Park et al. [14] quantified the thermal performance of buildings and the potential savings from monthly bills, using change-point regression models, and partial building physical characteristics
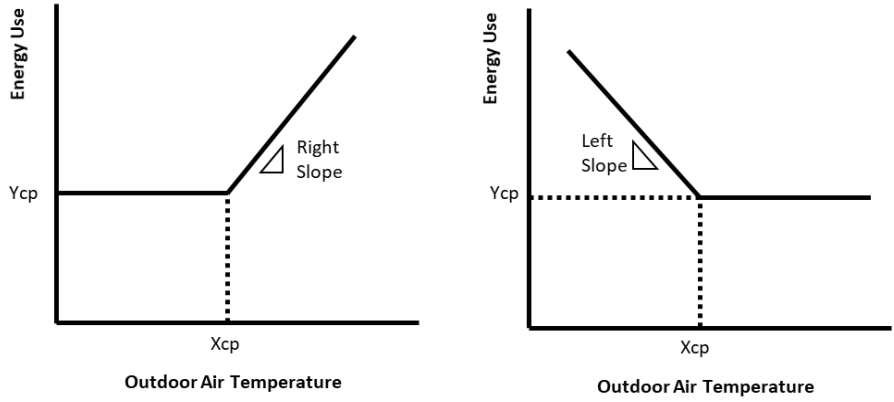
(a) 3P Change Point Liner Regression Model (left: cooling, right: heating)
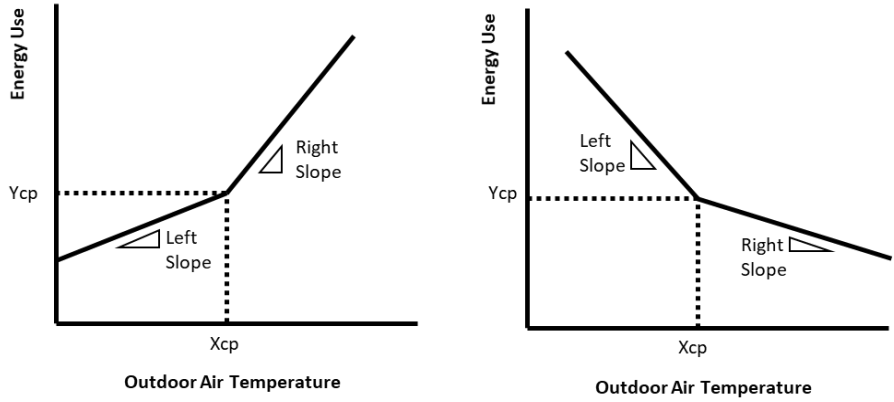
(b) 4P Change Point Liner Regression Model (left: cooling, right heating)
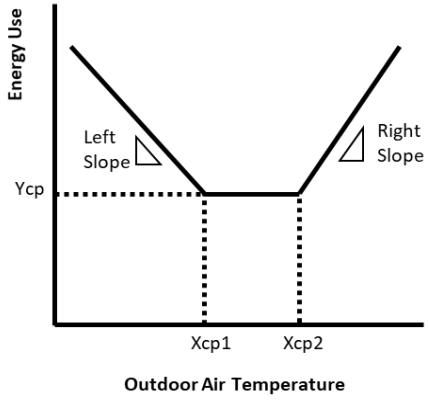
(c) 5P Change Point Liner Regression Model

Figure 1 - Representations of Change Point Linear Regression Models. (Adapted from ASHRAE Fundamentals, 2017 [3])

2.2.1.2 Multiple Linear Regression

Multiple linear regression can be seen an extension of simple linear regression in terms of its mathematical expression. Unlike simple linear regression, multiple linear regression uses more than one independent variable to describe a dependent variable. The concept of a change-point can also be applied to multiple linear regression using indicator variables, the values of which varies between 1 or 0 depending on whether the outdoor temperature is above or below the change-point temperature [15]. Energy consumption in buildings is not only dependent on outdoor air temperature, but also on multiple variables such as humidity, solar insolation, occupancy levels, and HVAC equipment operating schedules. Thus, using more than one variable can generate more accurate models if inputs are chosen properly. Katipamula et al. [16] showed that multiple linear regression models were generally superior to simple regression models (2P) in terms of accuracy. Haberl and Claridge [17] used multiple linear regression to predict the energy consumption of the Recreation Center of Texas A&M university campus. A variety of independent variables were used including ambient temperature, solar radiation, day of the week and scheduled operating hours. This model was fed into an expert system that was designed to makes suggestions concerning the possible causes of abnormal operation. However, there are some drawbacks. Since it requires more meters, the possibility of data being unavailable due to meter errors is higher. They showed it is still not strong in describing non-linear patterns. Finally, there is a chance that one input variable is dependent on another input variable, which is called multicolinearity. This can disturb the model, making model unreliable [9].The impact of

multicolinearity could be significant if the correlation between independent variables is stronger than the correlation between a dependent variable and any of independent variable [16, 18]. According to ASHRAE Fundamentals [3], the general form of equation for a steady-state, multi-variate model is shown below:

$$Q_{bldg} = a + bT_o + cI + dIT_o + eT_{dp}^+ + fq_{sol} + gE_{int} \qquad \textbf{(EQ. 6)}$$

where *I* is an indicator variable having a value of either 1 or 0 to make different slopes based on a change-point outdoor temperature. Accordingly, the terms having the coefficient *c* and *d* will be zero when a change point does not exist. T_{dp}^+ indicates the dew point of outdoor air and becomes zero when the dew point of the outdoor air is lower than the average supply temperature of the cooling coil. This variable exists mostly to account for the influence of latent loads. Specifically, most latent loads of commercial buildings are generated from outdoor ventilation air rather than infiltration or occupancy. When outdoor air is drawn into an air handling unit, and if the dew point of the outdoor air is higher than the surface temperature of cooling coils, condensation occurs. Since the driving force of this condensation is the cooling effect of cooling coils, this causes an additional load to the cooling coil. This load is also called a latent load. This latent load does not appear if the dew point of the outdoor air is lower than the surface temperature of cooling coil because there is no condensation. q_{sol} indicates solar radiation effect and E_{int} indicates internal heat gain from electric equipment and lighting.

2.2.2 ARIMA (Autoregressive Integrated Moving Average)

ARIMA is one of the most general methods to predict time series data. ARIMA model is specified by three elements: The number of lag terms for autoregressive (*p*), the order of difference (*d*), the number of lag terms for moving average (*q*). The AR (Autoregressive) part of the equation displays how the variable of interest is regressed on its own past values. AR(*p*) with p lags is expressed as below [19]:

$$x_t - \mu = \sum_{i=1}^{p} \gamma_i x_{t-i} + \omega_t \qquad \text{(EQ. 7)}$$

Where x_t is the output of a time series function in time *t*, μ is the mean of x_t, γ_i is the coefficient for the lagged variable in time *t-i*, and ω_t is assumed to be a white noise with mean zero.

The MA (Moving Average) part shows a linear combination of residuals from past periods. MA(*q*) with *q* lags is expressed as below:

$$x_t = \omega_t + \sum_{i=1}^{q} \theta_i \omega_{t-i} \qquad \text{(EQ. 8)}$$

Where, x_t, is the output of a time series function in time *t*, ω_t, is a white noise with mean zero, θ_i, is the coefficient for the lagged error term in time *t-i*.

The integrated part, *I*, is needed to remove trends that lead the data far from its mean value over time. As a result of this term, the trend changes to a stationary value. For example, when a variable y_t is not stationary and has a linearly increasing trend, inserting the following first order of differences can effectively eliminate the trend effect:

$$\Delta x_t = x_t - x_{t-1}$$
(EQ. 9)

Abdel-Aal and Al-Garni [20] used ARIMA with 5 years of data to forecast monthly

domestic electric energy consumption for the next year in the Eastern Province of Saudi

Arabia. They used Abductive Network Machine Learning (AIM) and multiple linear

regression for the comparison of performance. The AIM can be seen as a generalized

version of the artificial neural network. However, the structure of AIM is not limited to

neuron analogies. AIM can take diverse forms of structure that connects inputs to

outputs depending on the context. This characteristic can be useful in simplifying a

complex task. Compared to multi-variate regression and Abductive Network Machine-

learning(AIM) models previously developed on the same data, the ARIMA models

require less data, fewer coefficients, yet shows more accurate results. The optimum

ARIMA model created in their study revealed a mean percentage error or 3.8%, while

the MLR and AIM showed 8.1% and 5.6%, respectively.

Kandananond [21] used ARIMA, ANN, and MLR to predict electricity demand in

Thailand and compared their performance. The ANN outperformed the other two

models, showing a Mean Absolute Percentage Error (MAPE) of 0.996% while the

ARIMA model showed 2.810% and the MLR model showed 3.260%. Despite the result,

the research concluded that the ARIMA or MLR might be preferable to ANN in terms of

the balance between performance and simplicity. Pasapitch et al. [22] used the

Autoregressive Moving Average (ARMA) and ARIMA to forecast electricity

consumption and compared the results. As previously stated, the role of *I* in the ARIMA

model is to remove the impact of trends in the data. Hence, the effectiveness of ARIMA

over ARMA depends on whether the data has non-seasonal difference. The results showed that the ARIMA model was more suitable for monthly and quarterly forecasting periods whereas the ARMA model showed improved results for daily and weekly forecasting periods.

2.2.3 Fourier Series

Fourier series modeling works well for capturing temporal patterns of building energy consumption. Thus, it is useful for buildings that have a distinct schedule or periodic patterns such as commercial buildings. In addition, although it is more complex than simple linear or multilinear regression models, it can effectively model a nonlinear relationship that possibly exists between outside air temperature and energy consumption of buildings. The generalized model equation proposed for weather dependent energy use takes the following form [23]:

$$E_{h,d} = \sum_{k=1}(X_k + Y_k + Z_k) \tag{EQ. 10}$$

$$X_k = k \sum_{i=0}^{i_{max}} [\gamma_{k,i}\left(asin\ \frac{2\pi}{P_i}d\right) + \delta_{k,i}\left(acos\ \frac{2\pi}{P_i}d\right)] \tag{EQ. 11}$$

$$Y_k = k \sum_{j=0}^{j_{max}} [\alpha_{k,j}\left(asin\ \frac{2\pi}{P_i}h\right) + \beta_{k,j}\left(acos\ \frac{2\pi}{P_i}h\right)] \tag{EQ. 12}$$

$$Z_k = k \sum_{i=0}^{imax} \sum_{j=0}^{jmax} [\emptyset_{k,i}\left(asin\ \frac{2\pi}{P_i}d\right) + \varphi_{k,i}\left(acos\ \frac{2\pi}{P_i}d\right)] \times [\eta_{k,j}\left(asin\ \frac{2\pi}{P_j}h\right) +$$

$$\varsigma_{k,j}\left(acos\ \frac{2\pi}{P_j}h\right)] \tag{EQ. 13}$$

Where *k* is an index denoting the series corresponding to the internal loads or weather variables, *d* is the day-of-year representing the annual cycle, *h* is the hour-of-day representing the daily cycle, *X* is the Fourier Series representing seasonal cycle, *Y* is the Fourier Series representing diurnal cycle, *Z* is the Fourier Series accounting for interaction effects. Seem and Braun [24] also used a Fourier series model to forecast the electric demand in a building on an hourly basis. Dhar et al. [25] separated days of the year when buildings operate differently and showed improved statistical fits.

2.2.4 Machine Learning

Numerous attempts have been made to apply machine learning techniques to building energy modeling. Among them, the Artificial Neural Network (ANN) has been the most popular technique [26], followed by Support Vector Machine (SVM).

2.2.4.1 Artificial Neural Networks – ANNs

The ANN learns the relationship between the input variable and the output variable in a similar way as human brains learn and it does not necessarily require as many parameters as detailed simulations. Although ANNs can look similar to multi-variate regression models, it does not necessitate any form of the presumed models. Instead, it learns the key information patterns within a multidimensional information domain [27]. To be specific, ANN structures are made of layers that contains neurons. Neurons in different layers are connected to each other. Although neurons can receive multiple

signals from inputs, each neuron discharges only one signal, and the intensity of the

signal is adjusted by a weighting factor assigned to the connection [28]. Since the ANN

can make models in a flexible way, it is good for handling non-linear energy patterns.

There are various types of ANN models, including: Back-propagation neural network

(BPNN); Radial Basis Function Neural Network (RBFNN); and General Regression

Neural Network (GRNN) models. The BPNN is the most widely used. It is arranged to

minimize the mean square error between the predicted output and a target value(s).

RBFNN does not use weighting factor for the connection between the inputs and hidden

layer. In the RBFNN Gaussian function is usually used as the transfer function. Finally,

the GRNN uses a dynamic structure. Its merits lie in its quick learning and its ability to

convergence to the optimum model [29].
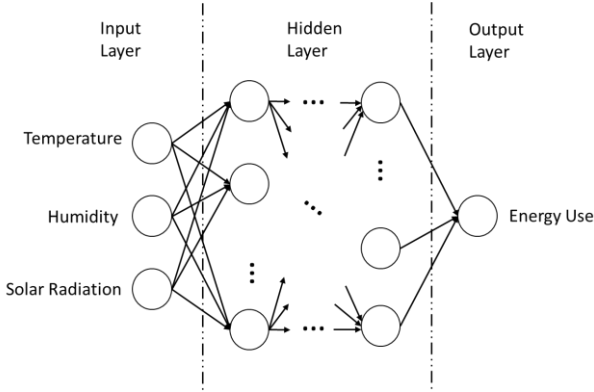
Figure 2 - Artificial Neural Network Structure

2.2.4.2 Support Vector Machines - SVM

Support vector machine (SVM) offers one of the most robust and accurate methods

among all well-known machine learning algorithms. The SVM was identified as one of

the top-ten algorithms in data mining by IEEE International Conference on Data Mining [30]. Support vector machine (SVM) is known to be higher accuracy than ANN. Unlike ANN, the solution of the SVM is unique. The solution depends only on a set of training data points, namely support vectors. What should be noted is that the unique solution in a SVM depends on the selection of hyperparameters and kernel function. The main drawback of a SVM is calculation time. It increases in proportional to approximately the square or cube of the samples size [31]. To overcome this drawback, other types of SVMs have appeared. For example, Suykens and Vandewalle [32] proposed a least squares SVM (LS-SVM). Their LS-SVM is a reformulation of the standard SVM. Whereas the standard SVM requires a quadratic programming calculation, the LS-SVM operates only with a linear equation as a result of a modified optimization function. Thus, the LS-SVM has an enhanced calculation efficiency. Yet, this advantage comes with a loss of the sparseness of support vectors. This results in an increase in model complexity, difficulty in generalizing data patterns, and in an increase in the demand of computer memory space.

2.2.5 The Great Energy Predictor Shootout

In an attempt to identify the most accurate inverse modeling method for predicting building energy consumption given hourly measurement data, ASHRAE TC 1.5 and TC 4.7 opened a competition in the summer of 1993 [33]. The contestants were given two sets of data. The A dataset included hourly whole-building chilled water, heating hot

water and electricity use along with weather data. The B dataset consisted of the

measurement from four different pyranometers at different orientations and one

measurement from a pyrheliometer. The contestant was tasked with creating a model

using the training dataset without any further information about the data file. In each of

the dataset, only the training data were available to the contestant. A test dataset from the

same data file was not available to the contestant that was used to test the models. The

performance was assessed by comparing the actual and predicted values during the

testing period. The result showed that neural networks were the most widely used

methods among the top entrees. However, it is also used by many lower ranking

contestants as well. That implies the performance of the neural network depended not

only on the capacity of the ANN itself, but also on how strategically the modelers

organized the given data before using the ANN. For instance, the first-place winner of

the competition, Mackay, used the Automatic Relevance Determination (ARD) model

before applying the ANN. The ARD is a Bayesian model to distinguish the relevant

independent variables from the irrelevant one. [34] By using ARD, Mackay could

successfully prevent overfitting and won the competition. Similarly, the second-place

winner Ohlsson et al used a technique called delta test to determine relevant independent

variables [35]. The third-place winner, Feuston and Thurtell,used the Principal

Component Analysis (PCA) before using the ANN [33]. Through the PCA, they

transformed the independent variables to smaller numbers of orthogonal independent

variable sets that cover most of the information in the original input data. By doing so,

they could generate the model, the implementation time of which was much faster than

the other winners. Another noteworthy result was that none of the models could successfully predict the energy consumption when unexpected events or a different schedule appeared in the test dataset that could not be predicted by the training dataset. In 1994, ASHRAE TC 1.5 and TC 4.7 held a second competition [36]. In the second shootout, the contestants were asked to train models based on pre-retrofit period building energy use data. Similar to the previous ASHRAE competition, the trained models were assessed by the difference between actual and predicted values. Unlike the first competition, the different savings during the post-retrofit periods that were calculated by different trained models were compared against the measured savings. As in the first competition, neural networks were the most popular method among the top ranked contestants. However, in contrast to the first competition, a well-assembled multiple linear regression model showed comparable performance with the neural network models [37]. In addition, although there were small differences in the Coefficient of Variation (CV) among the different models, large variations in the predicted savings were observed. This was most likely caused by different assumptions between the models. This also implies the difficulty of calculating savings in a reliably manner as models that have a small CV might not necessarily be indicative of better models. In 2019, ASHRAE held a third competition [38]. The contestants were given hourly energy consumption data that includes electricity, steam, chilled water, and hot water from 1448 buildings with meteorological data and building information such as primary use of building and gross floor area. Most winners used ensembles of multiple machine learning algorithms. Unlike the last two competition, a technique called gradient

boosting was used most dominantly. In addition to the selection of methodology,

preprocessing of the given data played a key role that differentiated top performing

methods from others.

The summary of the background is as below:

1. There are two methods of building energy modeling: forward modeling and inverse

modeling. Forward modeling uses the principles of thermodynamics and heat transfer,

and requires a large number of parameters related to building properties. Inverse

modeling uses statistical methods, and requires smaller numbers of inputs than the

forward modeling. The building energy inverse model is made using the building end

use energy.

2. There are a wide variety of methods for the inverse modeling of building energy use

that include simple linear or multiple linear regression, time-series analysis, and machine

learning.

3. To test the performance of different building energy inverse modeling methods,

ASHRAE held three competitions. In the two competitions, the majority of winners used

ANN. However, there was a large difference in the performance for the same ANN users

depending on how they preprocessed and organized the given input data. In the second

competition, a contestant using the multiple linear regression had the second highest

score, showing that a well-prepared conventional regression could be as good as or

better than more complicated and labor-intensive new methods. In the third competition,

gradient boosting was most dominantly used, and most winners used ensembles of

multiple machine learning algorithms. Similar to the last two competitions,

preprocessing of the given data critically affected the effectiveness of models.

CHAPTER III LITERATURE REVIEW OF SVM FOR BUILDING ENERGY MODELING

Numerous researchers have applied SVM to building energy use modeling or to compare a variety of inverse modeling.

Dong et al.[31] introduced the SVM for building energy estimation in 2005. In this study, Monthly outdoor air temperature, relative humidity, and solar radiation were used as inputs to predict the energy use of 4 commercial buildings. More than 2 years of data was used for training the SVM model, and one year of data was used for testing. Although it is significant in that it is considered to be one of the first papers using a SVM on building energy modeling, the performance of SVM was difficult to analyze because they did not compare the result with other existing methods on the same dataset.

Many studies have shown that the performance of SVM is better than or similar to that of ANN. For instance, Li et al. [6] used SVM and a Back Propagation ANN to predict hourly office building cooling loads. Outdoor air temperature, relative humidity, and solar radiation were used as independent variables. SVMs showed better performance than the ANN. Li et al. [28] used three types of ANNs, namely back propagation neural network (BPNN), the radial basis function neural network (RBFNN) and general regression neural network (GRNN), along with a support vector machine (SVM) to predict hourly cooling consumption of an office building. For the inputs of each model, current and past outdoor air temperatures, current relative humidity, and current and past solar radiation were used. Cooling load values were generated from building energy simulation. The results showed that the SVM and the GRNN methods achieved improved accuracy and generalization

when compared to the BPNN and RBFNN methods. Li et al. [39] used LS-SVM to predict hourly cooling loads of an office building. In their study, Present and historical outdoor air temperature, present humidity, present and historical solar radiation were used as inputs. When the performance of LS-SVM was compared to that of BPNN, the LS-SVM showed better results. Zhao and Magoulès [7] classified prediction methods into five categories: engineering methods, statistical methods, neural networks, support vector machines, grey models. Their study also showed SVMs showed superior performance over ANNs in many cases. In addition, the level of accuracy of the SVM was in the same range as elaborate engineering simulation models.

In addition to the previously mentioned studies, one study run SVM with a large number of independent variables. Zhao and Magoulès [40] used SVM to predict hourly office building electricity and district heating energy use. In their study, 24 independent variables were used, including schedules, weather conditions, infiltration, zone mean temperatures, and heat gain from lights. Five months of data were used for training, and for testing, two days of data were used. The R^2 for the electricity prediction was 0.9599, and the R^2 for the district heating was 0.9277. Although the testing period was short, this is a good example that shows SVM can make an excellent hourly inverse model with numerous independent variables.

Selecting which independent variables to use can be a topic within itself for an of inverse model, this process is also called feature selection. Massana et al. [8] compared MLR, Multilayer Perception (MLP) and SVR for non-residential building forecasting of hourly, daily and monthly electricity use. In this work, they also analyzed which attributes were most relevant. The result showed MLR and MLP provided the best result with temperature,

calendar and occupancy as independent variables. SVR provided the best results with only

temperature and occupancy data. The MLR model, unlike the other models, offers

coefficients that have physical significance, which informs the relevance of each

independent variable. However, MLR had a lower goodness of fit versus MLP and SVR;

and the SVR model provided the highest accuracy. All the models showed higher accuracy

with real occupancy data versus assumed calendar data. Surprisingly, the indoor ambient

data did not enhance the performance. Finally, the feature selection process did make it

possible for models with few independent variables to indicate an accuracy similar to the

ones with more independent variables.

Recently, many researchers reviewed and did comparative studies of many different types

of machine learning inverse models including a couple of traditional statistical ones. Deb

et al. [41] reviewed 9 time series forecasting techniques for building energy consumption:

ANN, ARIMA, SVM, CBR, Fuzzy, Grey, Moving Average & Exponential Smoothing

(MA & ES), K – Nearest Neighbor prediction method (kNN), and Hybrid. According to

the assessment, SVM performs well for long-term data with small data frequency and for

non-linear data, but time-consuming calculation was pointed out as its weakness. This

assessment of SVM is in line with Wei et al. [42] who also pointed out low processing

efficiency as drawback of SVM. Contrarily, Wang et al. [4] mentioned another aspect of

SVM as its advantage, which is relatively fewer number of inputs to tune as an advantage

of SVM. Yildiz, et al [43] compared several models to predict day ahead hourly and daily

peak electricity load: Multiple Linear Regression (MLR), Neural Network with Levenberg

Marquitd (NN LM), Neural Network with Bayesian Regulation Backpropagation (NN

BR), Nonlinear Autoregressive Network with Exogenous Inputs with Levenberg Marquitd (NARX LM), Nonlinear Autoregressive Network with Exogenous Inputs with Bayesian Regulation Backpropagation (NARX BM), Regression Trees (RT), Support Vector Regression (SVR). Most machine learning models used in the study indicated a better forecast performance than the MLR models. However, there is a study that showed the opposite result. For instance, Tereshchenko et al. [44] assembled a couple of linear regression models of different schedules for temperature dependent period and hourly temporal profiles for temperature independent period to predict the hourly domestic hot water plus space heating energy. This method showed better results than SVM although the comparison was not entirely fair in the sense it the input of schedule information was used only to the linear regression.

There are studies that has provided a valuable insight into how machine learning-related building energy model study has proceed so far. For instance, Wang et al. [4] reviewed 35 papers and derived statistical figures on several basis. In the research, 42% is for educational and research space and 33% is for commercial space. Residential accounts for only 17%. As for the chosen inverse modeling method, ANN takes up the largest portion (42%), followed by statistical regression (26%), and SVR is used in 12% of the papers. Regarding energy type, whole building electricity takes up the biggest portion (57%), followed by cooling (13%), heating (11%), combined heating and cooling (11%), and others (8%). In terms of prediction time scale, hourly prediction is most dominant, followed by daily (19%), and yearly (8%). Remaining portion include monthly, weekly, 15-minute, and minute data. The research divides input data into three types: meteorology,

occupancy, others. Meteorological data occupies the largest portion (60%). Occupancy takes up only 29% despite its impact on building energy use. This stems from difficulty in data acquisition. Others (54%) includes historical energy consumption data, indoor environmental information, building characteristics, operational schedule and day type. Similarly, Amasyali and El-Gohary [5] compares 63 papers related to machine learning application to building energy and sorts out them according to several attributes. 81% of the reviewed papers worked on non-residential buildings and 19% on residential ones. Regarding temporal granularity, 12% did research on sub-hourly data, 57% on hourly, 15% on daily, 4% on monthly, and 12% annual data. Concerning the type of inverse modeling methodologies, ANN takes up the most portion of 47%, followed by SVM of 25 %. Decision Tree constitute 4%. Others methodologies include General Linear Regression, Polynomial Regression, Exponential Regression, Multiple Linear Regression, and so on. The size of input data set varies from 2-week to 4 year. 56% uses one month to one-year long data. 31% uses longer-than one year. Only 9% uses shorter-than on month data.

The conclusion of literature review is summarized as below:

- SVM has been the second most widely used machine learning algorithms in building energy inverse model research after ANN

- SVM has fewer parameters to determine compared to ANN, but in many cases, it has been reported to show higher accuracy than ANN.

- Despite usually higher accuracy of machine learning techniques, linear regression models still popular because of its relative simplicity, interpretability, and fair accuracy.

- Outdoor air temperature is by far the most frequently used input to SVM as well as other inverse models. Other meaningful meteorological data include humidity and solar radiation. Occupancy and schedule data also can add accuracy to models. When a large number of attributes are used as input to SVM, its performance is comparable to elaborate engineering models.

- SVM and other machine learning methods have been usually applied to hourly and daily application.

- Given the large amount of comparative study involving SVM, change-point regression has been rarely considered.

Table 1 - Summary of the Literature Review on SVM Application to Building Energy Modeling

	Description
Popularity of SVM in building energy modeling area	1. Within machine learning application area, SMV has been second most popular after ANN. 2.Including all forms of inverse modeling methods, traditional statistical methods such as change point linear regression or multiple linear regression are still popular due to its ease of use and fair accuracy
Typical independent variables used	1. Meteorological data is by far most widely used; within this category, temperature, humidity, and solar radiation are generally used.

Table 1 Continued

	Description
Typical independent variables used	2. Building schedule and occupancy can be added to increase the predictability. Between these two, the effects of the former have been more widely studied due to data availability.
	3. Historical weather and consumption data can be used for times series data to account for time-lag effects if the data frequency is smaller than daily.
	4. With further detailed data, the SVM performance can be closer to that of forward modeling.
Data frequency	Hourly and daily application is general. Annual, monthly, and sub-hourly application was less frequently mentioned in the research field.
Advantages	1) High accuracy
	2) Flexibility (SVM can be applied both linear and non-linear data)
	3) Ease of use (due to relatively fewer number of parameters to adjust compared to other machine learning algorithms)
Disadvantages	1) Expensive computational cost (Calculation time increases in proportional to the square of data size, and it even with smaller data, it cannot be faster traditional statistical methods)
	2) Inconvenient to apply to big size data (caused by the first issue)
	3) Lack of interpretability (which is the limit of machine learning algorithm)
Others	Change-point regression has been rarely considered in the studies of SVM application to building energy inverse modeling.

CHAPTER IV PROCEDURE AND METHODOLOGY

The key principle of SVM regression consists of a couple of steps. First, the basic regression model is expressed as follow:

$$f(x) = \boldsymbol{\omega} \cdot \boldsymbol{x} + b \qquad \text{(EQ. 14)}$$

Second, the user must determine the size of a boundary, which is supposed to deviate from the future regression model by the length of ε. Third, an ε-insensitive loss function is set up. This function measures how each data set deviates from the regression model, and if the deviation is lower than ε, it results in zero; otherwise, deviation minus ε becomes the function value. Fourth, the sum of $\boldsymbol{\omega}$ and the loss function for all given data sets is minimized. Moreover, the relative weight of the term can be adjusted by another user-defined parameter, C. This minimization function is expressed as follows:

$$\text{Min} \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C \sum_{i=1}^{n}(\xi_i + \xi_i^*) \qquad \text{(EQ. 15)}$$

(a) ε-insensitive Loss Function   (b) SVM regression plot and ε
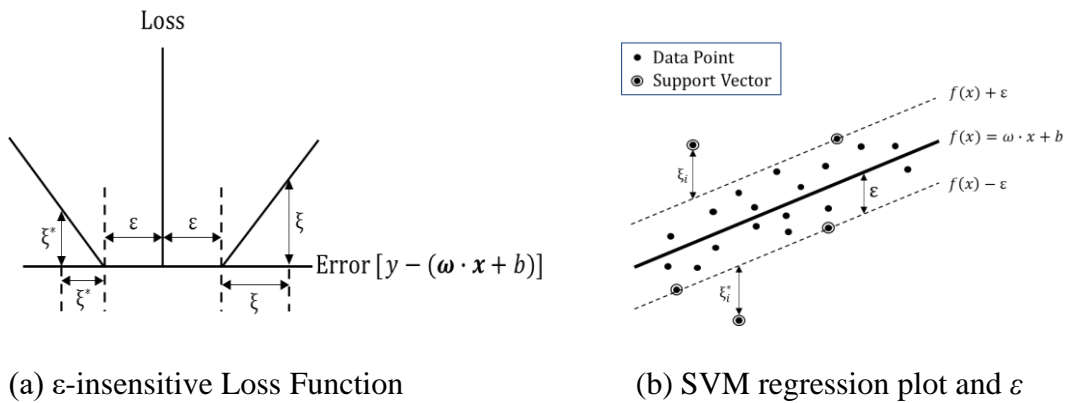
Figure 3 - SVM Linear Regression

The resultant regression plot is attained as a result of this minimization function. By using a technique called Kernel Trick, SVM can also cover a non-linear dataset. For a more detailed explanation of the SVM fundamentals, see Appendix A.

SVM has many parameters, and those that must be determined by the user before running SVM are called hyperparameters. A variety of options can search for optimal hyperparameters. This search procedure is called tuning. The performance of SVM depends on the appropriate choice of hyperparameters and tuning methods. Thus, the first section of the experiment will select the tuning method used in the remaining sections.

SVM is well known for its flexibility and high accuracy. Observing these characteristics constitutes the second part of the experiment. The experiment will expose SVM to many different types of data sets. The data is either actual building energy data or synthetic data that resembles actual data.

SVM is a supervised learning algorithm. Therefore, it requires training to make a model. The training data set is a combination of independent and dependent variables. If the relationship between independent and dependent variables of the training data set is not strong, SVM cannot produce a good model, because SVM is supposed to capture the pattern-connecting independent variables to dependent ones, which are assumed to exist. Hence, if the link is weak in the first place, there is little for SVM to learn from the given training data set. In this sense, the third section of the experiment will investigate the influence of different independent variables on building-energy modeling.

In the last part, the performance of SVM regression is compared to two other statistical regressions, which are multiple linear regression and change-point linear regression.

Attention should be paid when comparing the performance. Unlike traditional statistical methods, the performance of the machine-learning algorithm is assessed by the performance in the training period and the testing period, as the machine-learning algorithm can exponentially raise its training period performance merely by memorizing the data. This risk can be effectively eliminated by using a new data set that was not used during the training period. The performance measure derived from this new data set is called testing-period performance or simply testing performance. However, the testing-period performance also has limitations because a new energy consumption pattern may have started during the testing period. In this case, the reliability of the testing period will also be limited. Moreover, the performance of SVM depends on many factors, such as hyperparameters, tuning methods, and the quality of the given data set. Therefore, a qualitative comparison is essential in addition to the quantitative comparison. As a platform to operate SVM, the programming language R is used. R is a free and open-source programming language and environment for statistical computing, initially written by Ross Ihaka and Robert Gentleman at the Department of Statistics of the University of Auckland in Auckland, New Zealand [45]. The functionality of R can be easily extended by installing a variety of packages available online, and it can be also used to implement machine-learning algorithms. The functions in R can be written in R itself, but for computationally intensive tasks, other languages such as C and Fortran code can be used.

To implement the SVM in R, a package named 'e1071', which contains many statistical and machine-learning functions, such as fuzzy clustering and SVM, was used [46]. In the package 'e1071', the SVM library, which is called LIBSVM, was written by Chang and Lin [47]

Concerning what is stated above, the experiment consists of the following sections:

1) Determination of the SVM Implementation Settings

2) SVM Regression Pattern Observation

3) Investigation of the influence of different independent variables

4) Comparative study

4.1 The Procedural Description of Experiments

4.1.1 Determination of the SVM Implementation Settings

The performance of SVM depends on the choice of kernel function and hyperparameters.

The choice of kernel predetermines the expected shape of the regression model. The

selection of hyperparameters determines the regression quality.

4.1.1.1 Kernel Function

As for kernel, radial basis function (RBF) is predominant for regression purposes. This is

because its high flexibility allows the SVM algorithm to fit any form of data. Unlike RBF,

linear kernel constraint the shape of its model to be linear. Likewise, polynomial kernel

also limits its model only to the shape that can be expressed by polynomial expression.

These two may be appropriate where the given data set neatly falls under these categories

or the dataset should be limited to those shapes. However, this study selected SVM

because of its flexibility, thus limiting the flexibility is against the object. Hence, RBF is used as a kernel by default.

4.1.1.2 Hyperparameters Selection and Tuning

RBF kernel SVM has three hyperparameters to specify: epsilon, cost, and gamma. (See Appendix B for details) When selecting the optimal hyperparameter set, or tuning, there are many methods to search for optimal parameters. Each method has its benefits and drawbacks. Essentially, the competition between different methods centers on the trade-off between accuracy and speed. The efficient selection of hyperparameters is another significant research topic. In this research, two methods are used for comparison. The first is the most conservative method and the other is a newly proposed modified grid-search method.

4.1.1.2.1 Exhaustive Grid Search (EGS)

An exhaustive grid search, or more simply stated, a grid search, is the most conservative, robust, but generally used tuning method. The principle is straightforward. It defines a grid in which each dimension represents one hyperparameter. In the cases of RBF SVM, the grid has a total of three dimensions, which will check every possible combination of hyperparameters represented by a point in the grid, one at a time. Although it is very time consuming, it guarantees robust performance.

4.1.1.2.2 One Dimensional Grid Search with Recommended Initial Values (ODGS)

This is a newly proposed method in this research. However, most of the key ideas are from other literature.

Tang et al. [48] took an analytical approach, the golden section search algorithm specifically, to determine the initial gamma and therewith to conduct a one-dimensional grid search to find the optimal C. With the obtained C, the grid search is repeated to determine the optimal gamma. However, they did not mention epsilon.

Cherkassky et al. [49] proposed a formula to obtain C based on statistical reasoning:

$$C = max(\bar{y} + 3\sigma_y , \bar{y} - 3\sigma_y)$$
(EQ. 16)

Where $\bar{y}$ is the mean of training data, σ_y is the standard deviation of training data, and σ is the standard deviation of input noise.

Kaneko and Funatsu [50] referenced the literature of Tang et al. [48] and Cherkassky et al. [49]. and proposed a procedure to efficiently find the hyperparameters epsilon, C, gamma. It uses the Eq.16 to find an initial C and uses the analytical approach suggested by Tang et al. [48] to determine an initial gamma. With two initial hyperparameters, it conducted a one-dimension grid search to find the optimal epsilon. With the optimal epsilon and initial gamma, the optimal C is attained by a one-dimension grid search. The optimal gamma is obtained in the same way. This method demonstrated almost the same performance as exhaustive grid search.

	ε	C	γ
Initial State		$\text{Max}(\bar{y}+3\sigma_y, \bar{y}-3\sigma_y)$	Golden Section Method

One Dimension grid search over a range of ε

| | Optimized ε | Same as above | Same as above |

One Dimension grid search over a range of C

| | Optimized ε | Optimized C | Same as above |

One Dimension grid search over a range of γ

| Final State | Optimized ε | Optimized C | Optimized C |
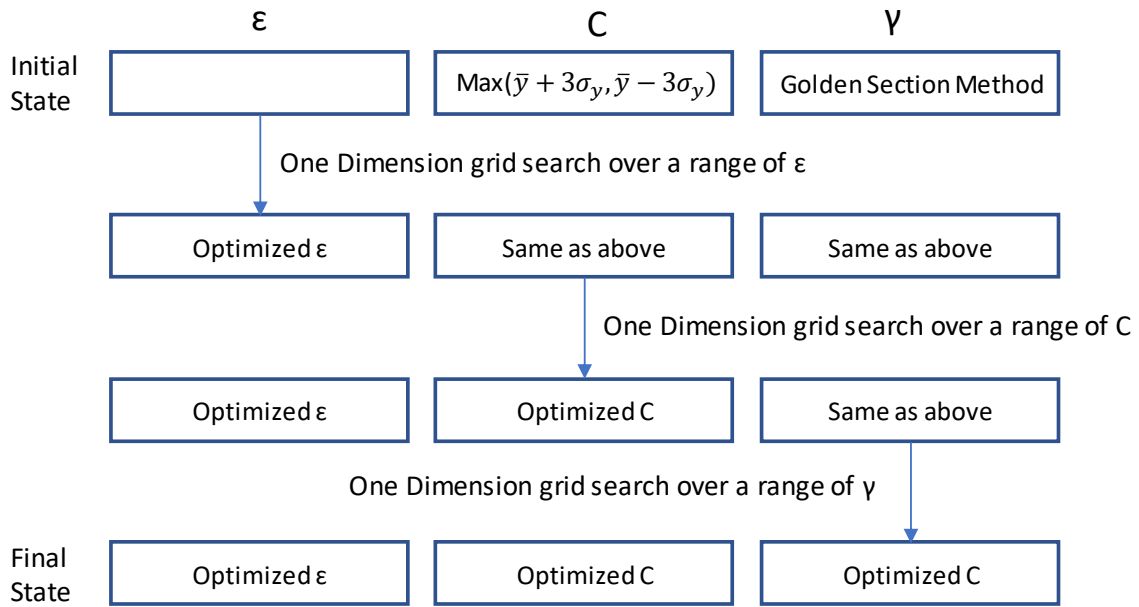
Figure 4 - Flow of Hyperparamter Tuning Method Proposed by Kaneko and Funatsu [50]

The proposed method in this research is a simpler version of the method suggested by Kaneko and Funatsu [50]. The difference is the method to obtain the initial gamma. Instead of using the analytical approach, an empirical value is used. The empirical value is shown as follows and it references the literature of Louw and Steel [51]:

$$\gamma = \frac{1}{p} \qquad \textbf{(EQ. 17)}$$

where *p* is the number of the input attribute.

The motive for using the empirical value instead of an analytical approach lies in its simplicity and reduced calculation time.

	ε	C	γ
Initial State		$\text{Max}(\bar{y} + 3\sigma_y, \bar{y} - 3\sigma_y)$	1/p

One Dimension grid search over a range of ε

	ε	C	γ
	Optimized ε	Same as above	Same as above

One Dimension grid search over a range of C

	ε	C	γ
	Optimized ε	Optimized C	Same as above

One Dimension grid search over a range of γ

	ε	C	γ
Final State	Optimized ε	Optimized C	Optimized C
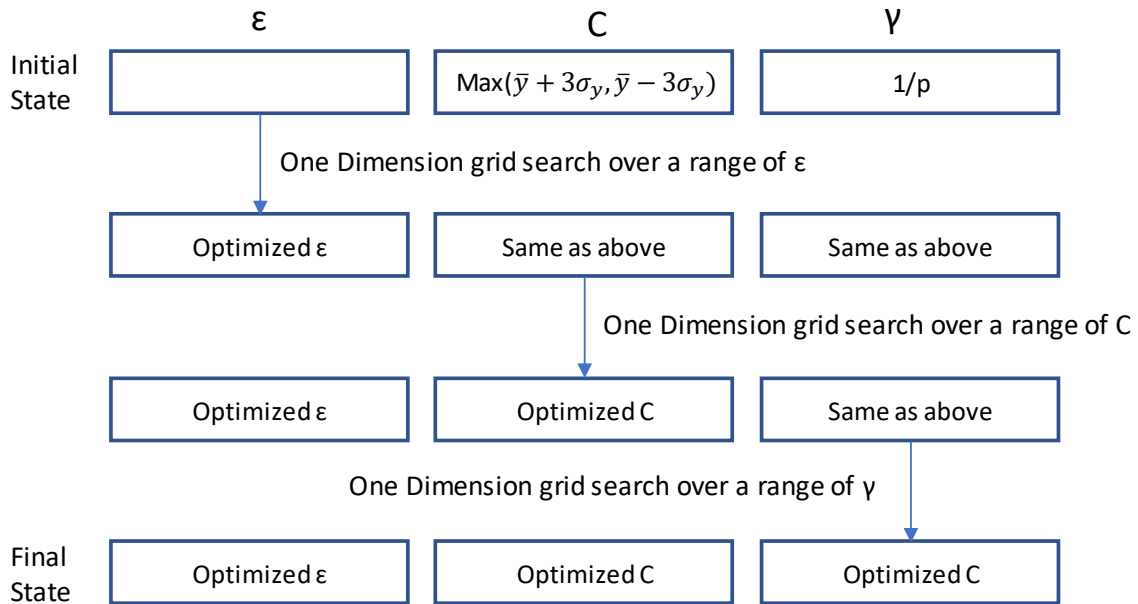
Figure 5 - Flow of Hyperparamter Tuning Method Used in This Research

Repeated K-Fold Cross Validation

A grid search usually accompanies cross-validation (CV) to prevent optimistically biased tuning, in other words, overfitting. Overfitting occurs because the SVM algorithm is created to make sense of given data in its way without any explanatory equations. This may result in the regression model fitting the given data perfectly but result in a far worse performance with new data. CV effectively addresses this by dividing the data into *k* folds, training a model with *k*-l fold data, validating it with the other fold, and repeating it *k* times with a different fold for validation, and averaging the result to specify the optimal result. Although CV is usually robust by itself, it may return an inconsistent performance. This happens in the stage when the training data set is divided into *k*-fold because there are many partitioning cases. The inconsistency caused at this stage is not significant when the

data set is clean, but it can affect the result if the data set is not clean, such as having many outliers. The partition is usually done by a random computer algorithm by default. To minimize this inconsistency, CV is often repeated multiple times to reduce the bias in the result. In this study, 5-fold cross-validation is used by default, and 10-repeated 5-fold cross-validation is used for comparative study.

4.1.2 Pattern Observation

The fitting performance of the SVM regression is visually investigated with synthetic and actual data. The purpose is to graphically understand the way in which the SVM algorithm regresses data points and to check whether the pattern looks reasonable. SVM regression generally works with multiple independent variables. However, for the ease of visual observation and considering that a single input attribute SVM regression is the main option of this research, the number of independent variables is limited to one. As a kernel function, the radial basis function is used by default because it is predominantly used for non-linear regression. If the given pattern is linear, the linear kernel option is also used for comparative purposes. The number of cross-validations is set to five unless noted otherwise. For tuning, a one-dimensional grid search with recommended initial values (ODGS) is used. The validity of this method is tested in the next section.

The observation consists of the following three parts: linear, piecewise linear (two pieces), and special cases. For the linear case, both linear kernel and RBF kernel are used for comparative purposes. The piecewise linear case consists of two parts: the case of one piece constrained to a zero slope and of both pieces having no constraint. The former

corresponds to a 3P model of change point linear regression and the latter 4P. The special

case parts handle various cases: piecewise linear (three pieces) patterns, hourly data (8760

data points), data showing different patterns according to schedules, scattered data, and

data with breakpoints. Fu et al. [52] introduced the concept of break point was introduced

to describe the case where the whole building's heating hot water consumption spikes in

the moderate temperature range. The linear, 2-piecewise linear, and 3-piecewise linear

patterns are investigated with synthetic data with both ideally clean data and noisy data.

Different levels of noise are used to observe how SVM regression responds to varying

noise (10%, 30%, 50%, and 100%). Two types of noise are used: homoscedastic and

heteroscedastic noise. The former refers to the noise that is consistent throughout the

whole range of independent variables. The latter refers to the noise that varies throughout

the range of the independent variables. The degree of noise is calculated based on the

deviation from the line where the ideal data would lie divided by the mean of the

dependent variable. They are analyzed according to a monthly and daily frequency. For

hourly, scheduled, scattering and breakpoint patterns, actual building energy use data with

daily frequency is used. The following Table 2 is the summary of the pattern observation

plan.

Table 2 - SVM Pattern Observation Plan

Data Patterns	Description	Data Frequency	Data Type

Linear	Use RBF kernel*	Daily	- Ideally clean synthetic data
			- Synthetic data with 10% noise
			  1) homoscedastic noise
			  2) heteroscedastic noise
			- Synthetic data with 30% noise
			  1) homoscedastic noise
			  2) heteroscedastic noise
			- Synthetic data with 50% noise
			  1) homoscedastic noise
			  2) heteroscedastic noise
			- Synthetic data with 100% noise
			  1) homoscedastic noise
			  2) heteroscedastic noise

	Use RBF kernel*	Monthly	
	Use Linear kernel	Daily	
	Use Linear kernel	Monthly	
Piecewise Linear (2 pieces)	one piece no slope	Daily	
		Monthly	
	both pieces sloped	Daily	
		Monthly	
Special	Piecewise (3 pieces)	Daily	
		Monthly	
	Hourly data	Hourly	Actual data
	Scheduled data	Daily	Actual data
	Scattering data	Daily	Actual data
	break pointed data	Daily	Actual data

* RBF kernel function is used by default in this research unless noted otherwise.

4.1.3 Investigation on the Influence of Different Independent Variables

SVM is a supervised learning algorithm, and its performance depends on how relevant the

input data attributes are to response data. Thus, it should be noted that it does not

automatically fit the data to its regression model without relevant categorical information.

In this sense, using proper attributes as input is essential. As dependent variables, whole

building chilled water (CHW) and heating hot water (HHW) are used. As stated in Chapter

3, SVM input attributes include outdoor weather conditions, indoor environmental

conditions, time features, past weather conditions, energy consumption, occupancy

schedule, building operation schedule, and physical information regarding the building.

While more information is usually better than less, unnecessary or excessive attributes only

increase the calculation time. In building energy baseline modeling, meteorological data is

the most influential factor and is relatively easy to collect. Concerning the meteorological

data, outside air temperature (OAT) is the most dominant factor.

Both chilled water and heating hot water are sensitive to OAT. When the outdoor dry-bulb

temperature is high, the use of chilled water increases. When the outdoor dry-bulb

temperature is low, the use of heating hot water increases. However, in the case of chilled

water, outdoor dry-bulb temperature alone does not provide enough of a clue about its

energy usage pattern. Considering both temperature and humidity provides a clearer

explanation. The main source of humidity is usually outdoor air entering the building

through ventilation. Different buildings have different ventilation design strategies and

requirements depending on their type. In this research, humidity is considered in three

different forms. For the first form, dew point temperature is added as an additional

independent variable (OAT + DPT). Although this is a straightforward manner to consider

humidity, collinearity between these two factors makes this method less reliable. The

second form is to use outdoor air enthalpy (OAE). Enthalpy is calculated by summing the

sensible heat and latent heat of the moist air in the atmosphere. The following is the

equation for deriving OAE.

$$h = h_a + Xh_w \tag{EQ. 18}$$

where h is specific enthalpy of moist air, h_a is specific enthalpy of dry air, h_w is specific

enthalpy of water vapor, and X is humidity ratio.

$$h_a = c_{pa}t \tag{EQ. 19}$$

Where c_{pa} is the specific heat of air at constant pressure and t is air temperature.

$$h_w = c_{pw}t + h_{we} \tag{EQ. 20}$$

Where c_{pw} is the specific heat of water vapor at constant pressure, t is water vapor

temperature, and h_{we} is evaporation heat of water at 32°F. The third form is to use

operational effective enthalpy (OAE). Li and Baltazar [53] introduced this term and

showed that regression performance can be further improved using this OAE as a

regressor. The idea is to adjust the inclusion of humidity by considering what range of

temperature and humidity practically affect the use of chilled water. Latent loads increase

the chilled water consumption only when condensation occurs at cooling coils. In this

sense, typical cooling coil leaving air temperature of 55F and 95% RH was chosen as a

reference to decide whether to take humidity into the consideration of the analysis of

chilled water use. The following is the equation for deriving the OEE.

$$h_{oee} = h_s + (h_l - h_{ref})^+ \tag{EQ. 21}$$

Where h_s is specific enthalpy of dry air, h_l is specific enthalpy from moisture, and h_{ref} is

reference enthalpy. For evaluation metric and sample size related to this investigation, see

4.2 and 4.3, respectively.

4.1.4 Comparative Study

Multiple linear regression and change point linear regression are widely used models for

establishing building energy baselines. Multiple linear regression is the simplest type of

regression methods, yet still commonly used in many fields because of its robustness. As a

dependent variable, CHW is used. HHW is excluded because it is affected mostly by one meteorological variable, outdoor dry-bulb temperature.

The change-point regression model, or piecewise linear regression model, in other words, fits well with the application to creating building energy baseline models because buildings consume energy that way. Hence, the strongest advantage of the method is that it provides a physical interpretation of modelers. The drawback, however, is that it cannot catch non-linear or complex patterns. As dependent variables, CHW and HHW are used. Although the change point linear regression models include 3P, 4P, and 5P models, only 4P is used as an object of comparison. This is because 4P usually shows better performance than 3P in terms of reported accuracy even when buildings consume energy on the 3P model. 5P is excluded because it is not a generally applied model when the dependent variable is CHW or HHW. The performance of these two models are compared to that of SVM models in many different levels such as in terms of data frequency, energy type, and input attributes. As dependent variables, whole building chilled water (CHW) and heating hot water (HHW) are used. To implement the Change Point Linear Regression, an algorithm proposed by Kissock et al. [15] is used. For evaluation metric and sample size related to this study, see 4.2 and 4.3, respectively.

4.2 Evaluation Metric

Establishing clear criteria of assessment is essential so that experiments can be considered reliable. The experiment in this research is measured by two different types of criteria. One

is for assessing the fitting performance of the model and the other is to evaluate the

significance of the different fitting performances.

4.2.1 Fitting Performance

1) Coefficient of Variance (CV)

CV measures the gap between the actual values and predicted values regardless of being

positive or negative. The value is relative to the mean of actual values.

$$CV = \frac{\sqrt{\frac{1}{N-1}\Sigma_{i=1}^{N}(y_i-\hat{y}_i)^2}}{\bar{y}} \times 100 \qquad \text{(EQ. 22)}$$

Where $\hat{y}_i$ is the predicted energy consumption, y_i is the actual energy consumption, $\bar{y}$ is the

average energy consumption, N is the number of samples

2) Coefficient of Determination (R^2)

R^2 is an index to indicate how much the variance of total data can be dealt with by the

regression model. The formula of R^2 is as follows.

$$R^2 = 1 - \frac{\Sigma_I(y_i-\hat{y}_i)^2}{\Sigma_I(y_i-\bar{y})^2} = \times 100 \qquad \text{(EQ. 23)}$$

Where $\hat{y}_i$ is the predicted energy consumption, y_i is the actual energy consumption, $\bar{y}$ is the

average energy consumption, If the regression model successfully explains the pattern of

dependent variables well, the variance of the independent variable from the regression

model will be much smaller than the variance from the mean of the dependent variable.

This will make R^2 close to 1. On the contrary, if the regression model does not describe the

data pattern well, the variance of dependent variables from the regression model will not be much different from that from the mean of the dependent variables. This will make R^2 smaller.

There is an important factor to note when reporting the performance of machine-learning methods. Unlike statistical regressions, the performance of the machine-learning algorithm is assessed in two levels. This is because there is a chance that the algorithm raises its performance not by learning, but by aimlessly memorizing the training data set. The data of the testing period is supposed to catch it if it happens. Thus, it is the testing performance that counts as the 'actual' performance of a machine-learning algorithm. However, testing data does not always work as it is intended when it is used for building energy baseline modeling, unfortunately. The energy use patterns of a building varies for many reasons, such as a change in operational strategy, renovation, and unexpected events. In this case, the testing period performance would be much different from trading period performance even when the SVM model catches the pattern of training data well without overfitting. Concerning such an issue, it is worthwhile to pay attention to the fact that except for the one case, all other experiment is conducted with one input attribute. There are advantages when only one categorical data is used as input. First, it provides a clear description of what the SVM algorithm generated out of training data set because they can be clearly displayed on a two-dimension plot. The second benefit comes from the first one. The prevention of overfitting becomes easier. Without using testing data, which is independent of training data set, implementers can intuitively check if the regression is reasonable. These two benefits make the training period performance as significant as or even more significant than the testing period performance.

4.2.2 Significance of Different Fitting Performance

To quantitatively assess the difference in fitting performances among the different independent variables or different regression models, a t-test is adopted. The t-test is a statistical method that compares the mean between different groups. It is a type of hypothesis test. With limited information from samples, experimenters quantify the significance of their observation, and either reject or accept the null hypothesis. Depending on the case, three different types of t-test can be used: a one-sample t-test, a two-sample t-test, and a paired t-test. A one-sample t-test is used to compare the mean of one sample group to another hypothetic mean value. A two-sample t-test compares the means from two groups. A paired t-test compares the mean of two groups and the samples from two groups are not independent, such as the case where two different treatments are applied to the sample object. Regardless of such the differences, the way each t-test assesses the significance from observation is fundamentally the same. As a first step, t-value is calculated as follows:

$$t = \frac{\bar{x} - \mu_0}{\frac{s}{\sqrt{n}}} \qquad\qquad\qquad \textbf{(EQ. 24)}$$

Where $\bar{x}$ is a sample mean, μ_0 is the mean value of the null hypothesis, s is the standard deviation of a sample, and n is a sample size. A more intuitive approach to the formula is to view it as a signal-to-noise ratio. The purpose of experiments, in general, is to find or prove a significant difference, or sometimes non-difference, between groups. The extent of this difference, or signal, is located on the numerator. However, for any experiments that involve sampling groups, there is a risk of being misled by what happens by chance. The

extent of the risk, or in other words, noise, can be quantified by the standard deviation of the sample. Such a risk reduces as the sample size increases, as seen in the Eq.24

However, just deriving the *t* value is not sufficient to provide meaningful information. This value should be interpreted in a broader context. It is *t*-distribution that provides the context. (See Figure 6)
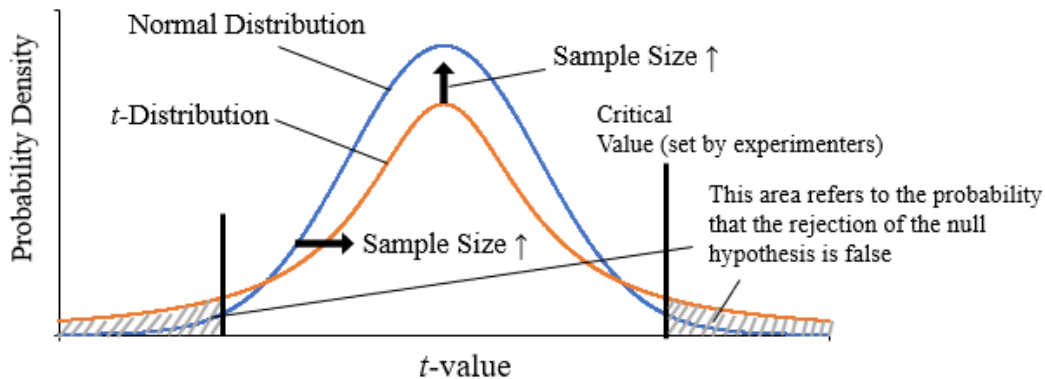
Figure 6 - Generic *t*-distribution Representation

For a given *t* value, *t*-distribution indicates the probability for the null hypothesis to be true. What should be noted about the interpretation of the distribution graph is the way to calculate the probability. It is attained by the area under the curve, not by the value of the vertical axis. The probability calculated from the distribution curve can be translated to a more generally well-known term, *p*-value. The shape of the probability distribution varies depending on the sample size. The more the sample size is, the more probability distribution is concentrated on the center, and *t*-distribution becomes closer to normal distribution. This trend stems from the premise of *t*-distribution, which assumes that the

null hypothesis is true. With this premise, it is natural that more data changes the distribution in a way that supports the null hypothesis.

The experiment of this study is carried out in such a way as to apply two different methods to the same sample buildings. In other words, samples from different groups are not independent. Thus, a paired *t*-test is used to assess the result of the following set of experiments.

a. Input Attribute study

- Outdoor Air Temperature (OAT) vs Outdoor Air Enthalpy (OAE)

- Outdoor Air Temperature (OAT) vs Operational Effective Enthalpy (OEE)

- Outdoor Air Enthalpy (OAE) vs Operational Effective Enthalpy (OEE)

b. Comparative study

- SVM vs MLR

- SVM vs CPLR

4.3 Sample Size Determination

The sample size is related to the reliability of experiments. Since a paired *t*-test is adopted as an evaluation metric, the sample size is determined in connection with that. To effectively reject or accept the null hypothesis of a paired *t*-test, which states that there is no significant difference between groups, a reasonably high number of samples is needed. However, increasing the number of sampling comes with increased calculation cost and time. Thus, the optimal sampling number needed to be determined before the experiment.

Type I Error (α)

Type I error, generally passing for α, is the case where the null hypothesis is rejected when it is true.

Type II error (β)

Type II Error, generally passing for β, is the case where the null hypothesis is not rejected when it is false.

Effect Size

Effect size is a measure of how far the mean of one group is from that of another, or how far the mean of a treatment group is from that of a controlled group on a standardized scale. This is equivalent of the numerator of t-value or t-statistic. If the effect size is small, the more number of sample is needed to prove the significance of the difference between a treatment group and a controlled group. Likewise, if the effect size is substantial, then a small sample is enough to prove the significance of the experiment.

Power

Power is the probability of rejecting the null hypothesis when it is false. In other words, it can be described as the probability of not making a type II error. Thus, mathematically, it is equal to $1 - \beta$. Higher power means a higher chance of the alternative hypothesis being correct. The typical value of β is 0.2, which corresponds to 0.8 of power.

Table 3 - Comparative Table of Type I Error, Type II Error, and Power

		Null Hypothesis	
		Correct	Wrong
Decision	Fail to Reject the Null Hypothesis		Type II Error (β)
	Reject the Null Hypothesis	Type I Error (α)	Power

Power increases when a sample size increase, effect size increases, and when alpha decreases. The relation can be seen in the Figure 7 below.
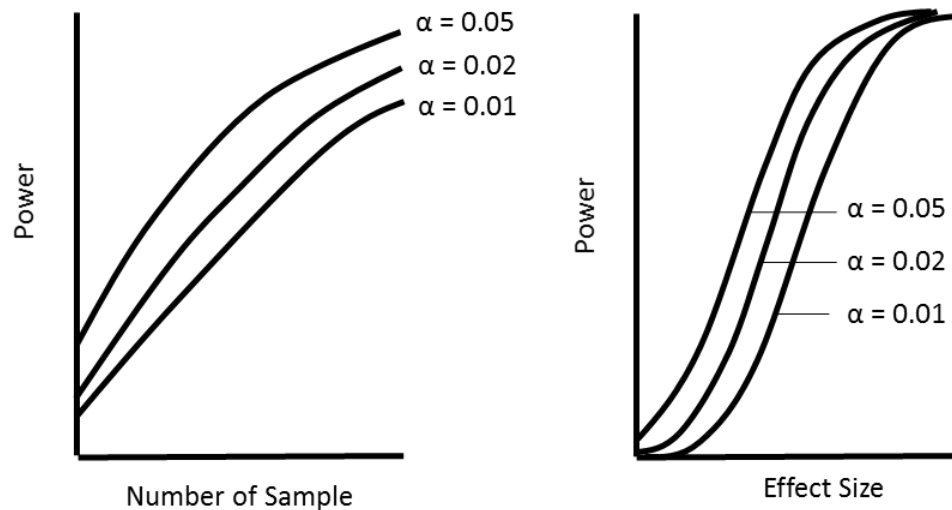
Figure 7 - The Relation between Sample Size, Effect Size and Power. (Adapted from Krzywinski and Altman, 2013 [54])

Cohen [55] listed numerous sample size equations and tables for different cases including different statistics, confidence, and the number of factors. For the determination of the paired *t*-test, significance level (α), power (p), and effect size (d) must be specified first. Once they are decided, a recommended sample size can be found on the sample table [55].

The level of significance (α) is called Type 1 Error and it is equal to one minus confidence. For instance, the confidence level of 95% corresponds to the level of confidence of 0.05.

In the application to sampling size determination, an effect size must be estimated based on a general understanding of the particular area. Thus, it varies area by area and feature by feature. Larger d means a larger probability of largely different responses from different groups. In the paired t-test, the effect size is marked as d_z' that is supposed to be equal to $\frac{m_z}{\sigma_z}$, where $Z = X - Y$ and X and Y are a pair. When referencing the sample size table to find the value d in the reference book, it should be multiplied by $\sqrt{2}$. That is, $d = d_z'\sqrt{2}$.

For alpha, a typical value of 0.05 is used, which corresponds to a 95% confidence level. For a power, a moderate value of 0.8 is used. In the research, the moderate value 0.5 is used for an initial value. Thus, d is approximately 0.7. If the effect size attained during the pilot experiment is far smaller than the initial value, it may be adjusted to reduce the sample size.

With inputs, the needed sample size can be attained in the table of the reference book [55]. To achieve a 95% significance level and 80% power, at least 34 samples are needed.

4.4 Description of Input Files

Descriptions of the input files used for the four different experiments are mentioned in the following section 4.4.1 and 4.4.2. In the 4.4.2, the selection of input files is compared to the condition of Great Energy Predictor Shootout I.

4.4.1 Input Files for Pattern Observation

For the pattern observation, both synthetic and actual data are used. (See the Table 2 for details). The actual energy consumption data is from the Texas A&M University main campus is used. The data is provided by the utility office.

4.4.2 Input files for Determination of the SVM Implementation Settings, Investigation on the Influence of Different Independent Variables, and Comparative Study

Energy consumption data of buildings on Texas A&M University main campus is used. The data is provided by the utility office. The data is provided by the utility office. The energy use data is measured on an hourly basis. Daily data and monthly data are derived by summing these hourly records. In this research, monthly consumption is normalized by rearranging it daily. This effectively prevents the different number of days of the month from biasing the result. Similar to the Great Energy Predictor Shootout I, the data from the earlier period is used for training, and the data from the later period is used for testing. Whereas four months of data were used for training and two months of data were used testing in the Great Energy Predictor Shootout I, twice longer periods of data are used for both training and testing in this study. Specifically, one year of data, from 1 February 2018

to 31 January 2019 is used for training, and four months of data, from 1 February 2019 to 31 May 2019, are used for testing Apparent outliers during the training period are removed as needed. Testing data is used mostly for SVM models because it is a typical practice of checking the possibility of overfitting, which happens when a machine-learning algorithm reduces the error by aimlessly memorizing the data for the training period, not by learning the patterns of the data. However, there is a limit to this method. The actual energy use of buildings does not always neatly follow the pattern that they showed during the training, or baseline period. Hence, there is a chance that even if the machine-learning algorithm captures the patters of the data of the training period, it can give far worse results during the test period. Fortunately, except for the experiment for the comparative study with MLR, most of the SVM experiments conducted in the study use only one independent variable. This means that whether the SVM-trained models make sense or not can be checked quite accurately just by using eyes because what patterns the SVM algorithm generates is seen on a two-dimensional plot.

Dependent Variables: Energy Use

Two types of energy are used as a dependent variable of the experiment: chilled water (CHW) and heating hot water (HHW). Both CHW and HHW energy are calculated by the product of the water flow rate and the temperature difference between the inlet and outlet of the water system.

Most buildings on Texas A&M campus use chilled water for cooling and heating how water for heating. In general, each building has one CHW meter and one HHW meter for

its own, but some buildings share the same CHW and HHW meter and other buildings have more than one CHW or HHW meter per building.

Independent Variables: Weather data

Whereas the outdoor air dry-bulb temperature, absolute humidity ratio, wind speed, and horizontal insolation are given to the contestants Great Energy Predictor Shootout I, this study is conducted using smaller numbers of independent variables: outdoor air dry-bulb temperature (OAT) and dewpoint temperature (DPT). The hourly OAT and DPT data in the Easterwood Airport, College Station, TX, obtained from the National Center for Environmental Information is used. The daily and monthly OAT and DAT are generated by combining the hourly data. The OAE and OEE data are generated based on OAT and DPT. Table 4 is the list of the samples used.

Table 4 - Sample Table for Experiments

No.	TAMU BLDG#	Building Name	Bldg. Type[a]	Meter ID	Energy Type	Experiment Type			
						DIS[b]	IID[c]	CS[d]	
								MLR	CPLR
1	275	Liberal Arts and Arts & Humanities Building	O	7717	HHW		O		O
2	291	Rudder Residence Hall	D	2132	CHW		O	O	O
3	325-385	CE TTI Office & Lab Building	O	9123	CHW	O	O	O	O
				9124	HHW		O		O
4	353	Bright Aerospace Building	O	2746	CHW	O	O	O	O
				2757	HHW		O		O
5	358	Davis Football Player Development Center	G	7699	CHW	O	O	O	O
6	359-432	Architecture Building B&C	O	6419	CHW	O	O	O	O
				6423	HHW	O	O		O
7	376	Chemistry Building Addition	L	7119	HHW	O	O		O
8	383	Koldus Building	O	2863	CHW	O	O	O	O
9	384	Sanders Corps of Cadets Center	G	2583	CHW		O	O	O
				2587	HHW		O		O
10	386	Jack E. Brown Chemical Engineering Building	L	2250	CHW	O	O	O	O
				2254	HHW		O		O
11	387	Richardson Petroleum Engineering Building	L	5805	CHW		O	O	O

Table 4 Continued

No.	TAMU BLDG#	Building Name	Bldg. Type[a]	Meter ID	Energy Type	DIS[b]	IID[c]	CS[d] MLR	CS[d] CPLR
12	394	Underwood Residence Hall	O	2117	CHW		O	O	O
13	405	Lacy Hall - Dorm 6	D	7918	CHW		O	O	O
				7789	HHW		O		O
14	405-407-1402	Lacy Hall - Dorm 6, Harrell Hall and Leadership Learning Center	D	7722	CHW	O	O	O	O
				7723	HHW		O		O
				7919	HHW	O	O		O
15	408	Whitely Hall - Dorm 9	D	10036	HHW		O		O
16	420	Milner Hall	O	9146	HHW		O		O
17	425	Henderson Hall	O	2611	HHW		O		O
18	426-427-428	FHK Complex	D	2848	CHW	O	O	O	O
				2859	HHW	O	O		O
19	435	Harrington Education Center Office Tower	O	2796	HHW		O		O
20	436	Reed-McDonald Building	L	2423	HHW	O	O		O
21	443	Oceanography & Meteorology Building	O	6392	HHW	O	O		O
22	444	Peterson Building	L	2922	CHW		O	O	O
				6435	HHW		O		O
23	445	Teague Research Center	O	6415	HHW		O		O
24	449	Biological Sciences Building – West	L	3981	CHW		O	O	O
				3985	HHW	O	O		O
25	454	MSC	G	7584	CHW		O	O	O
				7585	HHW	O	O		O
26	468	Evans Library	G	3895	CHW	O	O	O	O
				3903	CHW		O	O	O
				3899	HHW	O	O		O
27	473	Williams Administration Building	O	7947	HHW		O		O
28	476	Francis Hall	O	8034	HHW	O	O		O
29	477	Anthropology Building	L	3668	HHW		O		O
30	478	Scoates Hall	L	7968	CHW	O	O	O	O
				7969	HHW	O	O		O
31	483	Thompson Hall	L	3891	HHW	O	O		O
32	484	Chemistry Building	L	7028	CHW	O	O	O	O
				7223	CHW	O	O	O	O
				7227	HHW	O	O		O
33	492	Civil Engineering Building	L	5950	CHW		O	O	O
34	496	Utilities & Energy Services Central Office	G	6933	HHW		O		O
35	508-1026	Veterinary Teaching Hospital	E	4166	CHW		O	O	O
36	513	Doherty Building	L	2898	CHW		O	O	O
37	517	DPC Annex	E	6563	CHW	O	O	O	O
38	520	Beutel Health Center	G	3933	CHW		O	O	O
39	682	Wisenbaker Engineering Research Center	L	3879	CHW	O	O	O	O
40	1085	Veterinary Small Animal Hospital	E	3656	CHW		O	O	O
				3660	HHW	O	O		O
41	1146	Biological Control Facility	L	5887	CHW		O	O	O
42	1156	Physical Plant Administration & Shops	G	7683	HHW		O		O
43	1184	Veterinary Anatomic Pathology	E	6999	HHW	O	O		O

Table 4 Continued

No.	TAMU BLDG#	Building Name	Bldg. Type[a]	Meter ID	Energy Type	Experiment Type			
						DIS[b]	IID[c]	CS[d]	
								MLR	CPLR
44	1501	Kleberg Center	L	2624	CHW		O	O	O
				2628	HHW		O		O
45	1502	Heep Center	L	2603	HHW	O	O		O
46	1503	Cater-Mattil Hall	L	8001	CHW		O	O	O
47	1504	Reynolds Medical Sciences Building	L	3989	CHW	O	O	O	O
				3993	HHW		O		O
48	1506	Horticulture-Forest Science Building	L	3967	CHW	O	O	O	O
49	1507	Biochemistry-Biophysics Building	L	3025	CHW		O	O	O
50	1508	Price Hobgood Ag. Engineering Research Lab	L	6005	CHW		O	O	O
51	1513	Borlaug Center for Southern Crop Improvement	L	5936	CHW		O	O	O
				5895	HHW		O		O
52	1530	Interdisciplinary Life Sciences Building	L	6290	CHW		O	O	O
53	1537	Wildlife Fisheries & Ecological Sciences Building	E	9983	CHW		O	O	O
				9984	HHW		O		O
54	1600	Gilchrist TTI Building	O	2649	CHW		O	O	O
55	1611	Engineering Research Building	E	8467	HHW		O		O
56	1800	General Services Complex	O	5472	HHW		O		O
57	1811	Vet Med Research Bldg Addition	L	6706	CHW		O	O	O
58	1904	Texas A&M Institute for Preclinical Studies A	E	6366	HHW		O		O
59	1911	Multi-Species Research Building	E	9129	CHW		O	O	O
				9133	HHW		O		O

a: Building Types: D - Dorms/Residence Hall , G – General/Dining/Sport, L – Lab/Classroom, O – Office/Classroom, E-Others
b: Determination of SVM Implementation Settings
c: Investigation on the Influence of Different Independent Variables
d: Comparative Study

CHAPTER V RESULTS ANALYSIS AND DISCUSSION

The experiment was conducted as described in Chapter 4. For SVM regression pattern observation, synthetic data have been used mostly. For the determination of the SVM implementation settings, investigation on the influence of different independent variables, and comparative study, the actual building energy data has been used. (See Table 4).

5.1 Determination of SVM Implementation Settings

The performance of the exhaustive grid search (EGS) method and the one-dimensional grid search with recommended input values (ODGS) method are compared in terms of forecasting capacity and running time. The comparison of forecasting capacity is carried out both for the training dataset and testing data set. This is needed due to the nature of machine learning that can result in overfitting. More specifically, even if both methods achieve the same level of accuracy for the given training data set, the way they fit the model may be quite different. If that is the case, they will achieve a different level of accuracy when they are tested with a testing data set. In the testing data part, the comparison of the running time is excluded because it is the training stage that takes up a significant portion of the time.

In addition to the issue of accuracy and efficiency, how each method has selected a set of hyperparameters is reviewed as well as the resultant number of support vectors. This information may reveal how the SVM performance is sensitive to the choice of hyperparameters. The number of samples is 32. The half of the sample, 16, are attained from CHW consumption data having OAT as input. The other half of the sample is

obtained from HHW consumption having OAT as input. (See TABLE 5) The results are

shown in the graphs below.
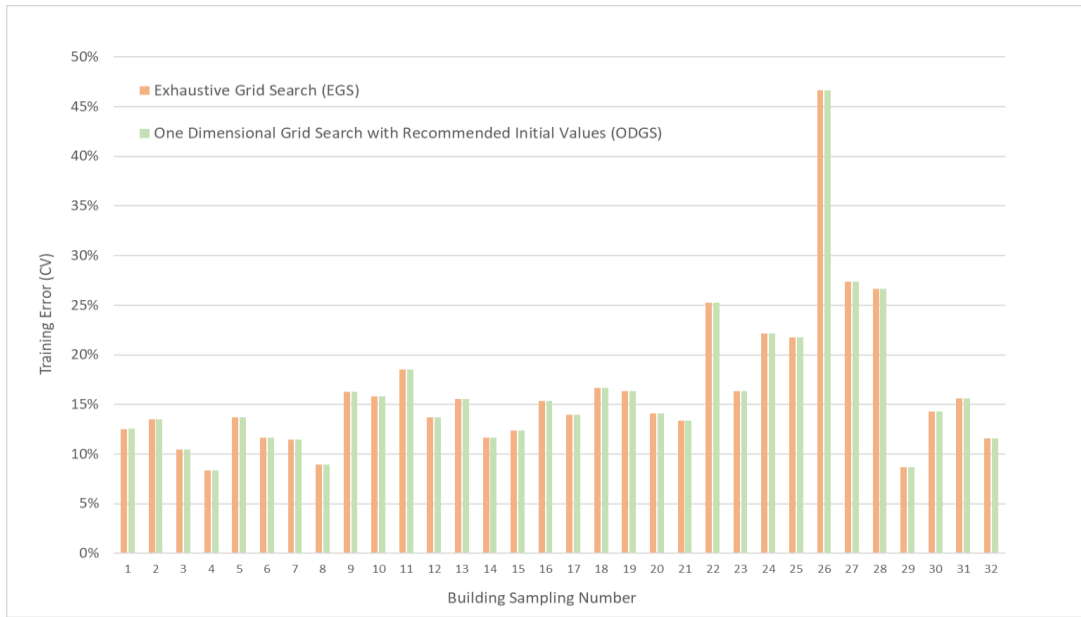
1) Training Data

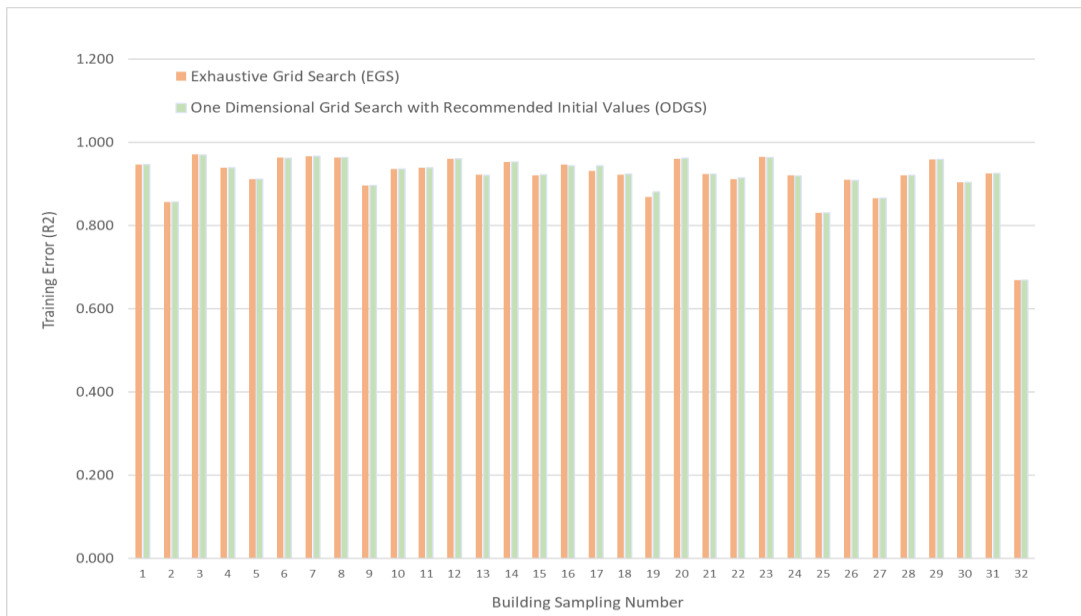
Figure 8 - Comparative Performance between EGS and ODGS (CV)

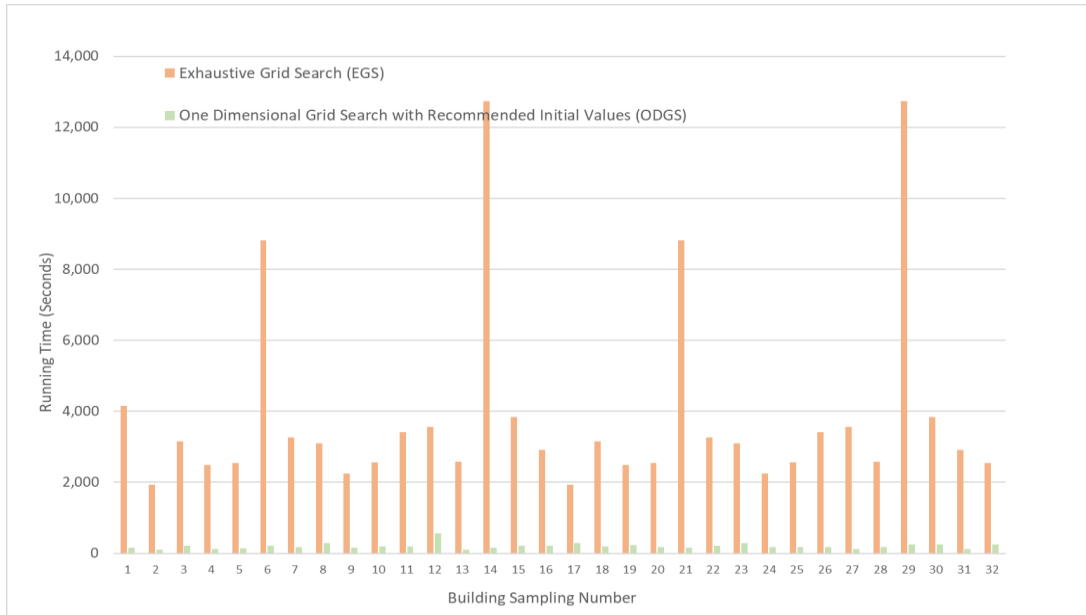Figure 9 - Comparative Performance between EGS and ODGS (R2)

59

Figure 10 - Comparative Performance between EGS and ODGS (running time, sec)
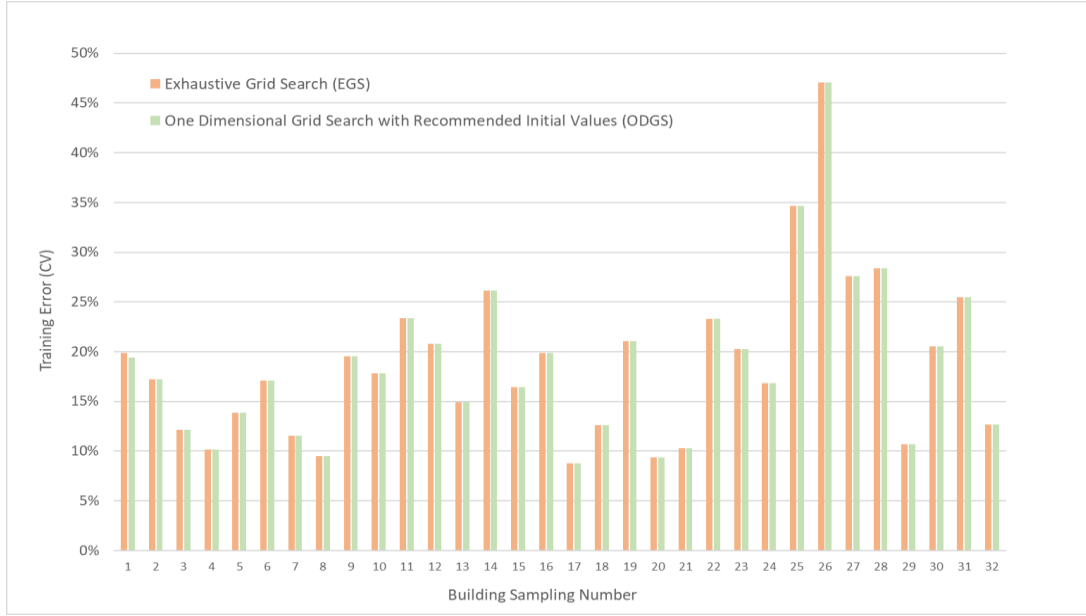
2) Testing Data

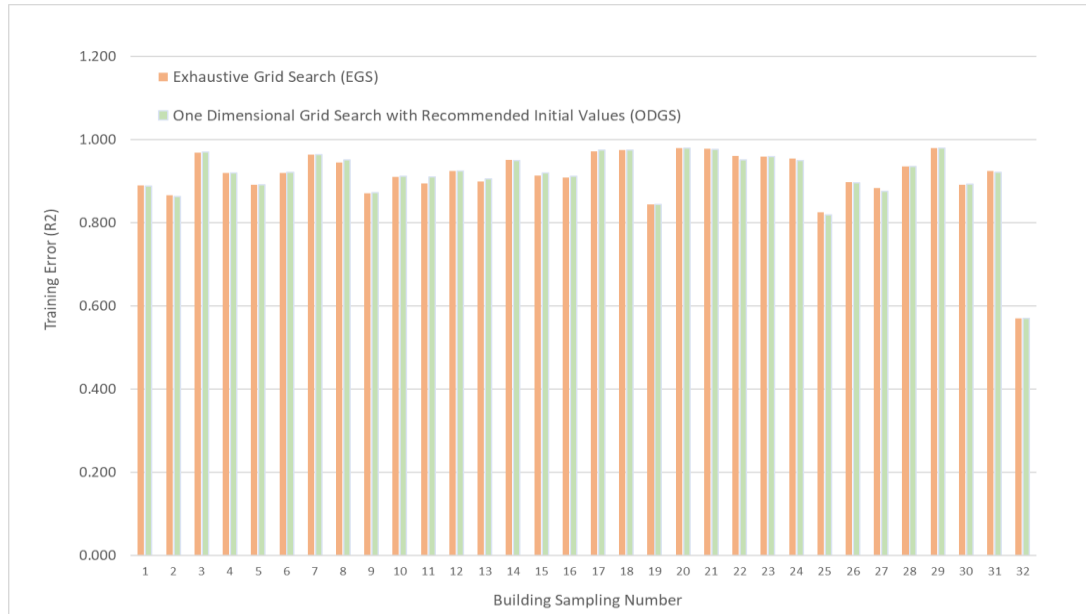Figure 11 - Comparative Performance between EGS and ODGS (CV)

Figure 12 - Comparative Performance between EGS and ODGS (R^2)

3) The result of hyperparameter selection

The combination of hyperparameters selected by the two tuning methods and the number of support vector for each case are listed in the Table 5 below.

Table 5 - Hyperparameters and the Number of Support Vector from EGS and ODGS

SP No	Bldg No.	Meter ID	Eng Type	EGS				ODGS				Same? *			
				ε [2^ε]	C [2^C]	[2^γ]	nSV	ε [2^ε]	C [2^C]	[2^γ]	nSV	ε	C	γ	nSV
1	325-385	9123	CHW	-4	14	-3	280	-4	-2	0	282	Y	N	N	N
2	353	2746	CHW	-2	12	-3	168	-2	5	-1	166	Y	N	N	N
3	358	7699	CHW	-3	13	1	141	-4	-2	-1	246	N	N	N	N
4	359-432	6419	CHW	-2	3	-2	110	-2	4	-2	109	Y	N	Y	N
5	383	2863	CHW	-2	0	-1	137	-2	2	-1	138	Y	N	Y	N
6	386	2250	CHW	-3	8	1	158	-3	5	1	154	Y	N	Y	N
7	405-407-1402	7722	CHW	-3	3	0	157	-3	5	0	158	Y	N	Y	N
8	426-427-428	2848	CHW	-3	11	0	145	-3	13	-3	147	Y	N	N	N
9	468	3895	CHW	-2	2	-2	159	-4	-1	-1	298	N	N	N	N
10	478	7968	CHW	-9	-2	-1	359	-4	-1	-1	284	N	N	Y	N

61

Table 5 Continued

SP No	Bldg No.	Meter ID	Eng Type	EGS				ODGS				Same? *			
				$\varepsilon\ [2^\varepsilon]$	$C\ [2^C]$	$[2^\gamma]$	nSV	$\varepsilon\ [2^\varepsilon]$	$C\ [2^C]$	$[2^\gamma]$	nSV	ε	C	γ	nSV
11	484	7028	CHW	-4	-1	2	230	-4	3	0	231	Y	N	N	N
12	484	7223	CHW	-4	-1	2	221	-4	14	0	218	Y	N	N	N
13	517	6563	CHW	-2	14	1	136	-2	8	-1	133	Y	N	N	N
14	682	3879	CHW	-3	4	-2	198	-3	0	-2	202	Y	N	Y	N
15	1504	3989	CHW	-4	1	1	293	-4	11	-2	291	Y	N	N	N
16	1506	3967	CHW	-4	13	1	269	-6	0	-1	342	N	N	N	N
17	359-432	6423	HHW	-4	15	-7	237	-7	6	-3	335	N	N	N	N
18	376	7119	HHW	-2	7	-1	183	-2	4	0	181	Y	N	N	N
19	405-407-1402	7919	HHW	-2	15	-5	151	-7	8	-3	346	N	N	N	N
20	426-427-428	2859	HHW	-7	2	-4	339	-3	2	-4	116	N	Y	Y	N
21	436	2423	HHW	-2	15	-7	103	-2	3	-2	100	Y	N	N	N
22	443	6392	HHW	-3	2	-3	213	-3	0	-2	206	Y	N	N	N
23	449	3985	HHW	-14	1	0	306	-8	2	-5	297	N	N	N	N
24	454	7585	HHW	-9	15	-5	351	-3	5	-1	139	N	N	N	N
25	468	3899	HHW	-2	5	-2	191	-2	2	-2	196	Y	N	Y	N
26	476	8034	HHW	-4	15	-5	189	-3	8	-4	150	N	N	N	N
27	478	7969	HHW	-15	8	-5	355	-1	4	-2	58	N	N	N	N
28	483	3891	HHW	-7	12	-6	297	-3	5	-1	133	N	N	N	N
29	484	7227	HHW	-11	9	-2	358	-3	4	-3	151	N	N	N	N
30	1085	3660	HHW	-4	15	-9	247	-5	3	-4	297	N	N	N	N
31	1184	6999	HHW	-2	5	0	86	-2	9	-1	86	Y	N	N	Y
32	1502	2603	HHW	-2	15	-11	219	-2	12	-9	220	Y	N	N	N

*Y: Yes, N: No

Conclusion of the comparison

The ODGS has shown nearly the same performance as EGS in both training data set and testing data set for significantly less time. This result implies two significant clues. First, ODGS would be much more useful at least for the condition in which this research is. Second, the performance of SVM for this research may not be much sensitive to the selection of hyper-parameters compared to other applications. This is different from a general understanding of SVM, the performance of which is highly sensitive to the combination of a set of hyperparameters. This difference may be attributed to the fact that

the regression condition for this research is quite simple. Only a single attribute is used as an input for one regression. Another possible reason could be the number of training data set, which is at most 365. This number may not be said small, but also cannot termed substantial.

Despite the result, it may be too early to state that the ODGS is always a better choice because the case is limited and the space between grids is also limited to the scale of log base 2. However, the result would be strong enough to use ODGS as a default option at least for the application of this research.

5.2 SVM Regression Pattern Observation

The plots in Appendix C show how SVM confronts different patterns of data, such as ideal linear, noisy linear, piecewise linear, scattering, and so on. For some portion of data, two different data frequencies, daily and monthly, are considered separately. This section reviews the SVM regression pattern observation results.

5.2.1 Linear Patterns

Two different kernel functions, linear and RBF, are used for comparative purposes. Even when the data set is linear, the performance of RBF kernel SVM is close to that of linear SVM. When the given data is perfectly linear, the regression plot from RBF kernel SVM is also almost linear to that extent that it is nearly impossible to distinguish it from the plot of linear SVM. When noise is imposed, the linearity of RBF kernel SVM is weakened, but it still maintains its linear form fairly well with a large noise. Thus, unless there is a specific

reason to constrain the SVM regression plot to a line, using the RBM kernel SVM for linear data would not be a problem.

How RBF kernel SVM generates a line

The regression plot made from RBF kernel SVM shows a slightly bumped shape even on the part where the given data is perfectly linear. This implies how RBF kernel SVM makes a line. This phenomenon can be understood by examining the mathematical representation of RBF kernel SVM regression.

$$f(x) = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)e^{-\gamma\|x-x_i\|^2} + b \qquad \textbf{(EQ. 25)}$$

N is a total number of data set used for SVM training, i represents the index for the training data set, and thus, x_i represents the ith training dataset. x represents a new data set. If the ith data set x_i is not a support vector, the corresponding coefficient parts $(\alpha_i - \alpha_i^*)$ becomes zero. Thus, the number of remaining exponential function, which is symmetrical about x_i, is the same as that of the support vector. By examining the equation, it is clear that the exponential part generates the bump in the regression plot. This can be intuitively understood by using a graph with simplified parameter settings for the RBF kernel SVM function. For instance, let us assume that there are three support vectors and that the number of independent variables for the dataset is just one, that x and x_i can remain as scalar with $x_1, x_2,$ and x_3 being 1, 3, and 6, respectively. If the $(\alpha_i - \alpha_i^*)$ corresponding to $x_1,$ $x_2,$ and x_3 are one, b is zero, and γ is one, the regression model takes the following mathematical representation, and the corresponding plot is as follows.

$$f(x) = e^{-(x-1)^2} + e^{-(x-3)^2} + e^{-(x-6)^2}$$
(EQ. 26)
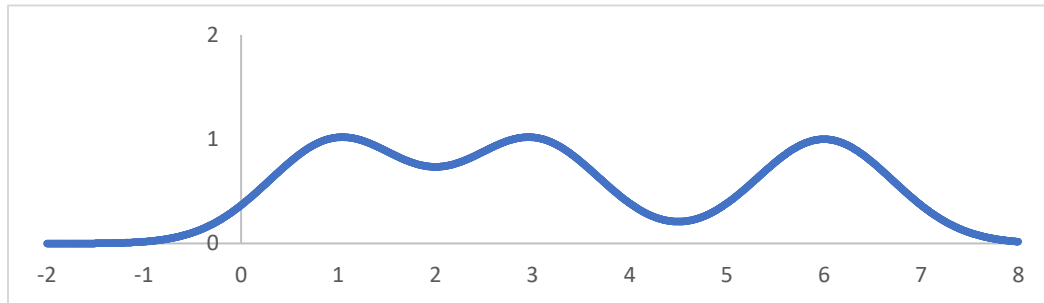
Figure 13 - Graph of a Combination of Exponential Functions 1

As stated above, the number of bumps corresponds to that of support vectors. In addition, the center of each bump corresponds to the location of each support vector. Upon observing the plot and mathematical representation, it can be deduced how RBF kernel SVM regression forms linear or flat patterns. The RBF kernel SVM regression is formed by a group of spikes, which is formed by the corresponding support vectors. To make a flat line out of spikes, the logical approach is to increase the number of spike for a given data range. That is equivalent to increase the number of support vector. For example, increasing the number of support vectors from three to six and having them equally spaced by one while keeping other conditions same, the equation takes the following mathematical form, and the corresponding plot is as follows.

$$f(x) = e^{-(x-1)^2} + e^{-(x-2)^2} + e^{-(x-3)^2} + e^{-(x-4)^2} + e^{-(x-5)^2} + e^{-(x-6)^2} \quad \textbf{(EQ. 27)}$$
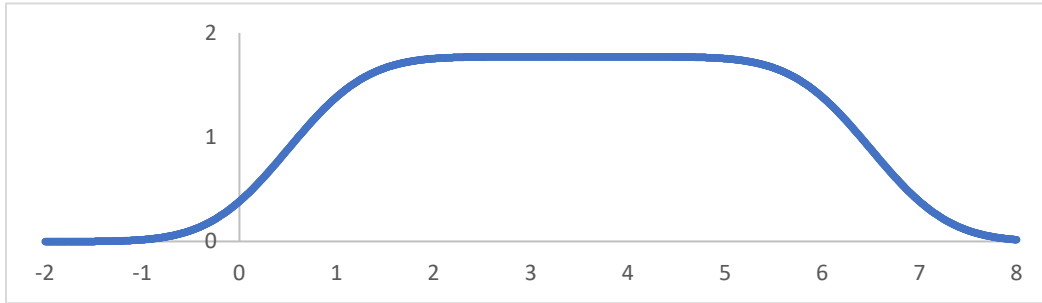
Figure 14 - Graph of a Combination of Exponential Functions 2

However, there is another aspect to consider. That is computational efficiency. More support vectors mean more computational cost because it makes the mathematical representation of SVM regression longer. Given a linear-patterned data set, RBF kernel SVM takes much more support vector than linear SVM does. This is confirmed in Figure C1 and Figure C15 where the number of support vector is displayed. To perform what appeared to be the same task, RBF kernel SVM needed 350 support vectors whereas the linear SVM used just two support vectors.

5.2.2 Piecewise Linear Patterns

When the data is ideally clean or have just a little noise, SVM regression models show two distinct pieces. As the noise is larger, it becomes closer to a curve. Compared to the case with daily data, when performance from monthly input data is inconsistent and more affected by noise. This characteristic may or may not be advantageous for building energy baseline modeling. If a building is certainly supposed to take a piecewise pattern, this will be disadvantageous, generating a false impression of how the building consumes energy

regardless of its reported performance. On the contrary, if the energy-use pattern of

buildings does not necessarily need to be constrained by a connection of lines and if they

follow or are close to a curved one, this trait would be useful, successfully visualizing a

pattern that cannot be realized by the piecewise linear regression models.

5.2.3 Daily vs Monthly

SVM from daily data (365 data points) shows much more robust results than monthly data

(12 data points), not easily distracted by noise. This is because more input means more

hints about what a regression plot is supposed to look like. Even if a data point fall under a

category considered a noise range, it may still help to prevent the SVM algorithm from

making worse prediction unless the data is severely outrageous. The 365 points of a single

attribute input appear to be sufficient for SVM to show a stable performance unless the

noise is severe whereas 12 points of monthly data could lose its way even with a moderate

noise level.

5.2.4 Scattered Patterns

When scattered input is given to SVM, it is more likely to lose its way and make a

complex shape than it is under other cases. This demonstrates that SVM would not be a

good option if the portion of scattering is large enough to blur the distinctness of the data

pattern.

5.2.5 Break Point patterns

Data with breakpoints have some scattering potions in the middle part. Despite its

weakness to scattering input stated above, it made sense of the given data and generated a

plot that smoothly goes through the middle of the scattering. From a different perspective,

it demonstrates that the breakpoint pattern is distinct enough for the SVM algorithm not to

lose its way despite the scattering.

5.2.6 Scheduled Patterns

When the given data set is schedule-dependent, but SVM does not have a schedule as an

input attribute, it demonstrates much lower performance than the case with the schedule

input, or lose its way. Since SVM is supervised learning, without guidance in the form of

input attributes, it has no way to catch the schedule by itself.

5.2.7 Hourly Data

Hourly data has more data points, 8760, than daily or monthly data, and has thick patterns.

However, SVM has no problem making sense of the pattern and generating a meaningful

plot. This is in line with the previous observation that SVM from daily data shows more

robust performance confronting noise than the monthly counterpart. This demonstrates that

when the number of data points increases, SVM is less like to lose its way.

5.3 Investigation on the Influence of Different Independent Variables

Experiments have been conducted to understand the influence of different inputs on SVM performance. Daily data and monthly data were considered separately. According to the typical format of machine-learning performance indication, both training period performance and testing period performance were recorded. The significance was checked with paired t-test in terms of both CV and R^2

5.3.1 Daily

The Table 6 and Table 7 are the result of attribute study for daily data. Relevant SVM regression plots are listed in the Appendix. D.1. As the regressor of CHW changed from OAT through OAE and OEE to OAT+DPT, the performance slightly increased. The CHW-OAT data set resulted in the most varying shape of the SVM regression plot whereas the other three data set resulted in more consistent and simpler SVM regression forms. This variety of SVM regression plots from the CHW-OAT dataset would be mostly caused by humidity, another important factor of CHW use. The simpler and consistent forms from CHW-OAE and CHW-OEE compared to the CHW-OAT dataset would be an indication that they can be a good predictor of CHW use by themselves. Although the mean difference in performance between CHW-OAE and CHW-OEE was just 0.5% in CV and 0.004 in R^2, the plot from CHW-OEE appears cleaner in general. In addition, whereas many plots from CHW-OAE are closer to a smoother version of the piecewise linear regression model, many of those from CHW-OEE are close to a linear pattern. That would the reason why the small difference was shown to be significant by the paired t-test. Although most CHW-OAT data sets became more stable in the corresponding CHW-OAE

and CHW-OEE plots, there are two exceptional cases, which are sample No. 6 and No. 7. Even when the predictors changed to OAE or OEE, the data patterns were not stabilized, and SVM also generated complex forms. The reason is unclear due to the limited information available. It may or may not be improved with modified SVM implementation settings. At least it teaches that switching the predictor from OAT to OAE or OEE does not always lead to a more stable data distribution although it does in most cases.

When it comes to OAT+DPT although it demonstrated the highest performance, its feasibility would be limited for the two reasons. First, the shape of the model is difficult to understand intuitively because it takes two independent variables. This may cause the implementer to hesitate to choose despite the highest performance. Second, humidity is not free from the chance of collinearity with temperature, thus likely to give optimistically biased information.

Although the SVM regression from HHW-OAT data demonstrated less performance than CHW-OAT in terms of CV and R2, they appear to take simpler forms than CHW-OAT. For instance, whereas many SVM plots from CHW-OAT have abruptly changed slope in the extreme range of OAT, it is often not the case with HHW-OAT plots. Compared to CHW-OAT, SVM generated more clean curved models. As stated above, such a complex form from CHW-OAT data can be attributed to the influence of humidity, which does not matter with the use of HHW. Hence, comparing CHW-OEE with HHW-OAT in this regard, there is still a noticeable difference. Whereas linear SVM plots are dominant in the CHW-OEE plots, curved plots are dominant in the HHW-OAT plots.

Table 6 - The Result of SVM Regression with Different Input Attributes (Daily)

Energy	Input Attributes	Training Period Performance		Testing Period Performance	
		CV	R2	CV	R2
CHW	OAT	12.2%	0.938	18.2%	0.871
CHW	OAE	11.2%	0.941	16.9%	0.880
CHW	OEE	10.7%	0.945	16.2%	0.889
CHW	OAT + DPT	9.32%	0.958	15.34%	0.896
HHW	OAT	19.52%	0.903	25.89%	0.884

Table 7 - The Comparison among of SVM Regressions with a Single Input Attribute (Daily)

	Energy	Training (Baseline) Period Performance				Paired t-test	Testing Period Performance				Paired t-test
		CV	R2	CV	R2	-	CV	R2	CV	R2	-
1	CHW	OAT		OAE		95% Confidence?	OAT		OAE		95% Confidence?
		12.2%	0.938	11.2%	0.941	CV	18.2%	0.871	16.9%	0.880	None
2	CHW	OAT		OEE		95% Confidence?	OAT		OEE		95% Confidence?
		12.2%	0.938	10.7%	0.945	CV	18.2%	0.871	16.2%	0.889	CV, R2
3	CHW	OAE		OEE		95% Confidence?	OAE		OEE		95% Confidence?
		11.2%	0.941	10.7%	0.945	CV, R2	16.9%	0.880	16.2%	0.889	CV, R2

5.3.2 Monthly

The Table 8 and Table 9 are the result of attribute study for monthly data. Relevant SVM

regression plots are listed in the Appendix. D.2. The performance of regressing the use of

CHW increased as the regressor changed from OAT to OAE and from OAT to OEE.

However, there is no significant difference in performance between the use of OAE and

that of OEE as a regressor. Unlike the case with daily data, the performance with

OAT+DPT is the worst. By observing the plots, it is clear that, contrary to the SVM plot

from daily data, there are no noticeable differences among the plots with different

regressors. Most of them take loosely linear forms and a few of them appear to lose their

way slightly. There may be two reasons for this. First, the pattern of original data is supposed to be simple because there is a chance that the impact of weekdays/weekends scheduled or other unidentified, yet existing schedule-dependent patterns might have been averaged out, causing the corresponding the pattern of monthly consumption to be simpler in terms of the regressors. Second, the number of data is not large enough for the SVM algorithm to effectively catch the actual pattern. This issue continues to be handled in the next part, comparative study.

Table 8 - The Result of SVM Regression with Different Input Attributes (Monthly)

Energy	Input Attributes	Training Period Performance		Testing Period Performance	
		CV	R2	CV	R2
CHW	OAT	8.0%	0.973	14.2%	0.971
CHW	OAE	5.4%	0.987	12.8%	0.978
CHW	OEE	5.4%	0.985	12.4%	0.977
CHW	OAT + DPT	5.27%	0.983	13.40%	0.966
HHW	OAT	19.52%	0.903	25.89%	0.884

Table 9 - The Comparison among of SVM Regressions with a Single Input Attribute (Monthly)

	Energy	Training (Baseline) Period Performance				Paired t-test	Testing Period Performance				Paired t-test
		CV	R2	CV	R2	-	CV	R2	CV	R2	-
1	CHW	OAT		OAE		95% Confidence?	OAT		OAE		95% Confidence?
		8.0%	0.973	5.4%	0.987	CV, R2	14.2%	0.971	12.8%	0.978	None
2	CHW	OAT		OEE		95% Confidence?	OAT		OEE		95% Confidence?
		8.0%	0.973	5.4%	0.985	CV, R2	14.2%	0.971	12.4%	0.977	CV
3	CHW	OAE		OEE		95% Confidence?	OAE		OEE		95% Confidence?
		5.4%	0.987	5.4%	0.985	None	12.8%	0.978	12.4%	0.977	None

5.4 Comparative Study

The performance of SVM was compared to that of multiple linear regression and change point linear regression. Daily data and monthly data were considered separately. As in the same way as chapter 5.4, both training period performance and testing period performance were recorded. The detailed record of performance for each building and meter is listed below. The significance was checked with paired t-test in terms of both CV and R^2

5.4.1 Multiple Linear Regression Model

With daily data, SVM showed better performance in almost all sample buildings. With monthly data, although SVM still showed better performance, the difference in performance reduced and some sample buildings did not demonstrate better results with SVM. The difference in performance is thought to be caused by the constraint MLR has. Although MLR has better performance than simple linear regression, it is still limited to a linear expression. Because of the decreased performance in the monthly application, it would be reasonable to state that monthly data has less of a problem being described as linear.

5.4.1.1 Daily Data

The Table 10, Figure 15 and Figure 16 are the results of the attribute study for daily data. Relevant SVM regression plots are listed in the Appendix. D.1.

Table 10 - Mean Difference in Performance between SVM and MLR (Daily)

Energy	Regressor	Training (Baseline) Period Performance					Testing Period Performance				
		SVM		MLR		95% Confidence?	SVM		MLR		95% Confidence?
		CV	R2	CV	R2		CV	R2	CV	R2	
CHW	OAT + DPT	9.3%	0.958	16.1%	0.896	CV, R2	15.3%	0.896	24.3%	0.797	CV, R2
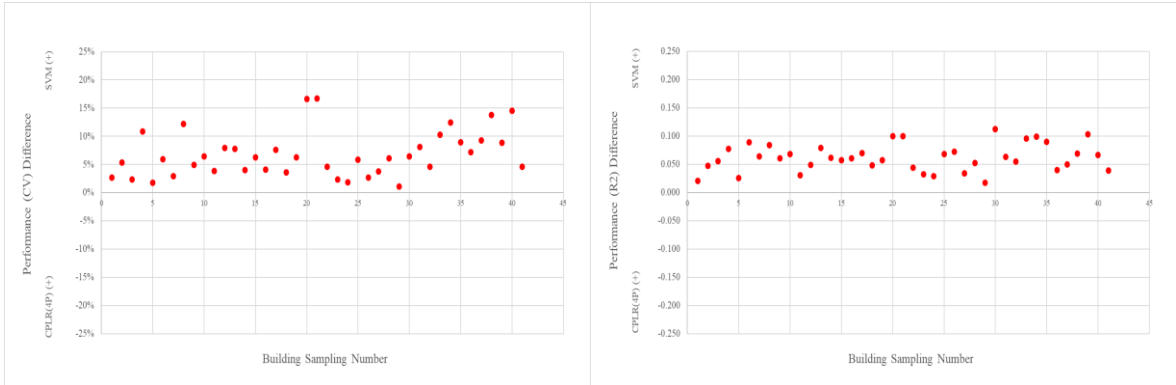
Figure 15 - SVM vs MLR for Training (Baseline) Period Performance (Daily)
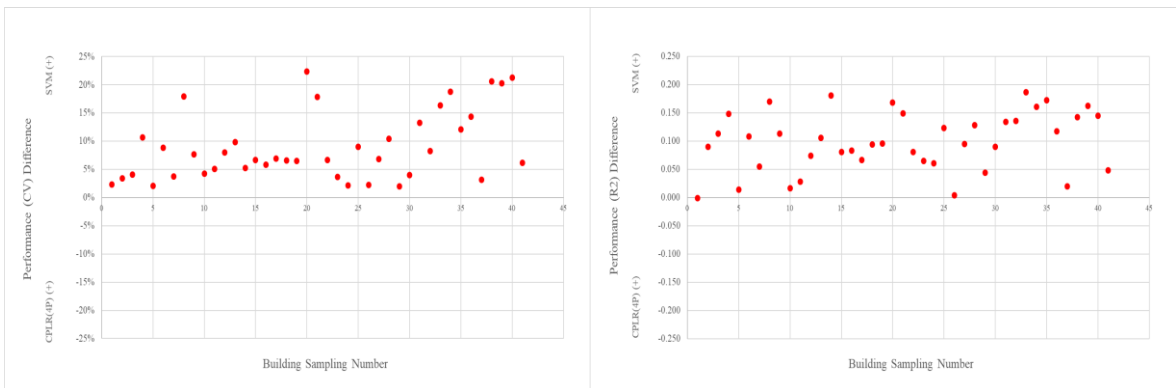
Figure 16 - SVM vs MLR for Testing Period Performance (Daily)

74

Results Based on Building Type

The difference in performance is arranged in terms of building types.

Table 11 - Difference in Performance between SVM and MLR for the Training period (Daily)

	Performance Difference*	
	R2	CV
Dorms/Res Hall	0.050	5.74%
General/Dining/Sport	0.066	5.28%
Lab/Classroom	0.066	8.16%
Office/Classroom	0.064	4.86%
Others	0.052	6.56%

*SVM showing higher performance is indicated positive.

Table 12 - Difference in Performance between SVM and MLR for the Testing Period (Daily)

	Performance Difference*	
	R2	CV
Dorms/Res Hall	0.014	1.54%
General/Dining/Sport	0.010	1.07%
Lab/Classroom	0.020	3.34%
Office/Classroom	0.019	2.51%
Others	0.017	3.35%

*SVM showing higher performance is indicated positive.

5.4.1.2 Monthly Data

The Table 13, Figure 17, and Figure 18 are the result of attribute study for daily data.

Relevant SVM regression plots are listed in the Appendix. D.2.

Table 13 - Mean Difference in Performance between SVM and MLR (Monthly)

		Training (Baseline) Period Performance					Testing Period Performance				
		SVM		MLR		95% Confidence?	SVM		MLR		95% Confidence?
Energy	Regressor	CV	R2	CV	R2		CV	R2	CV	R2	
CHW	OAT + DPT	5.3%	0.983	7.3%	0.970	CV, R2	13.4%	0.966	15.8%	0.950	CV
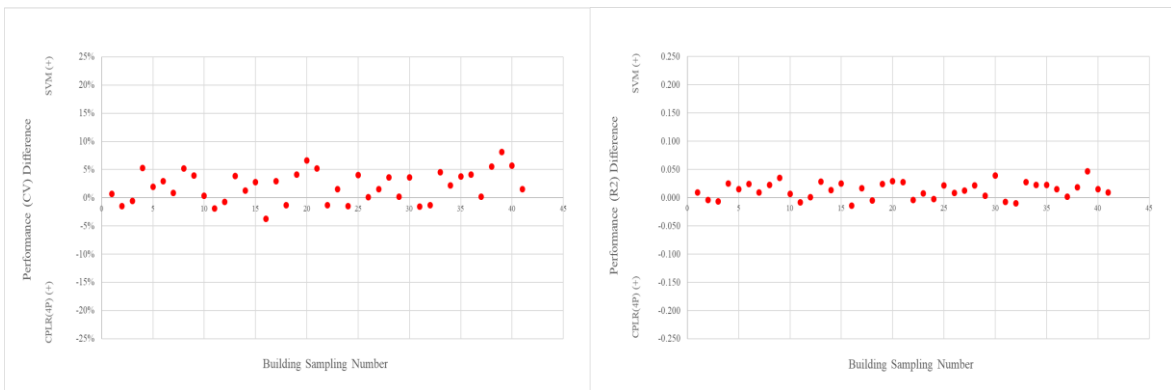
Figure 17 - SVM vs MLR for Training (Baseline) Period Performance (Monthly)
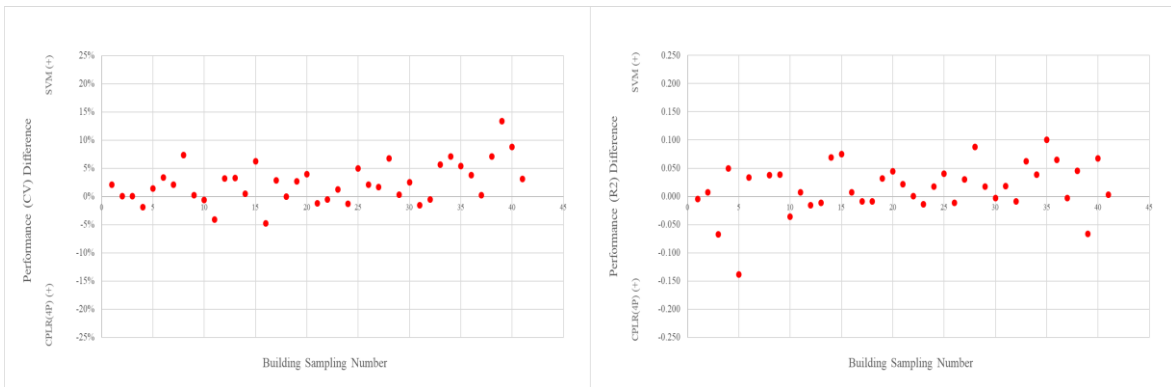
Figure 18 - SVM vs MLR for Testing Period Performance (Monthly)

Results Based on Building Type

The difference in performance is arranged in terms of building types.

Table 14 - Difference in Performance between SVM and MLR for the Training Period (Daily)

	Performance Difference*	
	R2	CV
Dorms/Res Hall	0.073	4.50%
General/Dining/Sport	0.092	6.78%
Lab/Classroom	0.108	9.25%
Office/Classroom	0.106	14.31%
Others	0.092	8.50%

*SVM showing higher performance is indicated positive.

Table 15 - Difference in Performance between SVM and MLR for the Testing Period (Daily)

	Performance Difference*	
	R2	CV
Dorms/Res Hall	-0.026	-0.38%
General/Dining/Sport	0.071	1.23%
Lab/Classroom	0.031	2.64%
Office/Classroom	-0.030	2.76%
Others	0.045	3.45%

*SVM showing higher performance is indicated positive.

5.4.2 Change Point Linear Regression Model

Daily

Table 16 is the summary of the comparative study with CPLR (4P) regression with daily data.

Table 16 - Summary of SVM vs CPLR Performance for Daily data

		Training (Baseline) Period Performance					Testing Period Performance				
		SVM		CPLR (4P)		95% Confidence?	SVM		CPLR (4P)		95% Confidence?
Energy	Regressor	CV	R2	CV	R2		CV	R2	CV	R2	
CHW	OAT	12.2%	0.938	13.1%	0.930	CV, R2	18.2%	0.871	17.7%	0.872	CV
CHW	OAE	11.2%	0.941	11.4%	0.940	CV, R2	16.9%	0.880	17.2%	0.878	CV
CHW	OEE	10.7%	0.945	10.9%	0.944	CV, R2	16.2%	0.889	16.3%	0.889	None
HHW	OAT	19.5%	0.903	20.1%	0.897	CV, R2	25.9%	0.884	26.6%	0.879	CV, R2

SVM shows slightly better performance in all the experiments for training (or baseline) period. This is somewhat expected results because SVM is an algorithm, which is designed in a way to fit the given data without any constraint on what the regression plot is supposed to look like as opposed to CPLR being constrained to a connection of two lines. Examining each case closely demonstrates that the SVM regression plots that have a form that is difficult to be described by CPLR tend to show relatively larger performance differences than others do. Such a form includes curved patterns and ones having an abrupt change in slope in the extreme range of regressor.

What should be noted with CHW is that, as the regressor changes from OAT through OAE to OEE, the difference in performance during the baseline period is reduced. In other

words, as the regressor changes in such an order, the CHW patterns becomes closer to the shape that is effectively described by CPLR.

In the testing data, CPLR demonstrates a better result when it fits CHW with OAT although the performance assessed by R^2 has not passed the paired t-test. SVM showed marginally better performance with the CHW-OAE testing data set. This is not much different from the result form the training data. Thus, it can be guessed that the CHW pattern described by OAT is more likely to change than CHW described by OAE. In other words, it indicates the limit of using OAT as a regressor for predicting CHW.

In the cases of OEE, although SVM shows marginally better performance in the training (baseline) period, there is no significant difference between the two when they were assessed with the training data set. Neither CV nor R^2 has passed the paired t-test. This will be more intuitively understandable as the plots are generated by both of them (See Appendix. D.1.3). Many of CHW-OEE data take clean linear patterns, and a significant portion of SVM and CPLR overlaps to the extent that it would be difficult to distinguish one from the other. This demonstrates how powerful OEE is as a regressor of CHW. HHW-OAT is the only case where SVM showed better performance in terms of CV and R^2 in both training period performance and testing period performance. As stated in Chapter 5.4, SVM plots from HHW-OAT data are characterized by dominant portions of the clean curve. Judging from the even better performance in the testing period, it may not be by chance that the original data during the training period appears to be a curve. Rather, it would not be unreasonable to assume that the curve generated by SVM actually may be closer to the pattern those buildings that actually consume HHW than a combination of two parts of the line.

Monthly

Table 17 summarizes the comparative study with CPLR (4P) regression with monthly data.

Table 17 - Summary of SVM vs CPLR Performance for Monthly Data

| | | Training (Baseline) Period Performance | | | | Testing Period Performance | | | | |
| | | SVM | | CPLR (4P) | | 95% Confidence? | SVM | | CPLR (4P) | | 95% Confidence? |
Energy	Regressor	CV	R2	CV	R2		CV	R2	CV	R2	
CHW	OAT	8.02%	0.973	7.63%	0.977	CV	14.2%	0.971	12.4%	0.977	CV
CHW	OAE	5.38%	0.986	5.34%	0.987	None	12.8%	0.978	12.3%	0.979	None
CHW	OEE	5.44%	0.985	5.27%	0.987	R2	12.4%	0.977	12.2%	0.982	R2
HHW	OAT	8.4%	0.970	8.8%	0.970	None	22.0%	0.918	20.5%	0.935	CV, R2

In almost all experiments, SVM showed worse performance than CPLR. More than half of the performance was significant, according to the paired *t*-test. This result is contrary to the case with daily data where SVM demonstrated better results in most cases although the degree of difference is marginal. Two things can be determined from this result. First, the number of input data for monthly baseline modeling, 12, is not enough for SVM to make sense of the pattern. This claim may be supported by the plot in the Appendix.D.2 where many SVM plots resemble a loosely drawn straight line when the corresponding CPLR generated two lines that have two distinctly different slopes. Second, cross-validation has a robust performance of preventing aimless fitting. Had the SVM algorithm and implementation settings been arranged in such a way that it focused only on minimizing errors aimlessly without concern for catching patterns, the training (baseline) period performance of SVM would have been higher than CPR in all cases. However, the opposite was observed. Although the reported performance is worse than CPLR in all cases except for HHW-OAT, it implies that cross-validation worked as intended.

5.4.2.1 Daily Data

The Table 18 through Table 25, and Figure 19 through Figure 28 display the result of comparative study for daily data. Relevant SVM regression and CPLR(4P) are listed in the Appendix. D.1.

Table 18 - Mean Difference in Performance between SVM and CPLR (4P) (CHW-OAT) (Daily)

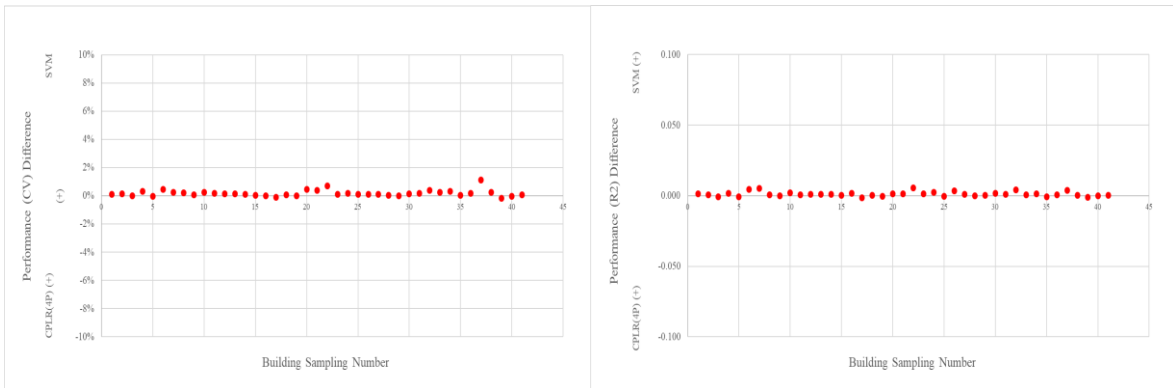| | | Training (Baseline) Period Performance | | | | | Testing Period Performance | | | | |
		SVM		CPLR (4P)		95% Confidence?	SVM		CPLR (4P)		95% Confidence?
Energy	Regressor	CV	R2	CV	R2		CV	R2	CV	R2	
CHW	OAT	12.2%	0.938	13.1%	0.930	CV, R2	18.2%	0.871	17.7%	0.872	CV

Figure 19 - SVM vs CPLR (4P) (CHW-OAT) for Training (Baseline) Period Performance (Daily)
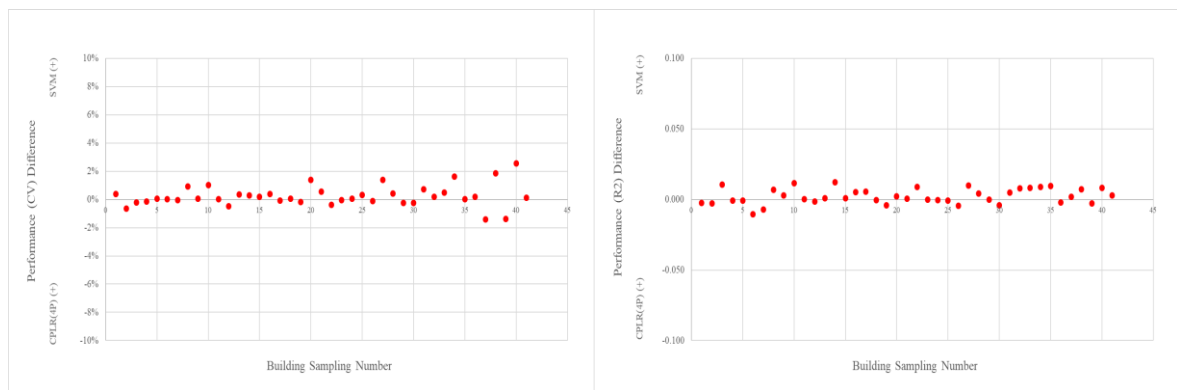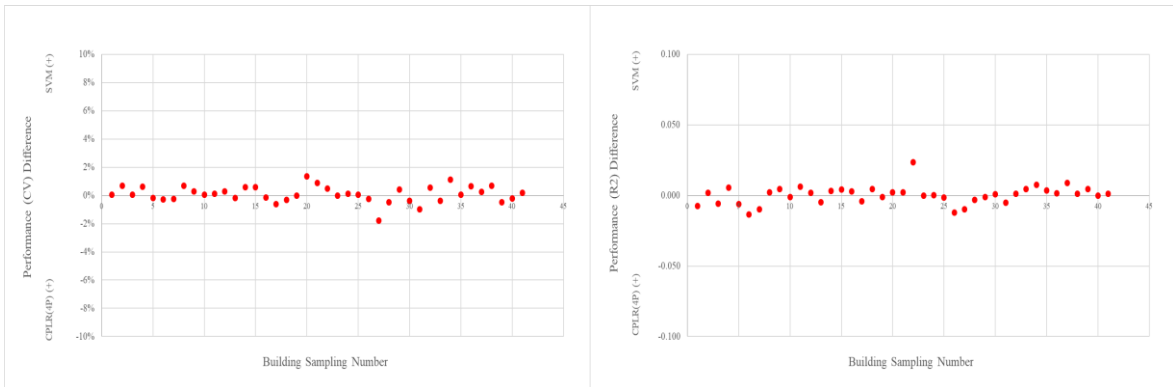

Figure 20 - SVM vs CPLR (4P) (CHW-OAT) for Testing Period Performance (Daily)

81

Table 19 - Difference in Performance between SVM and CPLR (4P) (CHW-OAT) for the Training Period (Daily)

Sampl. No.	Building No.	Meter ID	No. of data	SVM		CPLR (4P)		Difference		Remark (on SVM Model)
				R2	CV	R2	CV	R2	CV	
1	291	2132	338	0.963	9.4%	0.961	9.7%	0.002	0.3%	
2	325-385	9123	355	0.946	12.5%	0.942	13.1%	0.004	0.5%	
3	353	2746	317	0.857	13.5%	0.854	13.5%	0.003	0.0%	
4	358	7699	362	0.970	10.4%	0.965	11.4%	0.005	0.9%	
5	359-432	6419	362	0.939	8.3%	0.934	8.7%	0.005	0.3%	
6	383	2863	362	0.912	13.7%	0.909	14.1%	0.003	0.4%	
7	384	2583	361	0.926	8.0%	0.924	8.1%	0.002	0.1%	
8	386	2250	357	0.963	11.6%	0.953	13.2%	0.010	1.6%	wavering SVM regression model
9	387	5805	346	0.916	14.0%	0.909	14.6%	0.006	0.6%	
10	394	2117	358	0.924	15.4%	0.924	15.1%	0.000	-0.3%	
11	405	7918	360	0.954	11.4%	0.948	12.1%	0.006	0.6%	
12	405-407-1402	7722	358	0.967	11.5%	0.962	12.2%	0.004	0.7%	
13	426-427-428	2848	358	0.963	9.0%	0.962	9.1%	0.001	0.1%	
14	444	2922	360	0.953	7.3%	0.946	7.8%	0.006	0.5%	
15	449	3981	310	0.953	9.5%	0.941	10.7%	0.012	1.2%	decrease in slope in the high temp.
16	454	7584	362	0.871	17.6%	0.861	18.1%	0.010	0.5%	
17	468	3895	362	0.896	16.3%	0.887	16.9%	0.008	0.6%	
18	468	3903	353	0.967	8.9%	0.955	10.3%	0.012	1.4%	
19	478	7968	362	0.936	15.8%	0.932	16.4%	0.004	0.6%	
20	484	7028	362	0.939	18.5%	0.921	21.3%	0.018	2.8%	decrease in slope in the high temp.
21	484	7223	362	0.960	13.7%	0.948	15.7%	0.012	2.0%	decrease in slope in the high temp.
22	492	5950	357	0.922	15.9%	0.913	16.9%	0.009	0.9%	
23	508-1026	4166	359	0.962	5.7%	0.954	6.3%	0.008	0.6%	
24	513	2898	359	0.897	11.3%	0.883	12.1%	0.014	0.8%	
25	517	6563	361	0.921	15.5%	0.918	15.8%	0.002	0.3%	
26	520	3933	362	0.876	9.8%	0.874	9.9%	0.002	0.1%	
27	682	3879	362	0.953	11.6%	0.949	12.0%	0.003	0.4%	
28	1085	3656	359	0.962	9.4%	0.952	10.6%	0.010	1.2%	
29	1146	5887	354	0.972	3.8%	0.968	4.0%	0.004	0.3%	
30	1501	2624	362	0.897	12.3%	0.892	12.6%	0.005	0.3%	
31	1503	8001	359	0.928	17.6%	0.908	19.4%	0.020	1.9%	
32	1504	3989	359	0.923	12.4%	0.909	13.4%	0.014	1.0%	
33	1506	3967	362	0.943	15.4%	0.936	16.1%	0.007	0.8%	
34	1507	3025	340	0.925	18.1%	0.916	19.1%	0.009	1.0%	wavering SVM regression model
35	1508	6005	361	0.921	17.0%	0.914	17.8%	0.007	0.8%	wavering SVM regression model
36	1513	5936	318	0.975	8.8%	0.964	10.5%	0.010	1.7%	wavering SVM regression model
37	1530	6290	361	0.970	10.8%	0.945	14.6%	0.025	3.8%	decrease in slope in the high temp.
38	1537	9983	362	0.969	11.8%	0.956	14.2%	0.013	2.4%	decrease in slope in the high temp.
39	1600	2649	357	0.932	15.4%	0.930	15.6%	0.002	0.2%	
40	1811	6706	361	0.967	13.3%	0.954	15.8%	0.013	2.5%	decrease in slope in the high temp.
41	1911	9129	350	0.969	7.0%	0.963	7.6%	0.006	0.6%	
Average								0.008	0.93%	

Additional Comparative Study Experiment with CPLR (3P)

Although 4P CPLR is used for the comparative experiment, some of the buildings are more clearly explained by 3P CPLR model actually. Those buildings are reexamined with 3P model.

(a)

(b)
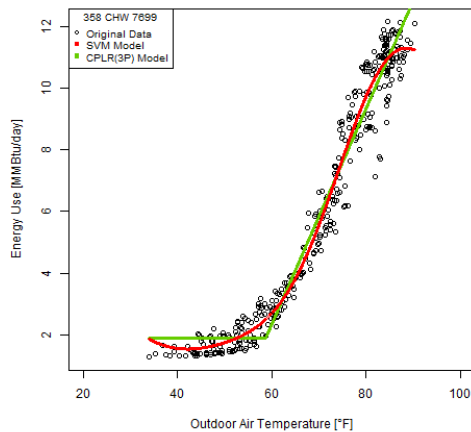
CPLR (3P): CV (%) = 11.7, R^2 = 0.963
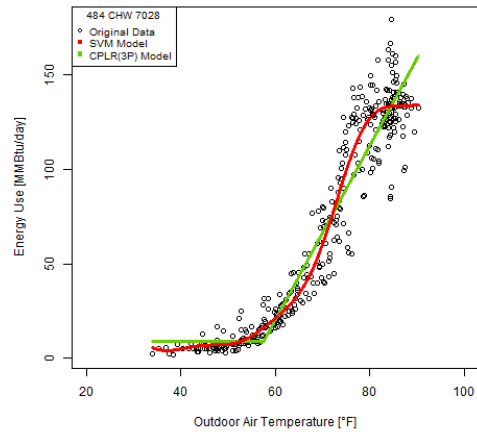
CPLR (3P): CV (%) = 21.4, R^2 = 0.920

SVM   : CV (%) = 10.4, R^2 = 0.970

SVM   : CV (%) = 18.5, R^2 = 0.939

(c)

(d)
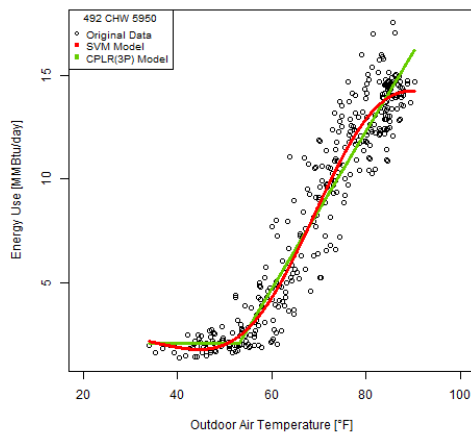
CPLR (3P): CV (%) = 16.9, R^2 = 0.913
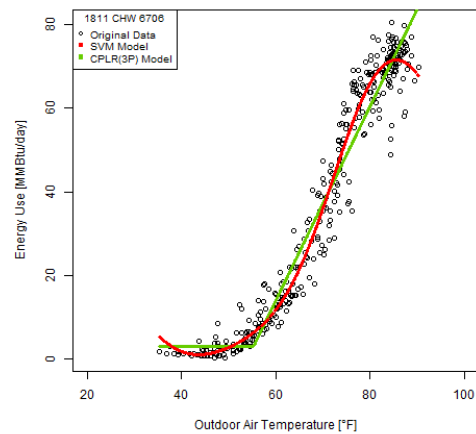
CPLR (3P): CV (%) =15.9, R^2 = 0.953

SVM   : CV (%) = 15.9, R^2 = 0.922

SVM   : CV (%) = 13.3, R^2 = 0.967

Figure 21 - SVM Regression vs CPLR (3P) (CHW-OAT)

Table 20 - Mean Difference in Performance between SVM and CPLR (4P) (CHW-OAE) (Daily)

		Training (Baseline) Period Performance					Testing Period Performance				
		SVM		CPLR (4P)		95% Confidence?	SVM		CPLR (4P)		95% Confidence?
Energy	Regressor	CV	R2	CV	R2		CV	R2	CV	R2	
CHW	OAT	11.2%	0.941	11.4%	0.940	CV, R2	16.9%	0.880	17.2%	0.878	CV
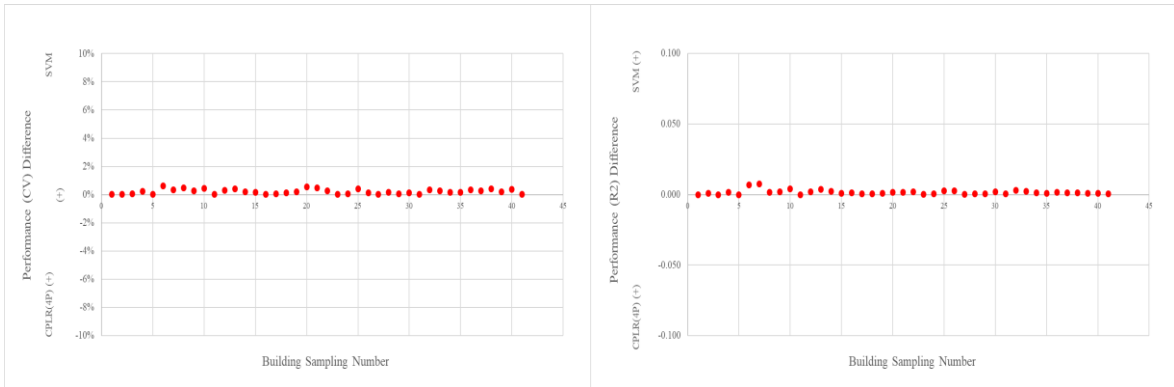
Figure 22 - SVM vs CPLR (4P) (CHW-OAE) for Training (Baseline) Period Performance (Daily)
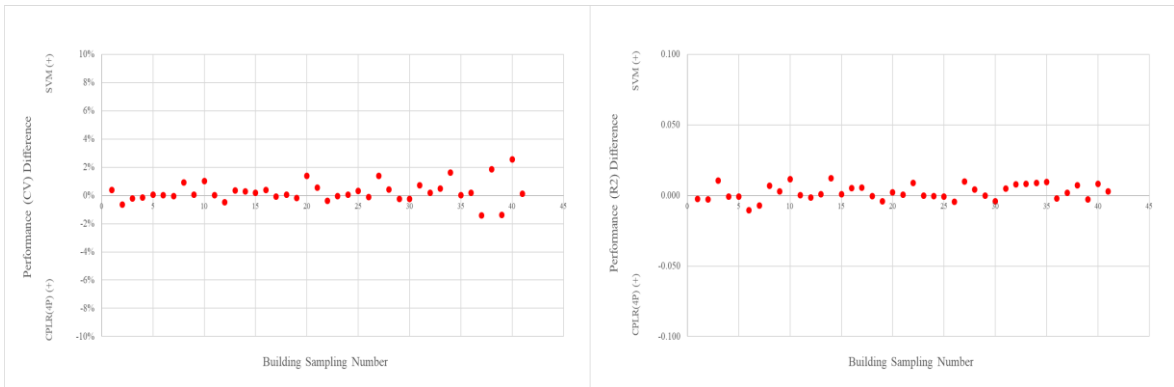
Figure 23 - SVM vs CPLR (4P) (CHW-OAE) for Testing Period Performance (Daily)

Table 21 - Difference in Performance between SVM and CPLR (4P) (CHW-OAE) for the Training Period (Daily)

Sampl. No.	Building No.	Meter ID	No. of data	SVM		CPLR (4P)		Difference		Remark (on SVM Model)
				R2	CV	R2	CV	R2	CV	
1	291	2132	338	0.941	11.9%	0.941	12.0%	0.000	0.0%	
2	325-385	9123	355	0.935	13.9%	0.934	14.0%	0.001	0.0%	
3	353	2746	317	0.848	13.7%	0.848	13.8%	0.000	0.0%	
4	358	7699	362	0.960	12.2%	0.959	12.4%	0.002	0.2%	
5	359-432	6419	362	0.937	8.4%	0.937	8.5%	0.000	0.0%	
6	383	2863	360	0.887	15.4%	0.880	16.1%	0.007	0.6%	wavering SVM regression model
7	384	2583	361	0.899	9.4%	0.891	9.7%	0.007	0.3%	wavering SVM regression model
8	386	2250	357	0.987	6.9%	0.985	7.4%	0.002	0.5%	
9	387	5805	346	0.933	12.4%	0.931	12.7%	0.002	0.3%	
10	394	2117	358	0.935	14.0%	0.931	14.5%	0.004	0.5%	
11	405	7918	360	0.953	11.4%	0.954	11.4%	0.000	0.0%	
12	405-407-1402	7722	358	0.970	10.9%	0.968	11.2%	0.002	0.3%	
13	426-427-428	2848	358	0.952	10.1%	0.949	10.5%	0.004	0.4%	
14	444	2922	360	0.941	8.1%	0.939	8.3%	0.002	0.2%	
15	449	3981	310	0.975	7.0%	0.974	7.1%	0.001	0.2%	
16	454	7584	362	0.894	15.9%	0.893	15.9%	0.001	0.0%	
17	468	3895	362	0.896	16.2%	0.895	16.3%	0.001	0.0%	
18	468	3903	353	0.965	9.0%	0.965	9.1%	0.001	0.1%	
19	478	7968	362	0.936	15.8%	0.935	16.0%	0.001	0.2%	
20	484	7028	362	0.986	9.0%	0.984	9.5%	0.002	0.5%	
21	484	7223	362	0.987	7.8%	0.985	8.3%	0.002	0.5%	
22	492	5950	357	0.938	14.1%	0.937	14.4%	0.002	0.3%	
23	508-1026	4166	359	0.960	5.8%	0.960	5.8%	0.000	0.0%	
24	513	2898	359	0.934	9.0%	0.934	9.1%	0.001	0.1%	
25	517	6563	361	0.915	16.0%	0.912	16.4%	0.003	0.4%	
26	520	3933	362	0.851	10.7%	0.849	10.8%	0.002	0.1%	
27	682	3879	362	0.937	13.5%	0.937	13.5%	0.000	0.0%	
28	1085	3656	359	0.972	8.0%	0.971	8.2%	0.001	0.1%	
29	1146	5887	354	0.930	6.0%	0.929	6.0%	0.001	0.0%	
30	1501	2624	362	0.927	10.4%	0.925	10.5%	0.002	0.1%	
31	1503	8001	359	0.954	13.8%	0.954	13.8%	0.001	0.0%	
32	1504	3989	362	0.948	10.1%	0.945	10.4%	0.003	0.3%	
33	1506	3967	362	0.924	17.6%	0.922	17.8%	0.002	0.3%	
34	1507	3025	340	0.964	12.7%	0.962	12.8%	0.001	0.2%	
35	1508	6005	361	0.920	17.0%	0.920	17.2%	0.001	0.2%	
36	1513	5936	318	0.987	6.3%	0.986	6.6%	0.001	0.3%	
37	1530	6290	361	0.984	7.7%	0.983	8.0%	0.001	0.3%	
38	1537	9983	362	0.992	6.1%	0.991	6.6%	0.001	0.4%	
39	1600	2649	357	0.871	21.1%	0.870	21.3%	0.001	0.2%	
40	1811	6706	361	0.992	6.5%	0.991	6.9%	0.001	0.4%	
41	1911	9129	350	0.977	6.0%	0.977	6.0%	0.000	0.0%	
Average								0.016	0.22%	

Table 22 - Mean Difference in Performance between SVM and CPLR (4P) (CHW-OEE) (Daily)

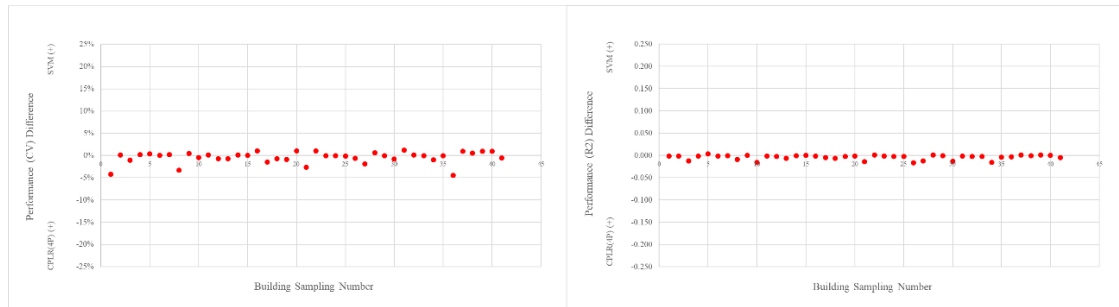|		Training (Baseline) Period Performance					Testing Period Performance				
		SVM		CPLR (4P)		95% Confidence?	SVM		CPLR (4P)		95% Confidence?
Energy	Regressor	CV	R2	CV	R2		CV	R2	CV	R2	
CHW	OEE	10.7%	0.945	10.9%	0.944	CV, R2	16.2%	0.889	16.3%	0.889	None

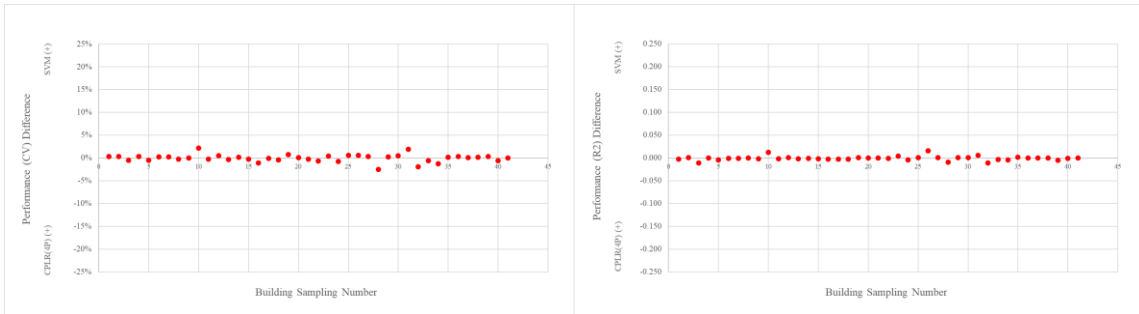Figure 24 - SVM vs CPLR (4P) (CHW-OEE) for Training (Baseline) Period Performance (Daily)

Figure 25 - SVM vs CPLR (4P) (CHW-OEE) for Testing Period Performance (Daily)

Table 23 - Difference in Performance between SVM and CPLR (4P) (CHW-OEE) for the Training Period (Daily)

Sampl. No.	Building No.	Meter ID	No. of data	SVM		CPLR (4P)		Difference		Remark (on SVM Model)
				R2	CV	R2	CV	R2	CV	
1	291	2132	338	0.940	12.1%	0.939	12.2%	0.001	0.1%	
2	325-385	9123	355	0.947	12.5%	0.946	12.6%	0.001	0.1%	
3	353	2746	317	0.853	13.5%	0.854	13.5%	-0.001	0.0%	
4	358	7699	362	0.965	11.4%	0.963	11.7%	0.002	0.3%	
5	359-432	6419	362	0.944	8.0%	0.945	8.0%	-0.001	0.0%	
6	383	2863	360	0.888	15.4%	0.884	15.8%	0.005	0.4%	wavering SVM regression model
7	384	2583	361	0.900	9.3%	0.895	9.6%	0.005	0.3%	wavering SVM regression model
8	386	2250	357	0.989	6.4%	0.988	6.6%	0.001	0.2%	
9	387	5805	346	0.934	12.4%	0.934	12.4%	0.000	0.1%	
10	394	2117	358	0.938	13.6%	0.936	13.8%	0.002	0.2%	
11	405	7918	360	0.961	10.3%	0.961	10.5%	0.001	0.2%	
12	405-407-1402	7722	358	0.974	10.3%	0.973	10.4%	0.001	0.1%	
13	426-427-428	2848	358	0.955	9.9%	0.954	10.0%	0.001	0.1%	
14	444	2922	360	0.946	7.7%	0.945	7.9%	0.001	0.1%	
15	449	3981	310	0.976	6.8%	0.976	6.8%	0.000	0.0%	
16	454	7584	362	0.899	15.6%	0.897	15.6%	0.002	0.0%	
17	468	3895	362	0.897	16.1%	0.899	16.0%	-0.002	-0.1%	
18	468	3903	353	0.968	8.7%	0.968	8.7%	0.000	0.1%	
19	478	7968	362	0.938	15.6%	0.939	15.6%	0.000	0.0%	
20	484	7028	362	0.988	8.2%	0.987	8.6%	0.001	0.5%	
21	484	7223	362	0.989	7.3%	0.988	7.7%	0.001	0.4%	
22	492	5950	357	0.941	13.8%	0.936	14.5%	0.005	0.7%	
23	508-1026	4166	359	0.967	5.3%	0.965	5.4%	0.001	0.1%	
24	513	2898	359	0.933	9.1%	0.930	9.3%	0.002	0.2%	
25	517	6563	361	0.915	15.9%	0.916	16.0%	0.000	0.1%	
26	520	3933	362	0.854	10.6%	0.851	10.7%	0.003	0.1%	
27	682	3879	362	0.946	12.5%	0.945	12.6%	0.001	0.1%	
28	1085	3656	359	0.971	8.1%	0.971	8.2%	0.000	0.0%	
29	1146	5887	354	0.940	5.5%	0.940	5.5%	0.000	0.0%	
30	1501	2624	362	0.930	10.2%	0.928	10.3%	0.002	0.1%	
31	1503	8001	359	0.960	12.7%	0.959	12.9%	0.001	0.2%	
32	1504	3989	362	0.953	9.6%	0.949	10.0%	0.004	0.4%	
33	1506	3967	362	0.925	17.3%	0.925	17.6%	0.001	0.2%	
34	1507	3025	340	0.966	12.1%	0.965	12.4%	0.001	0.3%	
35	1508	6005	361	0.925	16.5%	0.926	16.5%	-0.001	0.0%	
36	1513	5936	318	0.989	5.9%	0.988	6.0%	0.001	0.2%	
37	1530	6290	361	0.991	5.9%	0.987	7.0%	0.004	1.1%	
38	1537	9983	362	0.995	4.8%	0.995	5.0%	0.001	0.2%	
39	1600	2649	357	0.873	21.1%	0.874	21.0%	-0.001	-0.2%	
40	1811	6706	361	0.994	5.6%	0.994	5.5%	0.000	0.0%	
41	1911	9129	350	0.984	5.0%	0.984	5.1%	0.000	0.1%	
Average								0.0012	0.18%	

Table 24 - Mean Difference in Performance between SVM and CPLR (4P) (HHW-OAT)
(Daily)

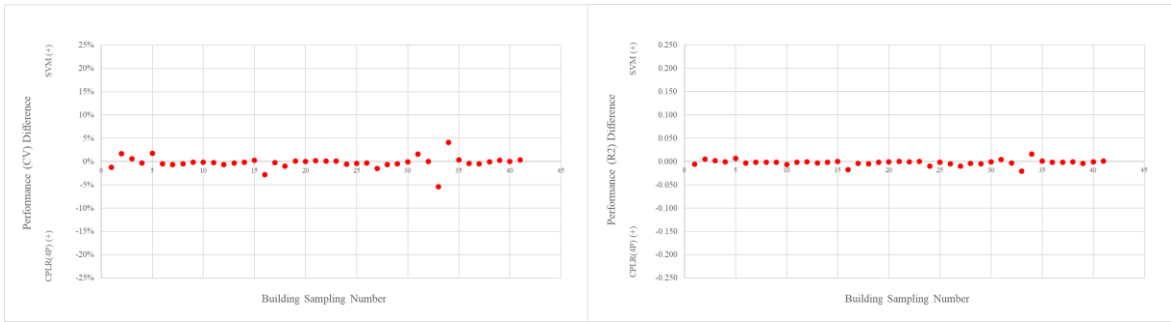|		Training (Baseline) Period Performance					Testing Period Performance				
		SVM		CPLR (4P)		95% Confidence?	SVM		CPLR (4P)		95% Confidence?
Energy	Regressor	CV	R2	CV	R2		CV	R2	CV	R2	
HHW	OAT	19.5%	0.903	20.1%	0.897	CV, R2	25.9%	0.884	26.6%	0.879	CV, R2
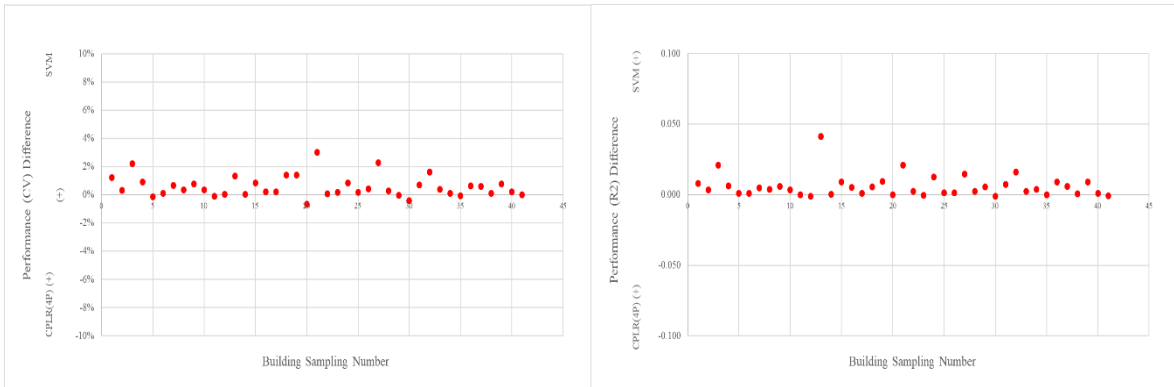
Figure 26 - SVM vs CPLR (4P) (HHW-OAT) for Training (Baseline) Period Performance
(Daily)
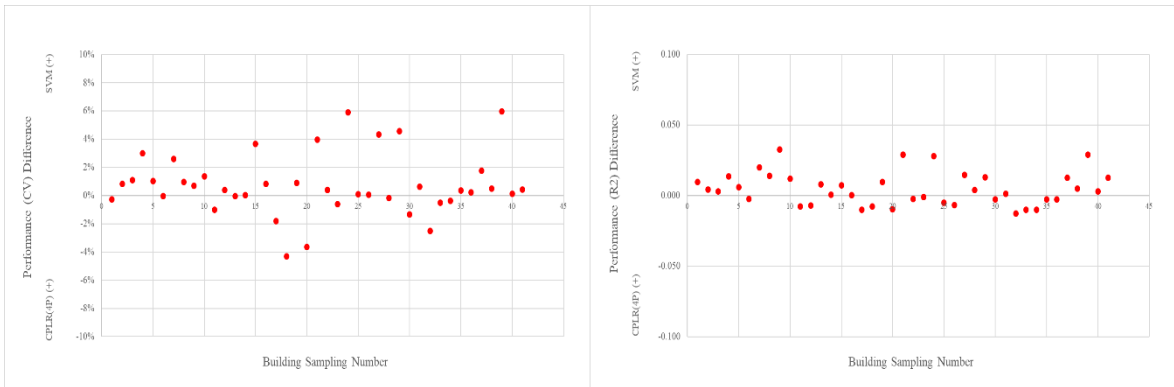
Figure 27 - SVM vs CPLR (4P) (HHW-OAT) for Testing Period Performance (Daily)

Table 25 - Difference in Performance between SVM and CPLR (4P) (HHW-OAT) for the Training Period (Daily)

Sampl. No.	Building No.	Meter ID	No. of data	SVM		CPLR (4P)		Difference		Remark (on SVM Model)
				R2	CV	R2	CV	R2	CV	
1	275	7717	358	0.903	33.3%	0.895	35%	0.008	1.2%	
2	325-385	9124	340	0.941	11.5%	0.938	12%	0.003	0.3%	
3	353	2757	311	0.907	28.1%	0.886	30%	0.021	2.2%	curved
4	359-432	6423	356	0.944	13.9%	0.938	15%	0.006	0.9%	
5	376	7119	360	0.925	16.7%	0.924	17%	0.001	-0.1%	
6	384	2587	361	0.962	6.9%	0.961	7%	0.001	0.1%	
7	386	2254	362	0.954	12.9%	0.95	14%	0.005	0.7%	
8	405	7789	352	0.882	16.3%	0.878	17%	0.004	0.4%	
9	405-407-1402	7723	355	0.911	22.1%	0.905	23%	0.006	0.8%	
10	405-407-1402	7919	349	0.881	16.3%	0.878	17%	0.004	0.3%	
11	408	10036	359	0.933	16.9%	0.933	17%	0.000	-0.1%	
12	420	9146	357	0.912	21.3%	0.913	21%	-0.001	0.0%	
13	425	2611	354	0.745	17.4%	0.704	19%	0.041	1.3%	Break-points
14	426-427-428	2859	353	0.963	14.1%	0.962	14%	0.000	0.1%	
15	435	2796	337	0.812	26.8%	0.803	28%	0.009	0.9%	
16	436	2423	361	0.924	13.3%	0.919	14%	0.005	0.2%	
17	443	6392	352	0.914	25.2%	0.913	25%	0.001	0.2%	
18	444	6435	325	0.962	18.7%	0.957	20%	0.006	1.4%	
19	445	6415	355	0.881	27.5%	0.871	29%	0.009	1.4%	
20	449	3985	306	0.964	16.4%	0.964	16%	0.000	-0.7%	
21	454	7585	362	0.920	22.1%	0.899	25%	0.021	3.0%	curved
22	468	3899	347	0.830	21.7%	0.828	22%	0.003	0.1%	
23	473	7947	352	0.895	28.7%	0.895	29%	0.000	0.2%	
24	476	8034	361	0.909	46.6%	0.896	47%	0.013	0.8%	
25	477	3668	356	0.959	13.8%	0.958	14%	0.001	0.2%	
26	478	7969	355	0.865	27.4%	0.864	28%	0.001	0.4%	
27	483	3891	311	0.921	26.7%	0.907	29%	0.015	2.3%	curved
28	484	7227	358	0.959	8.6%	0.957	9%	0.002	0.3%	
29	496	6933	353	0.836	58.8%	0.83	59%	0.005	0.0%	
30	1085	3660	351	0.904	14.3%	0.905	14%	-0.001	-0.4%	
31	1156	7683	338	0.911	20.4%	0.903	21%	0.007	0.7%	
32	1184	6999	344	0.925	15.6%	0.909	17%	0.016	1.6%	
33	1501	2628	360	0.954	16.4%	0.951	17%	0.003	0.4%	
34	1502	2603	360	0.669	11.6%	0.665	12%	0.004	0.1%	
35	1504	3993	362	0.814	18.1%	0.814	18%	0.000	-0.1%	
36	1513	5895	352	0.912	13.3%	0.903	14%	0.009	0.6%	
37	1537	9984	362	0.962	8.0%	0.957	9%	0.006	0.6%	
38	1611	8467	358	0.962	7.7%	0.961	8%	0.001	0.1%	
39	1800	5472	349	0.858	25.1%	0.849	26%	0.009	0.8%	
40	1904	6366	361	0.938	13.7%	0.937	14%	0.001	0.2%	
41	1911	9133	358	0.907	6.2%	0.908	6%	-0.001	0.0%	
Average								0.0060	0.57%	

Additional Comparative Study Experiment with CPLR (3P)

Although 4P CPLR is used for the comparative experiment, some of the buildings are more clearly explained by 3P CPLR model actually. Those buildings are reexamined with 3P model.
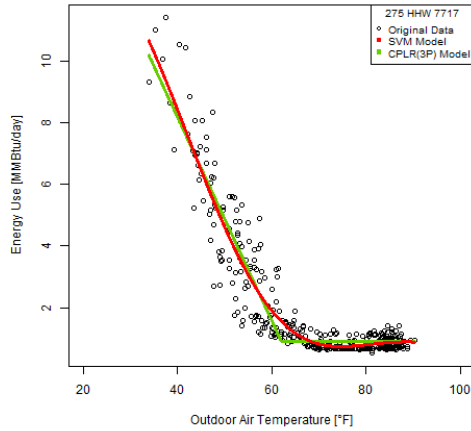
(a)

CPLR (3P): CV (%) = 34.8, R^2 = 0.894

SVM   : CV (%) = 33.3, R^2 = 0.903
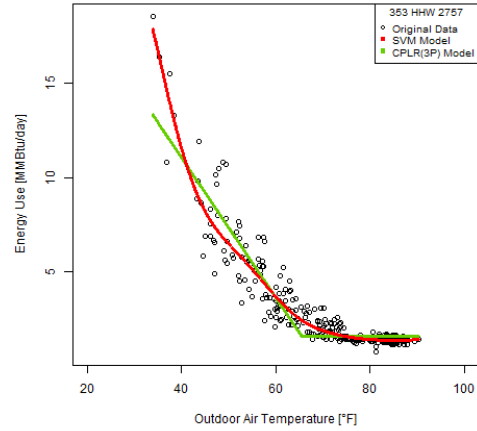
(b)

CPLR (3P): CV (%) = 32.0, R^2 = 0.873

SVM   : CV (%) = 28.1, R^2 = 0.907
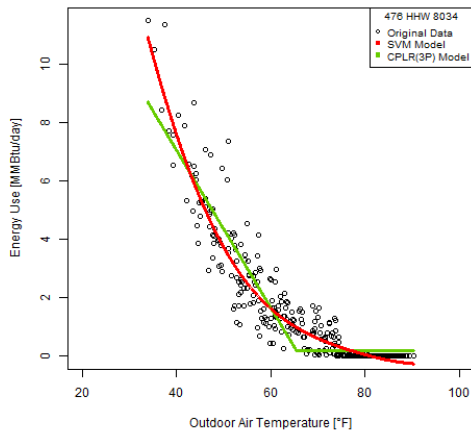
(c)

CPLR (3P): CV (%) = 52.2, R^2 = 0.874

SVM   : CV (%) = 46.6, R^2 = 0.909
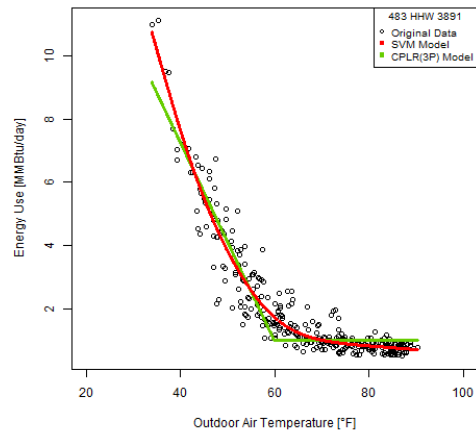
(d)

CPLR (3P): CV (%) =31.2, R^2 = 0.891

SVM   : CV (%) = 26.7, R^2 = 0.921

Figure 28 - SVM Regression vs CPLR (3P) (HHW-OAT)

Results Based on Building Type

The difference in performance is arranged in terms of building types.

Table 26 - Difference in Performance between SVM and CPLR (4P) for the Training Period (Daily)

	Performance Difference*							
	R2				CV			
	CHW -OAT	CHW -OAE	CHW -OEE	HHW -OAT	CHW -OAT	CHW -OAE	CHW -OEE	HHW -OAT
Dorms/Res Hall	0.003	0.002	0.001	0.003	0.30%	0.24%	0.16%	0.28%
General/Dining/Sport	0.006	0.002	0.002	0.007	0.63%	0.15%	0.11%	0.78%
Lab/Classroom	0.011	0.001	0.001	0.004	1.31%	0.24%	0.25%	0.44%
Office/Classroom	0.004	0.002	0.000	0.010	0.35%	0.20%	0.08%	0.86%
Others	0.008	0.001	0.000	0.004	1.01%	0.20%	0.11%	0.35%

*SVM showing higher performance is indicated positive.

Table 27 - Difference in Performance between SVM and CPLR (4P) for the Testing Period (Daily)

	Performance Difference*							
	R2				CV			
	CHW -OAT	CHW -OAE	CHW -OEE	HHW -OAT	CHW -OAT	CHW -OAE	CHW -OEE	HHW -OAT
Dorms/Res Hall	-0.002	0.001	-0.002	0.007	-0.43%	-0.11%	0.25%	1.13%
General/Dining/Sport	-0.006	0.003	0.002	0.000	-0.50%	0.13%	0.21%	0.65%
Lab/Classroom	0.000	0.003	0.001	0.001	-0.85%	0.32%	0.00%	0.32%
Office/Classroom	0.002	0.003	0.003	0.010	0.14%	0.33%	0.18%	1.50%
Others	0.000	0.001	-0.003	0.015	-0.54%	0.40%	0.11%	1.13%

*SVM showing higher performance is indicated positive.

5.4.2.2 Monthly Data

The Table 28 through Table 35, and Figure 29 through Figure 36 display the result of

comparative study for monthly data. Relevant SVM regression and CPLR(4P) are listed in

the Appendix. D.2.

Table 28 - Mean Difference in Performance between SVM and CPLR (4P) (CHW-OAT)
(Monthly)

		Training (Baseline) Period Performance					Testing Period Performance				
		SVM		CPLR (4P)		95% Confidence?	SVM		CPLR (4P)		95% Confidence?
Energy	Regressor	CV	R2	CV	R2		CV	R2	CV	R2	
CHW	OAT	8.02%	0.973	7.63%	0.977	CV	14.2%	0.971	12.4%	0.977	CV

Figure 29 - SVM vs CPLR (4P) (CHW-OAT) for Training (Baseline) Period Performance
(Monthly)
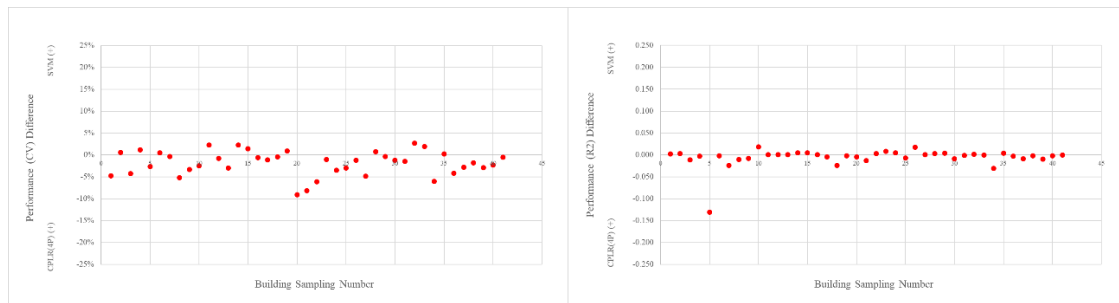
Figure 30 - SVM vs CPLR (4P) (CHW-OAT) for Testing Period Performance (Monthly)

Table 29 - Difference in Performance between SVM and CPLR (4P) (CHW-OAT) for the Training Period (Monthly)

Sampl. No.	Building No.	Meter ID	No. of data	SVM		CPLR (4P)		Difference		Remark (on SVM Model)
				R2	CV	R2	CV	R2	CV	
1	291	2132	12	0.985	10.3%	0.987	6.2%	-0.002	-4.2%	
2	325-385	9123	12	0.986	6.2%	0.988	6.4%	-0.002	0.1%	
3	353	2746	11	0.930	10.3%	0.942	9.2%	-0.013	-1.1%	
4	358	7699	12	0.985	7.3%	0.986	7.5%	-0.001	0.2%	
5	359-432	6419	12	0.983	4.5%	0.979	4.9%	0.004	0.4%	wavering SVM regression model
6	383	2863	12	0.990	4.5%	0.992	4.5%	-0.002	0.1%	
7	384	2583	12	0.988	3.1%	0.989	3.4%	0.000	0.2%	
8	386	2250	12	0.971	12.3%	0.980	8.9%	-0.009	-3.3%	
9	387	5805	12	0.967	8.5%	0.966	8.9%	0.000	0.4%	
10	394	2117	12	0.937	12.9%	0.953	12.4%	-0.016	-0.5%	
11	405	7918	12	0.979	7.7%	0.980	7.8%	-0.001	0.1%	
12	405-407-1402	7722	12	0.981	9.2%	0.983	8.5%	-0.002	-0.7%	
13	426-427-428	2848	12	0.981	6.2%	0.987	5.5%	-0.007	-0.7%	
14	444	2922	12	0.992	2.9%	0.993	2.9%	-0.001	0.1%	
15	449	3981	11	0.983	6.3%	0.983	6.4%	0.001	0.1%	
16	454	7584	12	0.925	12.3%	0.927	13.3%	-0.002	1.0%	
17	468	3895	12	0.988	5.8%	0.993	4.3%	-0.005	-1.5%	wavering SVM regression model
18	468	3903	12	0.952	11.3%	0.958	10.5%	-0.006	-0.8%	
19	478	7968	12	0.986	8.0%	0.988	7.1%	-0.002	-0.9%	
20	484	7028	12	0.973	11.6%	0.974	12.6%	-0.001	1.0%	
21	484	7223	12	0.967	12.7%	0.981	10.1%	-0.014	-2.6%	
22	492	5950	12	0.975	8.4%	0.974	9.5%	0.001	1.1%	
23	508-1026	4166	12	0.990	2.9%	0.991	2.9%	-0.001	-0.1%	
24	513	2898	12	0.930	9.4%	0.932	9.3%	-0.002	-0.1%	
25	517	6563	12	0.986	6.3%	0.988	6.2%	-0.003	-0.1%	
26	520	3933	12	0.943	6.6%	0.959	6.0%	-0.017	-0.6%	
27	682	3879	12	0.970	9.2%	0.983	7.3%	-0.013	-1.9%	
28	1085	3656	12	0.991	4.5%	0.990	5.1%	0.001	0.6%	
29	1146	5887	12	0.995	1.8%	0.995	1.7%	0.000	-0.1%	
30	1501	2624	12	0.960	7.3%	0.973	6.5%	-0.013	-0.8%	
31	1503	8001	12	0.950	13.3%	0.951	14.5%	-0.001	1.2%	
32	1504	3989	12	0.946	10.5%	0.948	10.6%	-0.002	0.2%	
33	1506	3967	12	0.984	8.3%	0.986	8.2%	-0.002	-0.1%	
34	1507	3025	12	0.938	15.4%	0.954	14.4%	-0.016	-1.0%	
35	1508	6005	12	0.977	9.0%	0.981	8.9%	-0.004	-0.1%	
36	1513	5936	12	0.988	10.0%	0.991	5.5%	-0.003	-4.5%	
37	1530	6290	12	0.985	7.0%	0.984	8.0%	0.001	1.0%	
38	1537	9983	12	0.985	7.9%	0.986	8.4%	0.000	0.6%	
39	1600	2649	12	0.997	3.2%	0.995	4.2%	0.001	1.0%	
40	1811	6706	12	0.986	8.2%	0.986	9.2%	0.000	0.9%	
41	1911	9129	12	0.982	5.4%	0.986	4.8%	-0.005	-0.6%	
Average								-0.004	-0.29%	

Table 30 - Mean Difference in Performance between SVM and CPLR (4P) (CHW-OAE) (Monthly)

		Training (Baseline) Period Performance					Testing Period Performance				
		SVM		CPLR (4P)		95% Confidence?	SVM		CPLR (4P)		95% Confidence?
Energy	Regressor	CV	R2	CV	R2		CV	R2	CV	R2	
CHW	OAE	5.38%	0.986	5.34%	0.987	None	12.8%	0.978	12.3%	0.979	None

Figure 31 - SVM vs CPLR (4P) (CHW-OAE) for Training (Baseline) Period Performance (Monthly)
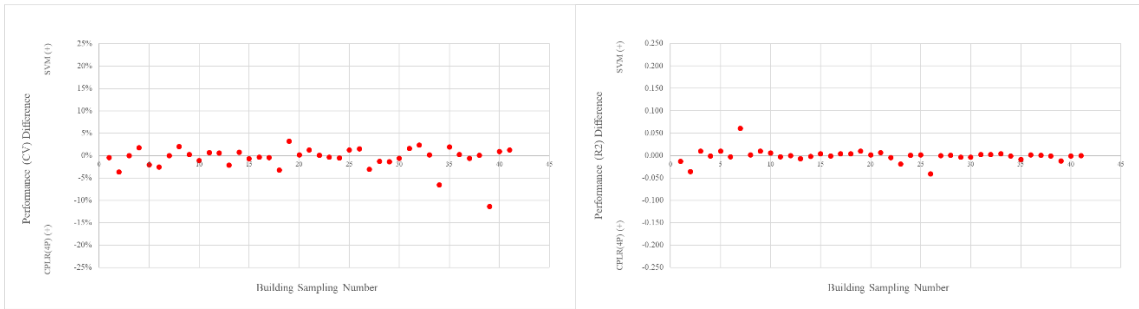
Figure 32 - SVM vs CPLR (4P) (CHW-OAE) for Testing Period Performance (Monthly)

Table 31 - Difference in Performance between SVM and CPLR (4P) (CHW-OAE) for the Training Period (Monthly)

Sampl. No.	Building No.	Meter ID	No. of data	SVM		CPLR (4P)		Difference		Remark (on SVM Model)
				R2	CV	R2	CV	R2	CV	
1	291	2132	12	0.975	7.6%	0.978	7.9%	-0.003	0.4%	
2	325-385	9123	12	0.989	5.9%	0.988	6.3%	0.001	0.3%	wavering SVM regression model
3	353	2746	12	0.946	8.3%	0.957	7.9%	-0.011	-0.5%	
4	358	7699	12	0.996	3.7%	0.996	4.0%	0.000	0.3%	
5	359-432	6419	12	0.986	3.9%	0.990	3.4%	-0.004	-0.5%	
6	383	2863	12	0.988	5.1%	0.988	5.3%	-0.001	0.2%	
7	384	2583	12	0.980	4.0%	0.981	4.3%	-0.001	0.3%	
8	386	2250	12	0.996	4.0%	0.997	3.7%	0.000	-0.3%	
9	387	5805	12	0.990	4.5%	0.991	4.5%	-0.002	0.0%	
10	394	2117	12	0.990	6.5%	0.977	8.6%	0.012	2.2%	wavering SVM regression model
11	405	7918	12	0.990	5.0%	0.992	4.7%	-0.002	-0.3%	
12	405-407-1402	7722	12	0.996	3.9%	0.995	4.4%	0.000	0.5%	
13	426-427-428	2848	12	0.989	4.8%	0.992	4.5%	-0.002	-0.3%	
14	444	2922	12	0.986	3.9%	0.987	4.0%	-0.001	0.1%	
15	449	3981	11	0.992	4.1%	0.994	3.8%	-0.002	-0.3%	
16	454	7584	12	0.956	11.1%	0.958	10.0%	-0.002	-1.1%	
17	468	3895	12	0.984	6.1%	0.986	6.0%	-0.003	-0.1%	
18	468	3903	12	0.988	5.4%	0.991	5.0%	-0.003	-0.4%	
19	478	7968	12	0.992	5.4%	0.991	6.1%	0.001	0.8%	
20	484	7028	12	0.997	3.9%	0.997	4.0%	0.000	0.1%	
21	484	7223	12	0.995	5.5%	0.995	5.3%	0.000	-0.3%	
22	492	5950	12	0.996	4.2%	0.996	3.5%	-0.001	-0.7%	
23	508-1026	4166	12	0.993	2.7%	0.989	3.2%	0.004	0.4%	
24	513	2898	12	0.975	5.9%	0.980	5.1%	-0.004	-0.8%	
25	517	6563	12	0.996	3.3%	0.996	3.8%	0.000	0.6%	
26	520	3933	12	0.963	6.2%	0.948	6.8%	0.015	0.6%	wavering SVM regression model
27	682	3879	12	0.978	8.1%	0.977	8.5%	0.001	0.3%	
28	1085	3656	12	0.979	8.1%	0.988	5.5%	-0.009	-2.5%	
29	1146	5887	12	0.974	3.7%	0.973	4.0%	0.001	0.3%	wavering SVM regression model
30	1501	2624	12	0.999	1.1%	0.998	1.6%	0.001	0.5%	
31	1503	8001	12	0.984	7.9%	0.978	9.7%	0.006	1.9%	wavering SVM regression model
32	1504	3989	12	0.975	7.4%	0.986	5.4%	-0.011	-1.9%	
33	1506	3967	12	0.987	7.3%	0.991	6.7%	-0.004	-0.6%	
34	1507	3025	12	0.975	10.9%	0.980	9.6%	-0.004	-1.3%	
35	1508	6005	12	0.995	5.1%	0.993	5.2%	0.001	0.1%	
36	1513	5936	12	0.995	3.6%	0.996	3.9%	0.000	0.3%	
37	1530	6290	12	0.996	3.5%	0.997	3.6%	0.000	0.1%	
38	1537	9983	12	0.999	2.4%	0.999	2.5%	0.000	0.1%	
39	1600	2649	12	0.964	10.7%	0.969	11.0%	-0.005	0.3%	
40	1811	6706	12	0.998	3.4%	0.999	2.8%	-0.001	-0.6%	
41	1911	9129	12	0.995	2.9%	0.995	2.9%	0.000	0.0%	
Average								-0.0008	-0.05%	

Table 32 - Mean Difference in Performance between SVM and CPLR (4P) (CHW-OEE) (Monthly)

		Training (Baseline) Period Performance					Testing Period Performance				
		SVM		CPLR (4P)		95% Confidence?	SVM		CPLR (4P)		95% Confidence?
Energy	Regressor	CV	R2	CV	R2		CV	R2	CV	R2	
CHW	OEE	5.44%	0.985	5.27%	0.987	R2	12.4%	0.977	12.2%	0.982	R2

Figure 33 - SVM vs CPLR (4P) (CHW-OAE) for Training (Baseline) Period Performance (Monthly)
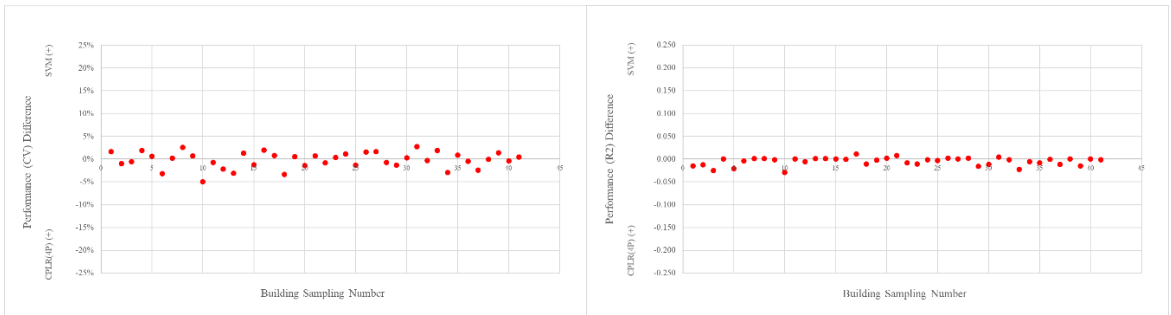
Figure 34 - SVM vs CPLR (4P) (CHW-OAE) for Testing Period Performance (Monthly)

Table 33 - Difference in Performance between SVM and CPLR (4P) (CHW-OEE) for the Training Period (Monthly)

Sampl. No.	Building No.	Meter ID	No. of data	SVM		CPLR (4P)		Difference		Remark (on SVM Model)
				R2	CV	R2	CV	R2	CV	
1	291	2132	12	0.969	9.6%	0.975	8.5%	-0.006	-1.2%	
2	325-385	9123	12	0.992	5.0%	0.986	6.7%	0.005	1.7%	
3	353	2746	11	0.971	6.2%	0.969	6.8%	0.002	0.6%	
4	358	7699	12	0.996	3.6%	0.997	3.3%	-0.001	-0.3%	
5	359-432	6419	12	0.998	1.4%	0.991	3.2%	0.007	1.8%	
6	383	2863	12	0.989	5.0%	0.992	4.5%	-0.003	-0.5%	
7	384	2583	12	0.978	5.1%	0.980	4.5%	-0.002	-0.7%	
8	386	2250	12	0.994	5.0%	0.995	4.5%	-0.001	-0.4%	
9	387	5805	12	0.991	4.3%	0.993	4.2%	-0.002	-0.2%	
10	394	2117	12	0.969	9.0%	0.976	8.9%	-0.006	-0.1%	
11	405	7918	12	0.992	4.7%	0.993	4.5%	-0.002	-0.2%	
12	405-407-1402	7722	12	0.993	5.8%	0.994	5.2%	-0.001	-0.6%	
13	426-427-428	2848	12	0.988	4.9%	0.991	4.6%	-0.003	-0.3%	
14	444	2922	12	0.987	3.9%	0.989	3.8%	-0.002	-0.1%	
15	449	3981	11	0.995	3.3%	0.995	3.6%	0.000	0.3%	
16	454	7584	12	0.953	11.2%	0.970	8.4%	-0.017	-2.8%	
17	468	3895	12	0.978	7.1%	0.982	6.9%	-0.004	-0.2%	
18	468	3903	12	0.988	5.3%	0.993	4.3%	-0.005	-0.9%	
19	478	7968	12	0.992	5.3%	0.993	5.4%	-0.001	0.1%	
20	484	7028	12	0.997	3.6%	0.998	3.6%	0.000	0.0%	
21	484	7223	12	0.995	5.0%	0.995	5.2%	0.000	0.2%	
22	492	5950	12	0.995	3.7%	0.996	3.8%	-0.001	0.1%	
23	508-1026	4166	12	0.991	2.8%	0.991	2.9%	0.000	0.1%	
24	513	2898	12	0.969	5.8%	0.979	5.2%	-0.010	-0.6%	
25	517	6563	12	0.995	3.9%	0.996	3.5%	-0.002	-0.4%	
26	520	3933	12	0.940	7.3%	0.944	7.0%	-0.004	-0.3%	
27	682	3879	12	0.968	9.9%	0.977	8.4%	-0.010	-1.5%	
28	1085	3656	12	0.984	6.0%	0.989	5.4%	-0.004	-0.6%	
29	1146	5887	12	0.968	4.4%	0.973	3.9%	-0.005	-0.5%	
30	1501	2624	12	0.997	2.0%	0.997	2.0%	-0.001	0.0%	
31	1503	8001	12	0.982	8.1%	0.978	9.8%	0.004	1.6%	
32	1504	3989	12	0.981	5.8%	0.984	5.8%	-0.003	0.0%	
33	1506	3967	12	0.971	11.6%	0.992	6.2%	-0.021	-5.4%	
34	1507	3025	12	0.994	5.9%	0.978	10.0%	0.016	4.1%	
35	1508	6005	12	0.994	5.1%	0.993	5.4%	0.001	0.3%	
36	1513	5936	12	0.994	4.1%	0.996	3.8%	-0.002	-0.4%	
37	1530	6290	12	0.996	3.8%	0.997	3.3%	-0.002	-0.5%	
38	1537	9983	12	0.998	2.6%	0.999	2.6%	0.000	-0.1%	
39	1600	2649	12	0.968	10.1%	0.972	10.4%	-0.004	0.3%	
40	1811	6706	12	0.998	3.3%	0.998	3.3%	0.000	0.0%	
41	1911	9129	12	0.996	2.4%	0.995	2.8%	0.001	0.4%	
Average								**-0.002**	**-0.15%**	

Table 34 - Mean Difference in Performance between SVM and CPLR (4P) (HHW-OAT)
(Monthly)

		Training (Baseline) Period Performance					Testing Period Performance				
		SVM		CPLR (4P)		95% Confidence?	SVM		CPLR (4P)		95% Confidence?
Energy	Regressor	CV	R2	CV	R2		CV	R2	CV	R2	
HHW	OAT	8.4%	0.970	8.8%	0.970	None	22.0%	0.918	20.5%	0.935	CV, R2
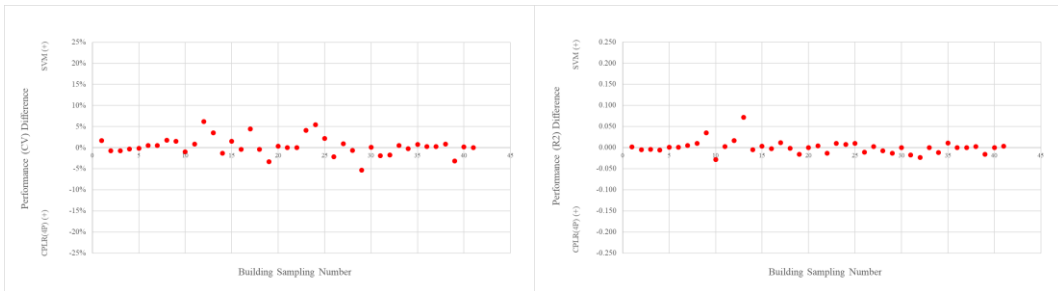
Figure 35 - SVM vs CPLR (4P) (CHW-OAE) for Training (Baseline) Period Performance
(Monthly)
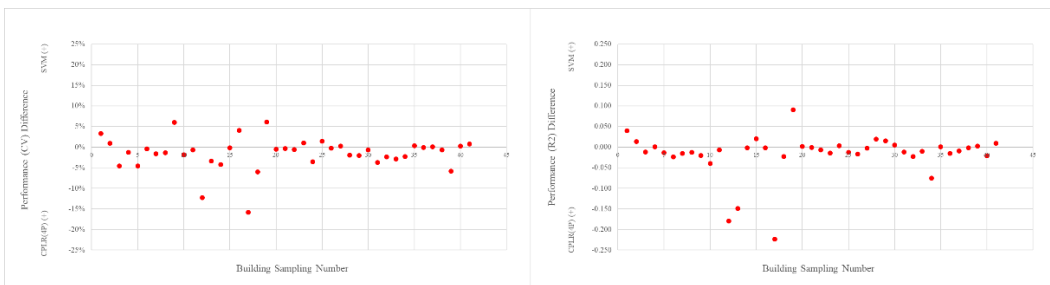
Figure 36 - SVM vs CPLR (4P) (CHW-OAE) for Testing Period Performance (Monthly)

Table 35 - Difference in Performance between SVM and CPLR (4P) (HHW-OAT) for the Training Period (Monthly)

Sampl. No.	Building No.	Meter ID	No. of data	SVM		CPLR (4P)		Difference		Remark (on SVM Model)
				R2	CV	R2	CV	R2	CV	
1	275	7717	12	0.997	4.8%	0.995	6%	0.002	1.6%	
2	325-385	9124	12	0.988	4.5%	0.993	4%	-0.005	-0.7%	
3	353	2757	11	0.935	19.5%	0.938	19%	-0.004	-0.7%	
4	359-432	6423	12	0.975	8.5%	0.98	8%	-0.006	-0.3%	
5	376	7119	12	0.947	13.4%	0.946	13%	0.001	-0.1%	wavering SVM regression model
6	384	2587	12	0.997	1.9%	0.996	2%	0.001	0.5%	
7	386	2254	12	0.959	9.9%	0.954	10%	0.005	0.5%	wavering SVM regression model
8	405	7789	12	0.957	8.7%	0.947	10%	0.010	1.8%	
9	405-407-1402	7723	12	0.982	8.9%	0.946	10%	0.035	1.6%	
10	405-407-1402	7919	12	0.955	10.0%	0.983	9%	-0.028	-1.0%	
11	408	10036	12	0.996	3.8%	0.994	5%	0.002	0.9%	
12	420	9146	12	0.997	3.8%	0.98	10%	0.016	6.2%	wavering SVM regression model
13	425	2611	12	0.946	5.7%	0.874	9%	0.072	3.5%	
14	426-427-428	2859	12	0.989	7.0%	0.994	6%	-0.005	-1.3%	
15	435	2796	12	0.996	3.1%	0.993	5%	0.003	1.5%	
16	436	2423	12	0.905	12.6%	0.907	12%	-0.002	-0.4%	
17	443	6392	12	0.986	9.3%	0.974	14%	0.011	4.4%	wavering SVM regression model
18	444	6435	12	0.994	6.5%	0.995	6%	-0.002	-0.4%	
19	445	6415	12	0.963	14.0%	0.979	11%	-0.016	-3.3%	
20	449	3985	11	0.999	2.3%	0.999	3%	0.000	0.3%	
21	454	7585	12	0.953	16.2%	0.948	16%	0.005	0.0%	
22	468	3899	12	0.895	13.9%	0.909	14%	-0.013	0.0%	
23	473	7947	12	0.983	10.2%	0.973	14%	0.010	4.1%	wavering SVM regression model
24	476	8034	12	0.989	11.9%	0.981	17%	0.008	5.4%	
25	477	3668	12	0.984	8.2%	0.974	10%	0.010	2.2%	
26	478	7969	12	0.975	10.8%	0.985	9%	-0.011	-2.2%	
27	483	3891	12	0.988	8.3%	0.985	9%	0.003	0.9%	
28	484	7227	12	0.981	4.7%	0.989	4%	-0.007	-0.7%	
29	496	6933	12	0.968	21.5%	0.982	16%	-0.013	-5.4%	
30	1085	3660	12	0.980	5.8%	0.98	6%	0.000	0.1%	
31	1156	7683	12	0.944	13.8%	0.961	12%	-0.017	-1.9%	
32	1184	6999	12	0.920	12.3%	0.943	11%	-0.024	-1.7%	
33	1501	2628	12	0.995	5.1%	0.995	6%	0.000	0.5%	
34	1502	2603	12	0.944	4.1%	0.955	4%	-0.012	-0.2%	
35	1504	3993	12	0.962	7.6%	0.951	8%	0.011	0.7%	Original data is abnormal
36	1513	5895	12	0.958	7.5%	0.958	8%	0.000	0.2%	
37	1537	9984	12	0.996	2.1%	0.996	2%	0.000	0.3%	
38	1611	8467	12	0.994	2.5%	0.991	3%	0.003	0.8%	
39	1800	5472	12	0.962	11.6%	0.978	8%	-0.016	-3.2%	
40	1904	6366	12	0.991	4.8%	0.991	5%	0.000	0.2%	wavering SVM regression model
41	1911	9133	12	0.961	4.1%	0.957	4%	0.003	0.0%	
Average								0.0008	0.36%	

Results Based on Building Type

The difference in performance is arranged in terms of building types.

Table 36 - Difference in Performance between SVM and CPLR (4P) for the Training Period (Monthly)

| | Performance Difference* | | | | | | | |
| | R2 | | | | CV | | | |
	CHW -OAT	CHW -OAE	CHW -OEE	HHW -OAT	CHW -OAT	CHW -OAE	CHW -OEE	HHW -OAT
Dorms/Res Hall	-0.005	0.001	-0.004	0.003	-1.20%	0.50%	-0.48%	0.30%
General/Dining/Sport	-0.005	0.001	-0.005	-0.008	-0.23%	-0.05%	-0.87%	-0.76%
Lab/Classroom	-0.004	-0.001	-0.002	0.000	-0.52%	-0.12%	-0.14%	-0.02%
Office/Classroom	-0.002	-0.004	0.001	0.006	0.14%	-0.02%	0.62%	0.64%
Others	-0.002	-0.001	-0.001	-0.003	0.09%	-0.28%	-0.12%	-0.28%

*SVM showing higher performance is indicated positive.

Table 37 - Difference in Performance between SVM and CPLR (4P) for the Testing Period (Monthly)

| | Performance Difference* | | | | | | | |
| | R2 | | | | CV | | | |
	CHW -OAT	CHW -OAE	CHW -OEE	HHW -OAT	CHW -OAT	CHW -OAE	CHW -OEE	HHW -OAT
Dorms/Res Hall	0.004	-0.004	-0.010	-0.016	-1.74%	-0.51%	-1.84%	-0.40%
General/Dining/Sport	-0.006	0.004	0.001	-0.003	-0.42%	-0.13%	0.51%	-1.06%
Lab/Classroom	-0.003	0.000	-0.004	-0.024	-2.40%	0.06%	0.13%	-1.64%
Office/Classroom	-0.026	-0.004	-0.013	-0.020	-2.00%	-3.26%	-0.32%	-3.02%
Others	0.000	-0.004	-0.002	-0.010	-1.13%	0.18%	-0.24%	-0.65%

*SVM showing higher performance is indicated positive.

a

CHAPTER VI SUMMARY AND CONCLUSION

- When establishing baseline models for building energy use with one parameter, using a one-dimensional grid search with recommended variables (ODGS) for each parameter can save a significant amount of time with a similar performance. As opposed to the general understanding of how SVM performance is sensitive to the choice of hyperparameters, it does not appear to be highly sensitive in the case of this study. This appears to be caused by a relatively simple setting for SVM regression: less than 365 number of input and one or two independent variables.

- RBF Kernel SVM regression is flexible enough to cover linear, piecewise linear, curved, or more complex patterns. Even when the data is linear, the performance of RBF SVM is similar to linear SVM. SVM usually shows robust performance in various noise conditions. However, it is relatively weak to scattering data that it may generate unreasonable plot. The performance of monthly data is also not robust compared to that of daily data. This happens because the SVM algorithm does not use any explanatory theories about the relationship between building energy use and meteorological conditions, which offers a clue about what a "normal pattern" should look like. Thus, if the percentage of outliers is high, the SVM is not a good option to generate baseline models unless the outliers are appropriately handled in the preprocessing stage.

- Regarding SVM regression on daily data, among independent variables for CHW, OAT+DPT shows the best performance, followed by OEE, OAE, and OAT. The means of CV throughout the samples is 9.32%, 10.7% 11.2%, and 12.2% respectively.

Although OAT+DPT demonstrates better performance than OEE, its feasibility as a

baseline model might be limited because using two inputs is disadvantageous for

visualization and there is a risk of collinearity. As the input changed from OAT through

OAE to OEE, the SVM plots FOR CHW became simpler and more linear. A greater

number of wavering patterns were found with the CHW-OAT plots than those with OAE

and OEE. This appears to indicate that SVM has difficulty making sense of the pattern

due to a lack of needed information, the humidity data. In the case of HHW, since it is

not affected by latent loads, the shape of the HHW-OAT SVM regression plots appears

to be more stable. Regarding monthly data, among single-variate independent variables

for CHW baseline modeling, the performance between operational effective enthalpy

(OEE) and outdoor air enthalpy (OAE) is not significantly different, and these two

showed a better performance than outdoor air temperature (OAT). As in the case with

daily data, OAT+DPT demonstrated the best performance in the training period.

However, the performance for the testing period is worse than the CHW-OAE and

CHW-OEE data. Regardless of different independent variables, SVM generated similar

plots. Compared to the daily application, a large number of wavering patterns were

found, which indicates insufficient input for SVM to make sense of the pattern.

- With daily data, the SVM demonstrates a slightly better fitting performance, 0.5% ~

3%, than the change point linear regression model, and much better performance, nearly

10%, than multiple linear regression. SVM tended to show a higher performance when it

made a pattern that is difficult to be expressed by CPLR. In the case of CHW-OAT, a

more than 2% higher performance of SVM regression was observed when SVM

generated plots that have an abrupt slope change in the high-temperature range. With HHW-OAT, a more than 2% higher performance was observed when the relationship between the dependent and independent variables took curved forms. For CHW with a single independent variable, as the predictor changes from OAT through OAE to OEE, the mean difference in CV between SVM and CPLR decreased from 0.9%, 0.2%, to less than 0.2%. Visual inspection of the regression plots offers insight into the reason for this. The shape of SVM and CPLR plots become more similar as the predictor changes from OAT through OAE to OEE. With monthly data, the performance of SVM is approximately 2% higher than multiple linear regression and is usually worse than change-point linear regression models. This implies that monthly data does not have sufficient data points for SVM to elaborately describe the patterns.

- Compared with MLR, SVM may always show superior results. Compared with CPLR, for daily data, the mean performance of SVM over that of CPLR is marginal: less than 1%. Hence, for a single-variate whole-building energy inverse modeling, it would be reasonable to use CPLR as a general-purpose modeler because of the balance between the effort and the performance. However, it would be worth considering using SVM if the data appears to be highly non-linear or if even a small gain in accuracy can be regarded as significant. For monthly data, SVM is not recommended as a whole-building energy inverse modeler.

REFERENCES

[1] US-EIA, Annual Energy Review, in, US Energy Information Administration, 2018.

[2] ASHRAE, ASHRAE Guideline 14-2014, Measurement of Energy, Demand, and Water Savings, (2014).

[3] ASHRAE Handbook - Fundamental, ASHRAE, 2017.

[4] Z. Wang, R.S. Srinivasan, A review of artificial intelligence based building energy use prediction: Contrasting the capabilities of single and ensemble prediction models, Renewable and Sustainable Energy Reviews, 75 (2017) 796-808.

[5] K. Amasyali, N.M. El-Gohary, A review of data-driven building energy consumption prediction studies, Renewable & Sustainable Energy Reviews, 81 (2018) 1192-1205.

[6] Q. Li, Q.L. Meng, J.J. Cai, H. Yoshino, A. Mochida, Applying support vector machine to predict hourly cooling load in the building, Applied Energy, 86 (10) (2009) 2249-2256.

[7] H.X. Zhao, F. Magoules, A review on the prediction of building energy consumption, Renewable & Sustainable Energy Reviews, 16 (6) (2012) 3586-3592.

[8] J. Massana, C. Pous, L. Burgas, J. Melendez, J. Colomer, Short-term load forecasting in a non-residential building contrasting models and attributes, Energy and Buildings, 92 (2015) 322-330.

[9] T.A. Reddy, J.F. Kreider, P.S. Curtiss, A. Rabl, Heating and cooling of buildings: principles and practice of energy efficient design, 3rd ed., CRC press, 2016.

[10] M.F. Fels, Measuring energy savings: the scorekeeping approach, Elsevier Sequoia, 1986.

[11] D. Ruch, D.E. Claridge, A 4-Parameter Change-Point Model for Predicting Energy-Consumption in Commercial Buildings, Journal of Solar Energy Engineering-Transactions of the Asme, 114 (2) (1992) 77-83.

[12] A. Rabl, L. Norford, J. Spadaro, Steady state models for analysis of commercial building energy data, in: Proceedings of the ACEEE Summer Study on Energy Efficiency in Buildings, Pacific Grove, CA, 1986, pp. 9.239-239.261.

[13] K.H. Kim, J.S. Haberl, Development of methodology for calibrated simulation in single-family residential buildings using three-parameter change-point regression model, Energy and Buildings, 99 (2015) 140-152.

[14] J.S. Park, S.J. Lee, K.H. Kim, K.W. Kwon, J.W. Jeong, Estimating thermal performance and energy saving potential of residential buildings using utility bills, Energy and Buildings, 110 (2016) 23-30.

[15] J.K. Kissock, J.S. Haberl, D.E. Claridge, Inverse modeling toolkit: numerical algorithms, ASHRAE transactions, 109 (2003) 425.

[16] S. Katipamula, T.A. Reddy, D.E. Claridge, Multivariate regression modeling, Journal of Solar Energy Engineering-Transactions of the Asme, 120 (3) (1998) 177-184.

[17] J.S. Haberl, D.E. Claridge, An expert system for building energy consumption analysis: prototype results, ASHRAE Transactions, 93 (1987) 979-998.

[18] N.R. Draper, H. Smith, Applied Regression Analysis, 2nd ed., Wiley, 1981.

[19] D.S.S. Robert H. Shumway, Time Series Analysis and Its Applications. With R Examples, Springer, New York, 2006.

[20] R.E. Abdel-Aal, A.Z. Al-Garni, Forecasting monthly electric energy consumption in eastern Saudi Arabia using univariate time-series analysis, Energy, 22 (11) (1997) 1059-1069.

[21] K. Kandananond, Forecasting Electricity Demand in Thailand with an Artificial Neural Network Approach, Energies, 4 (8) (2011) 1246-1257.

[22] P. Chujai, N. Kerdprasop, K. Kerdprasop, Time series analysis of household electric consumption with ARIMA and ARMA models, in: Proceedings of the International MultiConference of Engineers and Computer Scientists, 2013, pp. 295-300.

[23] A. Dhar, T.A. Reddy, D.E. Claridge, Modeling hourly energy use in commercial buildings with Fourier series functional forms, Journal of Solar Energy Engineering, 120 (3) (1998) 217-223.

[24] J.E. Seem, J.E. Braun, Adaptive methods for realtime forecasting of building electrical demand, in: ASHRAE Transactions, 1991, pp. 710-721.

[25] A. Dhar, T.A. Reddy, D.E. Claridge, Improved Fourier series approach to modeling hourly energy use in commercial buildings, in: Proceedings of the 1994 ASME/JSME/JSES International Solar Energy Conference, Publ by ASME, 1994.

[26] J.S. Chou, D.K. Bui, Modeling heating and cooling loads by artificial intelligence for energy-efficient building design, Energy and Buildings, 82 (2014) 437-446.

[27] S.A. Kalogirou, Artificial neural networks in energy applications in buildings, International Journal of Low-Carbon Technologies, 1 (3) (2006) 201-216.

[28] Q. Li, Q.L. Meng, J.J. Cai, H. Yoshino, A. Mochida, Predicting hourly cooling load in the building: A comparison of support vector machine and different artificial neural networks, Energy Conversion and Management, 50 (1) (2009) 90-96.

[29] A.E. Ben-Nakhi, M.A. Mahmoud, Cooling load prediction for buildings using general regression neural networks, Energy Conversion and Management, 45 (13-14) (2004) 2127-2141.

[30] X.D. Wu, V. Kumar, J.R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G.J. McLachlan, A. Ng, B. Liu, P.S. Yu, Z.H. Zhou, M. Steinbach, D.J. Hand, D. Steinberg, Top 10 algorithms in data mining, Knowledge and Information Systems, 14 (1) (2008) 1-37.

[31] B. Dong, C. Cao, S.E. Lee, Applying support vector machines to predict building energy consumption in tropical region, Energy and Buildings, 37 (5) (2005) 545-553.

[32] J.A.K. Suykens, J. Vandewalle, Least squares support vector machine classifiers, Neural processing letters, 9 (3) (1999) 293-300.

[33] J.F. Kreider, J.S. Haberl, Predicting hourly building energy use: The great energy predictor shootout--overview and discussion of results, ASHRAE Transactions, 100 (1994).

[34] D.J.C. MacKay, Bayesian nonlinear modeling for the prediction competition, ASHRAE transactions, 100 (2) (1994) 1053 - 1062.

[35] M.B.O. Ohlsson, C.O. Peterson, H. Pi, T.S. Rognvaldsson, B.P.W. Soderberg, Predicting System Loads with Artificial Neural Networks - Methods and Results from "The Great Energy Predictor Shootout", ASHRAE transactions, 100 (2) (1994) 1063-1074.

[36] J.S. Haberl, S. Thamilseran, Great energy predictor shootout II: Measuring retrofit savings--overview and discussion of results, ASHRAE Transactions, 102 (1996).

[37] S. Katipamula, Great energy predictor shootout II: modeling energy use in large commercial buildings, ASHRAE Transactions, 102 (1996).

[38] C. Miller, Data Collection and Result Analysis for the Predictor Shootout III, in: 2020 ASHRAE Winter Conference, ASHRAE, Chicago, 2020.

[39] X.M. Li, J.H. Lu, L.X. Ding, G. Xu, J.B. Li, Building Cooling Load Forecasting Model Based on LS-SVM, 2009 Asia-Pacific Conference on Information Processing (Apcip 2009), Vol 1, Proceedings, (2009) 55-+.

[40] H.X. Zhao, F. Magoules, Parallel Support Vector Machines Applied to the Prediction of Multiple Buildings Energy Consumption, Journal of Algorithms & Computational Technology, 4 (2) (2010) 231-249.

[41] C. Deb, F. Zhang, J. Yang, S.E. Lee, K.W. Shah, A review on time series forecasting techniques for building energy consumption, Renewable & Sustainable Energy Reviews 74 (2017) 902-924.

[42] Y.X. Wei, X.X. Zhang, Y. Shi, L. Xia, S. Pan, J.S. Wu, M.J. Han, X.Y. Zhao, A review of data-driven approaches for prediction and classification of building energy consumption, Renewable & Sustainable Energy Reviews, 82 (2018) 1027-1047.

[43] B. Yildiz, J.I. Bilbao, A.B. Sproul, A review and analysis of regression and machine learning models on commercial building electricity load forecasting, Renewable and Sustainable Energy Reviews, 73 (2017) 1104-1122.

[44] T. Tereshchenko, D. Ivanko, N. Nord, I. Sartori, Analysis of energy signatures and planning of heating and domestic hot water energy use in buildings in Norway, in: E3S Web Conferences, 2019.

[45] https://www.r-project.org/, in.

[46] D. Meyer, E. Dimitriadou, K. Hornik, A. Weingessel, F. Leisch, e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien, in, 2019.

[47] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machine, ACM Transactions on Intelligent Systems and Technology, 2 (3) (2011) 27:21 - 27:27.

[48] Y.H. Tang, W.M. Guo, J.H. Gao, Efficient Model Selection for Support Vector Machine with Gaussian Kernel Function, 2009 Ieee Symposium on Computational Intelligence and Data Mining, (2009) 40-45.

[49] V. Cherkassky, Y. Ma, Practical selection of SVM parameters and noise estimation for SVM regression, Neural networks, 17 (1) (2004) 113-126.

[50] H. Kaneko, K. Funatsu, Fast optimization of hyperparameters for support vector regression models with highly predictive ability, Chemometrics Intelligent Laboratory Systems 142 (2015) 64-69.

[51] N. Louw, S.J. Steel, Variable selection in kernel Fisher discriminant analysis by means of recursive feature elimination, Computational Statistics & Data Analysis, 51 (3) (2006) 2043-2055.

[52] H. Fu, J.C. Baltazar, D.E. Claridge, Break-Point Statistical Formulations for Whole Building Heating Hot Water Consumption Modeling, in: American Society of Heating, Ventilating, and Air Conditioning Engineers, Kansas City, Missouri, 2019.

[53] X. Li, J.C. Baltazar, Analysis of cooling regression models for hot and humid climates based on" operational effective enthalpy, (2013).

[54] M. Krzywinski, N. Altman, Power and sample size, Nature Methods, 10 (2013) 1139.

[55] J. Cohen, Statistical Power Analysis for the Behavioral Sciences, 2nd ed., LAWRENCE ERLBAUM ASSOCIATES, 1988.

[56] B.A. Turlach, A. Weingessel, C. Moler, quadprog: Functions to Solve Quadratic Programming Problems, in, 2019.

[57] C.-W. Hsu, C.-C. Chang, C.-J. Lin, A practical guide to support vector classification, (2003).

[58] S.F. Crone, J. Guajardo, R. Weber, The impact of preprocessing on support vector regression and neural networks in time series prediction, in: DMIN, 2006, pp. 37-44.

[59] V.N. Vapnik, The nature of statistical learning theory, Springer-Verlag New York, 1995.

[60] P. Tsirikoglou, S. Abraham, F. Contino, C. Lacor, G. Ghorbaniasl, A hyperparameters selection technique for support vector regression models, Applied Soft Computing, 61 (2017) 139-148.

[61] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, Journal of Machine Learning Research, 13 (Feb) (2012) 281-305.

[62] D. Mattera, S. Haykin, Support vector machines for dynamic reconstruction of a chaotic system, in: Advances in kernel methods, MIT Press, 1999, pp. 211-241.

[63] International Performance Measurement & Verification Protocol, Department of Energy, (DOE/GO-102002-1554) (2001).

APPENDIX A. FUNDAMENTALS OF SUPPORT VECTOR MACHINE REGRESSION

A.1 Fundamentals of SVM Regression

SVM regression originally starts by assuming a line that is expressed as below.

$$f(x) = \omega \cdot x + b \qquad \text{(EQ.A 1)}$$

Where ω is a coefficient vector, x is a vector independent of variables, and b is a bias. If the number of independent variables is k, the number of elements of vector ω is also k so that the inner product between ω and x results in a scalar. (Note: ε-SVM regression is a basic type of SVM regression, and in this thesis, only ε-SVM regression is used. Thus, from now, ε-SVM is denoted as simply SVM for simplicity.)
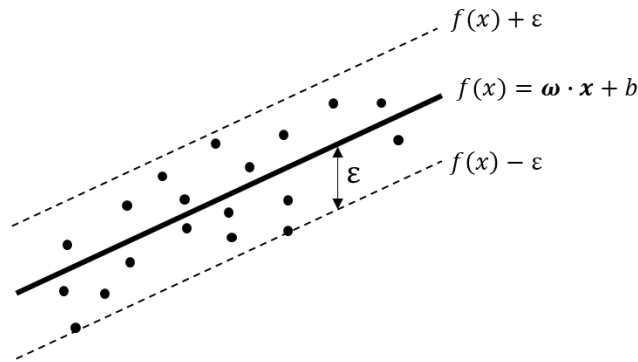
Figure A 1 - Linear Plot with ε

The next step is to find a line in the way that the ε deviation from the line contains all the data points while keeping the plot as flat as possible. The size of ε is defined by a user. (See Figure A2). It is expressed as the following inequality-constrained optimization formula.

$$\text{Min} \frac{1}{2}\|\boldsymbol{\omega}\|^2 \hspace{6cm} \text{(EQ.A 2)}$$

$$\text{Subject to} \begin{cases} y_i - f(\boldsymbol{x_i}) = y_i - (\boldsymbol{\omega} \cdot \boldsymbol{x_i} + b) \leq \varepsilon \\ f(\boldsymbol{x_i}) - y_i = (\boldsymbol{\omega} \cdot \boldsymbol{x_i} + b) - y_i \leq \varepsilon \end{cases}$$

Where the given training dataset is the $(\boldsymbol{x_1}, y_2)$, …, $(\boldsymbol{x_i}, y_i)$, …, $(\boldsymbol{x_n}, y_n)$, $\boldsymbol{x_i}$ is a vector consisting of independent variables of the ith observation, y_i is a scalar of dependent variable of the ith observation, n is the number of training data points, and $f(\boldsymbol{x_i})$ is the output from SVM model when $\boldsymbol{x_i}$ is plugged in. $\|\boldsymbol{\omega}\|^2$ is the square of the norm of the vector $\boldsymbol{\omega}$, and 0.5 is multiplied merely solely for mathematical convenience. (See EQ.A1 for the explanation of the $\boldsymbol{\omega}$ and b.)

The following figures will help to understand the optimization formula intuitively. Figure A2 is the original data, on which SVM models will be trained with different ε.
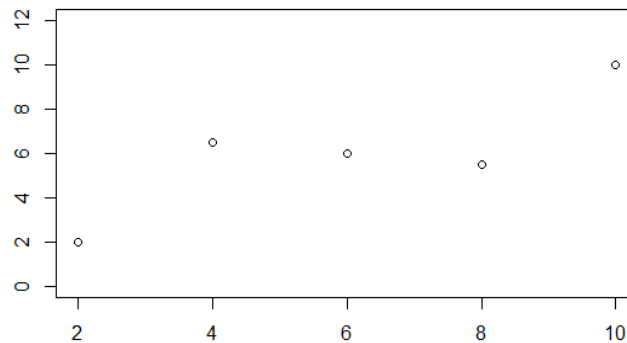
Figure A 2 - Five Sample points

Figure A3 is the SVM regression trained with different values of ε.

(a) $\varepsilon = 2$, $\|\boldsymbol{\omega}\|^2 = 0.245$
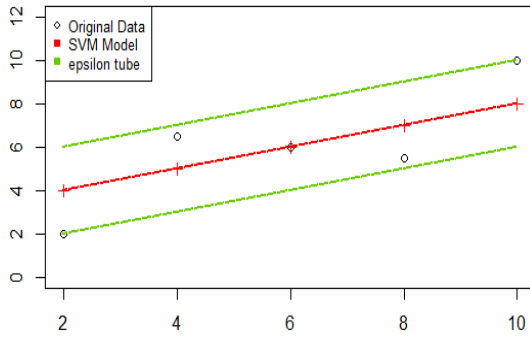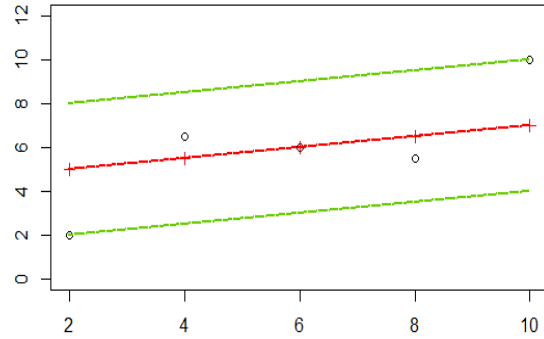
(b) $\varepsilon = 3$, $\|\boldsymbol{\omega}\|^2 = 0.0625$
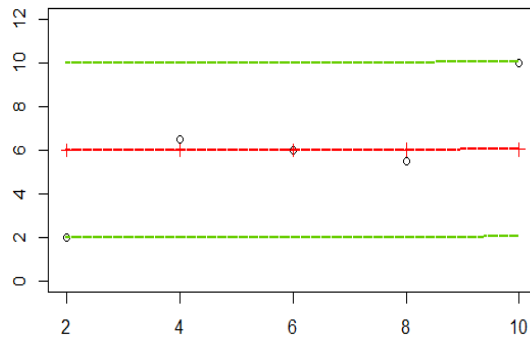
(c) $\varepsilon = 4$, $\|\boldsymbol{\omega}\|^2 = 1.01795$ e-5

Figure A 3 - SVM plots with different ε

As seen in the Figure A3 (a) through (c), as the size of ε becomes larger, the SVM plot becomes flatter while still keeping all data inside of the ε. In other words, What EQ.A2 does is to make the graph as flat as possible as long as all the data can be located within the ε. Another lesson learned here is the importance of selecting a proper ε. If the ε is

larger than needed, the SVM plot will be meaninglessly flat. The question to follow is what if the ε becomes smaller and smaller beyond the point where all the data cannot be located within the ε. The answer is simple. The EQ.A2 becomes unsolvable. In the case of the data of Figure A3, it would be okay even if it cannot be solved with a smaller value of ε because the plot of Figure A3 (a) looks already good enough. However, what if the original data looks like the one in Figure A4? Only one point becomes an outlier, but is there any way to make a reasonable plot while keeping all the data inside of the ε? The answer is No. In that sense, the EQ.A2 fails only with one outlier.
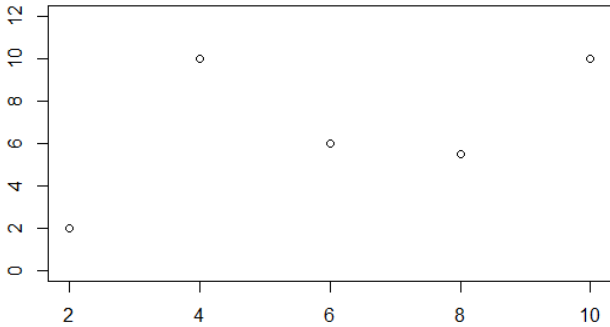
Figure A 4 - Five Sample points with one outlier

To generate a reasonable plot with the outlier, slack variables, ξ_i and ξ_i^* are needed. Slack variables refer to positive variables that are added to inequality constraints of optimization problem to change it to equality constraints. In the context of SVM regression, they are interpreted as variables that fill the gap between ε and the data whose deviation is large than ε. This idea is displayed in the Figure A5.
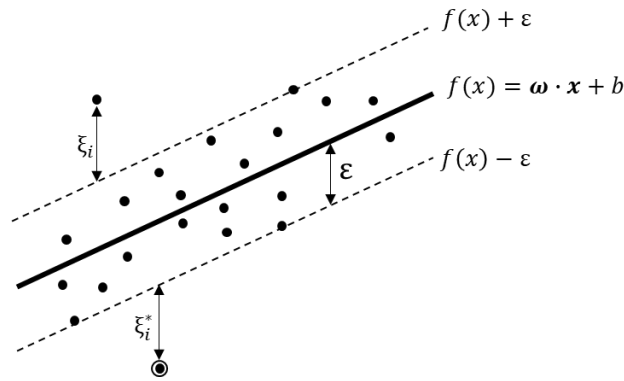
Figure A 5 - Linear Plot with ε, ξ_i and ξ_i^* (SVM plot)

Now, the issue of unsolvable case is resolved due to the use of ξ_i and ξ_i^*. However, another crucial agenda comes up. Being flexible and generous to exceptional cases, which mean the data whose deviation is larger than ε here, does not mean that they are welcomed. Although they are accepted for practical purposes, it is always better to minimize those exceptional cases. Otherwise, SVM will always generate a plot similar to Figure A6.
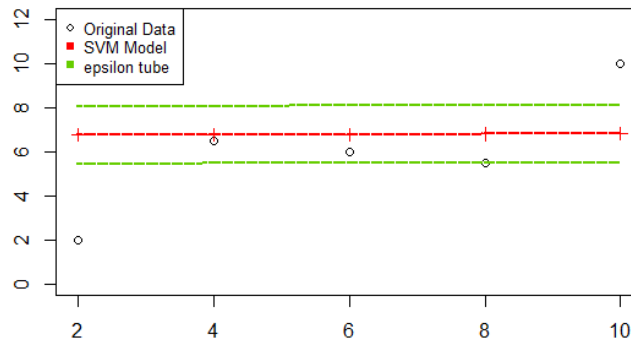
Figure A 6 - SVM plot with nearly no penalty on ξ_i and ξ_i^*

Such a case as Figure A6 happened because the SVM algorithm was not concerned of the size of ξ_i and ξ_i^*, and focused only on minimizing $\|\omega\|^2$. The result of Figure A6 is simply

the mean of the given data, and SVM users may not run SVM just to find the mean.

Another task now becomes clear. That is to minimize the size of ξ_i and ξ_i^* although they

are allowed to exist to cover some outrageous data. In this regard, a term, $\sum_{i=1}^{n}(\xi_i + \xi_i^*)$,

is inserted into the minimization process of EQ.A2. Now, a new optimization task, EQ.A3,

comes up.

$$\text{Min } \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C \sum_{i=1}^{n}(\xi_i + \xi_i^*) \qquad\qquad\qquad \text{(EQ.A 3)}$$

$$\text{Subject to } \begin{cases} \xi_i, \xi_i^* \geq 0 \\ y_i - (\boldsymbol{\omega} \cdot \boldsymbol{x} + b) \leq \varepsilon + \xi_i \\ (\boldsymbol{\omega} \cdot \boldsymbol{x} + b) - y_i \leq \varepsilon + \xi_i^* \end{cases}$$

In the EQ. A3, a new parameter C shows up and is multiplied by $\sum_{i=1}^{n}(\xi_i + \xi_i^*)$.

Conceptually, $min \frac{1}{2}\|\boldsymbol{\omega}\|^2$ and $min \sum_{i=1}^{n}(\xi_i + \xi_i^*)$ can be mutually exclusive. The

former wants to flatten the SVM plot even if the value of $\sum_{i=1}^{n}(\xi_i + \xi_i^*)$ becomes

extremely large. Inversely, the latter is not interested in flattening the plot, rather, it only

wants to minimize $\sum_{i=1}^{n}(\xi_i + \xi_i^*)$. The value of C determines the trade-off between these

two. As in the case with ε, the value of C is also determined by a user. The impact of C

will make more sense with the following Figure A7.

(a) $\varepsilon= 1$, C=0.05   (b) $\varepsilon= 1$, C=2
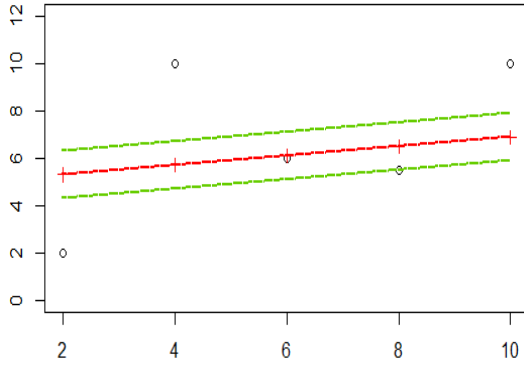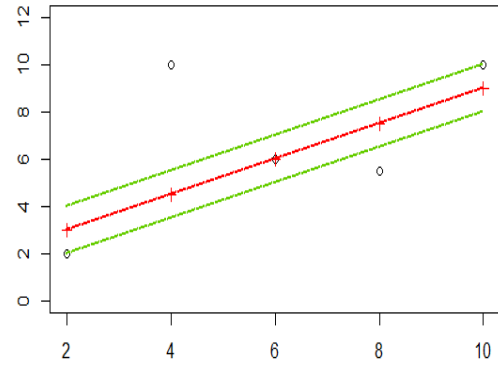
$\|\omega\|^2 = 0.0398, \sum_{i=1}^{5}(\xi_i + \xi_i^*) = 7$ $\|\omega\|^2 = 0.562, \sum_{i=1}^{5}(\xi_i + \xi_i^*) = 5.5$

Figure A 7 - SVM plots with difference C

In the Figure A7, (a) has a lower value C than (b). Seeing the Eq.3, that means (a) imposes less weight on penalizing the slack variables ξ_i and ξ_i^*. Naturally, more weight of minimization is placed upon the term $\|\omega\|^2$. As a result, (a) has a less value of $\|\omega\|^2$ and a more value of $\sum_{i=1}^{n}(\xi_i + \xi_i^*)$ than (b).

The meaning or virtue of 'flatness' of SVM plots may not feel tangible at this point. The significance of the 'flatness' is more well appreciated with non-linear SVM plots. For instance, in the Figure A8, lower value of C was applied to the (b). In other words, more weight was placed on flattening the plot than minimizing the magnitude of slack variables. In the (a) plot, the values of slack variables are nearly zero, and most of the data are located within the ε. On the contrary, bigger number of data were outside of the ε in the (b) plot. Yet the (b) plot is not necessarily worse than (a) plot because it can be seen as a result of focusing more on generalization than on reducing the error aimlessly.

(a) ε= 1.3, C= 8   (b) ε = 1.3, C= 4
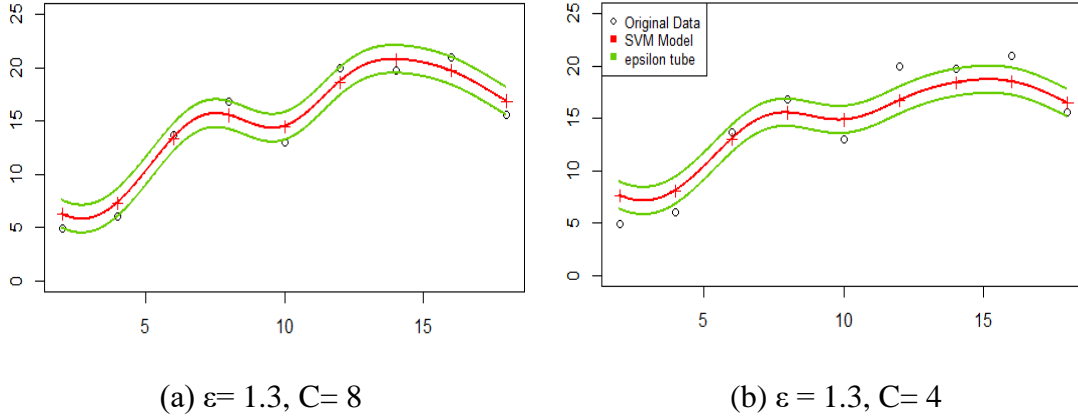
Figure A 8 - Non-linear SVM with different flatness

Eq.A3 can be solved itself, or it can be converted to a dual problem using Lagrange multipliers, and solved in that form. The process of conversion to the dual problem is described as below:

EQ.A3 is an optimization problem subject to constraints. Lagrange function combines the object function and constraints by using Lagrange multipliers as below:

$$L(\boldsymbol{\omega}, b, \xi, \xi^*, \alpha, \alpha^*, \beta, \beta^*)$$

$$= \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) - \sum_{i=1}^{n}\alpha_i\big((\omega \cdot \boldsymbol{x_i} + b) - y_i + \varepsilon + \xi_i\big)$$

$$- \sum_{i=1}^{n}\alpha_i^*\big(-(\omega \cdot \boldsymbol{x_i} + b) + y_i + \varepsilon + \xi_i^*\big) - \sum_{i=1}^{n}(\beta_i\xi_i + \beta_i^*\xi_i^*) \qquad \text{(EQ.A 4)}$$

Where $\alpha_i, \alpha_i^*, \beta_i, \beta_i^*$ are Lagrange multipliers and bound by the following condition:

$$\begin{cases} \alpha_i \geq 0 \\ \alpha_i^* \geq 0 \\ \beta_i \geq 0 \\ \beta_i^* \geq 0 \end{cases} \qquad \text{(EQ.A 5)}$$

For this optimization problem to have a solution, partial derivatives of L with respect to

primal variables, which are $\boldsymbol{\omega}, b, \xi_i, \xi_i^*$, have to be zero

$$\frac{\partial L}{\partial \boldsymbol{\omega}} = \boldsymbol{\omega} - \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)\boldsymbol{x_i} = 0 \qquad\qquad \text{(EQ.A 6)}$$

$$\frac{\partial L}{\partial b} = -\sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0 \qquad\qquad \text{(EQ.A 7)}$$

$$\frac{\partial L}{\partial \xi_i} = C - \alpha_i - \beta_i = 0 \qquad\qquad \text{(EQ.A 8)}$$

$$\frac{\partial L}{\partial \xi_i^*} = C - \alpha_i^* - \beta_i^* = 0 \qquad\qquad \text{(EQ.A 9)}$$

The equation (EQ.A6) can be rearranged in terms of $\boldsymbol{\omega}$ as below:

$$\boldsymbol{\omega} = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)\boldsymbol{x_i} \qquad\qquad \text{(EQ.A 10)}$$

Using EQ.A10, the first term on the right side of EQ.A4 can be arranged as below:

$$\frac{1}{2}\|\boldsymbol{\omega}\|^2 = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\boldsymbol{x_i} \cdot \boldsymbol{x_j} \qquad\qquad \text{(EQ.A 11)}$$

Using EQ.A8 and EQ.A9, the second and the fifth term on the right side of EQ.A4 can

be arranged as below:

$$C\sum_{i=1}^{n}(\xi_i + \xi_i^*) - \sum_{i=1}^{n}(\beta_i\xi_i + \beta_i^*\xi_i^*)$$

$$= \sum_{i=1}^{n}((C\xi_i + C\xi_i^*) - (\beta_i\xi_i + \beta_i^*\xi_i^*))$$

$$= \sum_{i=1}^{n}((C - \beta_i)\xi_i + (C - \beta_i^*)\xi_i^*)$$

$$= \sum_{i=1}^{n}(\alpha_i\xi_i + \alpha_i^*\xi_i^*) \qquad\qquad \text{(EQ.A 12)}$$

Using EQ.A7 and EQ.A10, the third and the fourth term on the right side of EQ.A4 can be arranged as below:

$$-\sum_{i=1}^{n}\alpha_i\big((\boldsymbol{\omega}\cdot\boldsymbol{x_i}+b)-y_i+\varepsilon+\xi_i\big)-\sum_{i=1}^{n}\alpha_i^*(-(\boldsymbol{\omega}\cdot\boldsymbol{x_i}+b)+y_i+\varepsilon+\xi_i^*)$$

$$=-\sum_{i=1}^{n}(\alpha_i-\alpha_i^*)\boldsymbol{\omega}\cdot\boldsymbol{x_i}-\sum_{i=1}^{n}(\alpha_i-\alpha_i^*)b+\sum_{i=1}^{n}(\alpha_i-\alpha_i^*)y_i$$

$$-\sum_{i=1}^{n}(\alpha_i+\alpha_i^*)\varepsilon-\sum_{i=1}^{n}(\alpha_i\xi_i+\alpha_i^*\xi_i^*)$$

$$=-\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)\boldsymbol{x_i}\cdot\boldsymbol{x_j}+\sum_{i=1}^{n}(\alpha_i-\alpha_i^*)y_i$$

$$-\sum_{i=1}^{n}(\alpha_i+\alpha_i^*)\varepsilon-\sum_{i=1}^{n}(\alpha_i\xi_i+\alpha_i^*\xi_i^*) \qquad\text{(EQ.A 13)}$$

Inserting EQ.A11, EQ.A12, EQ.A13 into EQ.A4, Lagrange dual function is derived in terms of Lagrange multipliers α_i and α_i^*, and the Lagrange dual problem is defined as below:

$$Max\ L(\alpha,\alpha^*)=-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)\boldsymbol{x_i}\cdot\boldsymbol{x_j}$$

$$+\sum_{i=1}^{n}(\alpha_i-\alpha_i^*)y_i-\sum_{i=1}^{n}(\alpha_i+\alpha_i^*)\varepsilon \qquad\text{(EQ.A 14)}$$

Subject to

$$\begin{cases}\sum_{i=1}^{n}(\alpha_i-\alpha_i^*)=0\\ 0\le\alpha_i\le C\\ 0\le\alpha_i^*\le C\end{cases}$$

EQ.A3 is called a primal form and EQ.A14 is called a dual form of SVM optimization formula. One cannot simply say one is always better than the other. Yet, the dual form is preferred when non-linear SVM is used due to the ease of using kernel function. For instance, EQ.A14 changes to non-linear SVM simply by replacing $\boldsymbol{x_i}\cdot\boldsymbol{x_j}$ with $K(\boldsymbol{x_i}\cdot\boldsymbol{x_j})$ (See Eq.A15). K is defined as $\varphi(\boldsymbol{x_i})^T\varphi(\boldsymbol{x_j})$ where φ is a mapping function that transfer

data to higher dimensions. The data is regressed in the different dimensions. (See Figure A9).
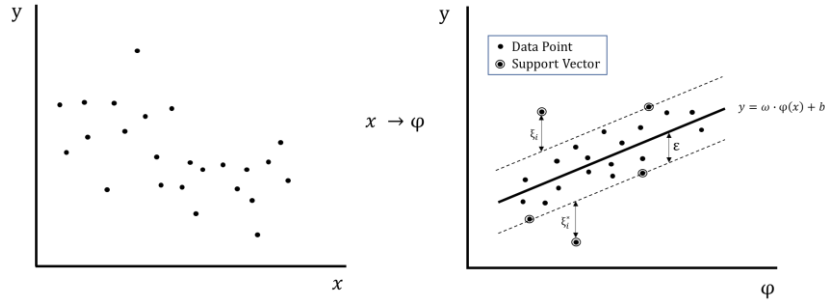
Figure A 9 - mapping of non-linear data into a feature space.

The explicit form of $\varphi(x)$ is not necessary here. Only the inner product between two mapped functions, which is K, is needed to turn the linear SVM to non-linear SVM.

$$Max\ L(\alpha, \alpha^*) = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)K(x_i \cdot x_j)$$

$$+ \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)y_i - \sum_{i=1}^{n}(\alpha_i + \alpha_i^*)\varepsilon \qquad \text{(EQ.A 15)}$$

Subject to

$$\sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0\ and\ 0 \le \alpha_i, \alpha_i^* \le C$$

Radial Basis Function is the most representative non-linear kernel of SVM. It takes the following form:

$$K(x_i \cdot x_j) = e^{-\gamma\|x_i - x_j\|^2} \qquad \text{(EQ.A 16)}$$

EQ.A3, EQ.A14, and EQ.A15 are a quadratic optimization problem with inequality constraints. The basic mathematical format of such a task can be expressed as EQ.A17.

$$Min \; b^T D b - d^T b \qquad\qquad\qquad\qquad\qquad \text{(EQ.A 17)}$$

Subject to

$$\begin{cases} A_1{}^T b = b_1 \\ A_2{}^T b \geq b_2 \end{cases}$$

The EQ.A3 can be expressed as a matrix form befitting the EQ.A17. In that case, each

term of the EQ.A17 has components as below:

(n: the number of observations, k: the number of independent variables)

$$b = \begin{pmatrix} b \\ \omega_1 \\ \vdots \\ \omega_k \\ \xi_1 \\ \vdots \\ \xi_n \\ \xi_1^* \\ \vdots \\ \xi_n^* \end{pmatrix}, d = \begin{pmatrix} 0 \\ a_1 = 0 \\ \vdots \\ a_k = 0 \\ b_1 = -C \\ \vdots \\ b_{2n} = -C \end{pmatrix}$$

$$A_2 = \begin{pmatrix} a_1 = 0 & \cdots & a_{2n} = 0 & b_1 = 1 & \cdots & b_n = 1 & b_{n+1} = -1 & \cdots & b_{2n} = -1 \\ c_{1,1} = 0 & \cdots & 0 & d_{1,1} = x_{1,1} & & d_{1,n} = x_{n,1} & d_{1,n+1} = -x_{1,1} & & d_{1,2n} = -x_{n,1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & c_{k,2n} = 0 & d_{k,1} = x_{1,k} & & d_{k,n} = x_{n,k} & d_{k,n+1} = -x_{1,k} & 0 & d_{k,2n} = -x_{n,k} \\ e_{1,1} = 1 & \cdots & 0 & f_{1,1} = 1 & 0 & & \cdots & & 0 \\ 0 & 1 & & \vdots & 0 & 1 & & & \\ \vdots & \ddots & & & & & \ddots & & \cdots \\ & & 1 & 0 & & & \ddots & 1 & 0 \\ 0 & & 0 & e_{2n,2n} = 1 & 0 & & \cdots & 0 & f_{2n,2n} = 1 \end{pmatrix}$$

$$b_2 = (a_1 = 0 \quad \cdots \quad a_{2n} = 0 \quad b_1 = y_1 - \varepsilon \quad \cdots \quad b_n = y_n - \varepsilon \quad c_1 = -y_n - \varepsilon \quad \cdots \quad c_n = -y_n - \varepsilon)$$

$$D = \begin{pmatrix} 0 & 0 & 0 & \cdots & & & & \\ 0 & a_{1,1} = 1 & 0 & & & & & \\ 0 & 0 & \ddots & & & & & \\ \vdots & & & a_{k,k} = 1 & & & & \\ & & & & b_{1,1} = 0 & 0 & & 0 \\ & & & & 0 & \ddots & & 0 \\ & & \cdots & & 0 & 0 & b_{2n,2n} = 0 \end{pmatrix}$$

Once the quadratic optimization problem is solved, the components of the vector b,

which are b, ω_i, ..., ω_k, ξ_1, ..., ξ_n, ξ_1^*, ..., ξ_n^*, are obtained.

The EQ.A14 and EQ.A15 can be expressed as a matrix form befitting the EQ.A17. In

that case, each term of the EQ.A17 has components is shown as below. (Note that the

original optimization task is maximization, but here it changes to minimization by

multiplying -1 by the original formula to fit EQ.A17.)

(n: the number of observations, k: the number of independent variables)

$$b = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \\ \alpha_1^* \\ \vdots \\ \alpha_n^* \end{pmatrix}, d = \begin{pmatrix} y_1 - \varepsilon \\ \vdots \\ y_n - \varepsilon \\ -y_1 - \varepsilon \\ \vdots \\ -y_n - \varepsilon \end{pmatrix}$$

$$A_1 = \begin{pmatrix} a_1 = 1 \\ \vdots \\ a_n = 1 \\ b_1 = -1 \\ \vdots \\ b_n = -1 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} a_{1,1} = 1 & 0 & 0 & 0 & 0 & b_{1,1} = -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & a_{2n,2n} = 1 & 0 & 0 & 0 & 0 & b_{2n,2n} = -1 \end{pmatrix}$$

$$b_1 = (0), b_2 = (a_1 = 0 \quad \cdots \quad a_{2n} = 0 \quad a_{2n+1} = -C \quad \cdots \quad a_{4n} = -C)$$

$$D = \begin{pmatrix} K & -K \\ -K & K \end{pmatrix}$$

Where:

$$K = \begin{pmatrix} K(x_1 \cdot x_1) & K(x_1 \cdot x_2) & \cdots & \\ K(x_2 \cdot x_1) & & & \\ \vdots & & \ddots & \\ & & \cdots & K(x_n \cdot x_{n-1}) & K(x_{n-1} \cdot x_n) \\ & & & K(x_n \cdot x_{n-1}) & K(x_n \cdot x_n) \end{pmatrix}$$

Refer to Appendix B for details of $K(x, y)$

Once the quadratic optimization problem is solved, the components of the vector b,

which are α_1, …, α_n, α_1^*, …, α_n^*, are obtained.

EQ.A3, EQ.A14, and EQ.A15 are inequality constrained optimization problems. The

local optimum of those problems is supposed to satisfy Karush-Kuhn-Tucker (KKT)

conditions. The KKT conditions are not sufficient conditions, but necessary conditions

by themselves. However, if the objective function, which is an object of minimization or

maximization, is convex or concave, it becomes sufficient conditions conveniently. It is

because a local minimum is the unique global minimum in the convex. The objective

function of EQ.3, EQ.9, and EQ.11 are convex. Hence, the KKT conditions can be used

to find the optimal solution to SVM optimization problems. The KKT conditions are

summarized below:

KKT Conditions

(The asterisk (*) indicates optimal values. $f(x)$ is an objective function, $g_i(x)$ is

inequality constraints, and $h_i(x)$ is equality constraints.)

Let's suppose there is an optimization task as below:

Min $f(x)$

Subject to

$g_i(x^*) \geq 0, i = 1, \dots, m$

$h_i(x^*) = 0, i = 1, \dots, k$

If x^* is optimum, it must satisfy the following conditions.

1. Feasible Constraints

$$g_i(\boldsymbol{x}^*) \geq 0, i = 1, \dots, m$$

$$h_i(\boldsymbol{x}^*) = 0, i = 1, \dots, k$$

The optimal value of $\boldsymbol{x}^*$ must keep the constraint conditions.

2. Stationary

$$\nabla f(\boldsymbol{x}^*) - \sum_{i=1}^{m} \mu_i \nabla g_i(\boldsymbol{x}^*) - \sum_{i=1}^{k} \lambda_i \nabla h_i(\boldsymbol{x}^*) = 0$$

There is no feasible direction in which the objective function can be furthered optimized.

3, Complementary Slackness

$$\mu_i g_i(\boldsymbol{x}^*) = 0$$

When the constraint is binding ($g_i(\boldsymbol{x}^*) = 0$) the Lagrange multiplier for inequality

constraints (μ_i) is positive. When the constraint is not binding ($g_i(\boldsymbol{x}^*) > 0$), the

Lagrange multiplier (μ_i) for inequality constraints is zero. Note that Lagrange multiplier

for equality constraints (λ_i) is free from this constraint.

4. Positive Lagrange Multipliers (for inequality constraints)

$$\mu_i \geq 0$$

By applying KKT conditions to the inequality-constraints optimization task, a set of

simultaneous equations are established. In general, it starts by assuming that all

constraints are binding ($g_i(\boldsymbol{x}^*) = 0 \ and \ \mu_i > 0$). If the solution from this assumption

violates any of the KKT conditions, the assumption should change and the equation

should be solved again until no violation of the KKT conditions occurs. For example, if

the resultant μ_2 is negative, it violates the KKT condition 4, and it should be assumed

that $\mu_2 = 0$ and another attempt should be made to solve those equations. This process is

repeated until the one that satisfies all KKT conditions is found. In the convex

optimization problem, the solution of the set of simultaneous equation is the global

optimum. Unfortunately, in SVM, the number of equations to be solved simultaneously

is overwhelming. For instance, even if the number of training data is just two, there are

14 equations to be solved. (An example follows to show this.) Thus, in SVM, this

equation is solved using numerical methods, such as Interior Point, Active Set, and SMO

(sequential minimal optimization), and so on.

The resultant plot from EQ.A14 and EQ.A15 are expressed as EQ.A18 and EQ.A19,

respectively.

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)x_i \cdot x + b \qquad\qquad \text{(EQ.A 18)}$$

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)k(x_i \cdot x) + b \qquad\qquad \text{(EQ.A 19)}$$

n is the number of training data, and i refers to an index for each observation. The

observation where $\alpha_i - \alpha_i^* \neq 0$ is called a support vector. Thus, the observation where

$\alpha_i - \alpha_i^* = 0$ is not a support vector. In other word, the plot of EQ.A18 and EQ.A19 can

be expressed only by support vectors. For this reason, n sometimes refers to the number

of support vector, not the number of training data. If that is the case, i refers to an index

for support vector. In the case of non-linear kernel SVM regression, the resultant plot

EQ.A19 cannot be reversed to the linear form of EQ.A1. Yet, it is possible for the

EQ.A18 to be reversed to EQ.A1 by using EQ. A10.

$$\boldsymbol{\omega} = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\boldsymbol{x_i} \qquad \qquad \text{(EQ.A 20)}$$

The bias term b is obtained by applying the KKT condition 3 to the primal form of SVM.

$$\alpha_i(\boldsymbol{x_i} \cdot \boldsymbol{\omega} + b + \xi_i + \varepsilon - y_i) = 0$$

$$\alpha_i^*(-\boldsymbol{x_i} \cdot \boldsymbol{\omega} - b + \xi_i^* + \varepsilon + y_i) = 0$$

The value of b can be known when the Lagrange multiplies α_i or α_i^* are nonzero, and the corresponding slack variables ξ_i or ξ_i^* are zero. This happens when the support vectors that sit exactly along the epsilon line. In this case, $0 < \alpha_i \text{ or } \alpha_i^* < C$. The support vector of this location is called unbounded (or free) support vector. For non-support vectors, both α_i and α_i^* become zero, and the righthand side of the KKT condition 3 becomes an unidentified non-zero value. For support vectors that sit away from the epsilon line, α_i or α_i^* becomes C, but the slack variable becomes non-zero. Since the value of slack variable is unknown at this point, the KKT condition 3 cannot be used here. For unbounded support vectors, the b can be found as below.

$$b = -\varepsilon + y_i - \boldsymbol{x_i} \cdot \boldsymbol{\omega} \text{ for } 0 < \alpha_i < C$$

$$b = \varepsilon + y_i - \boldsymbol{x_i} \cdot \boldsymbol{\omega} \text{ for } < 0 < \alpha_i^* < C$$

However, the equation above can be used only to linear SVM regression. For non-linear Kernel SVM regression, the following equation can be used instead.

$$b = -\varepsilon + y_i - \sum_{j=1}^{n}(\alpha_j - \alpha_j^*)\,k(\boldsymbol{x_j} \cdot \boldsymbol{x_i}) \text{ for } 0 < \alpha_i < C$$

$$b = \varepsilon + y_i - \sum_{j=1}^{n}(\alpha_j - \alpha_j^*)\,k(\boldsymbol{x_j} \cdot \boldsymbol{x_i}) \text{ for } < 0 < \alpha_i^* < C$$

The $x_i \cdot \omega$ term of linear SVM is functionally equivalent to the term $\sum_{j=1}^{n}(\alpha_j - \alpha_j^*) k(x_j \cdot x_i)$ of non-linear kernel SVM regression. (See EQ.A1 and EQ.A19).

A.2 An Example of Solving the Primal Form of SVM Regression

A.2.1 Establishment of Simultaneous Equations

Here is an example to set up a system of equations applying KKT conditions to SVM

task that has only three points to train. The training data is assumed to be two-

dimensional: (x1,y1), (x2,y2), (x3,y3)

Optimization task:

Minimize $f(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{3}(\xi_i + \xi_i^*)$

subject to
$$
\begin{cases}
g_1(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = \xi_1 \geq 0 \\
g_2(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = \xi_2 \geq 0 \\
g_3(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = \xi_3 \geq 0 \\
g_4(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = \xi_1^* \geq 0 \\
g_5(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = \xi_2^* \geq 0 \\
g_6(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = \xi_3^* \geq 0 \\
g_7(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = x_1\omega + b + \xi_1 - y_1 + \varepsilon \geq 0 \\
g_8(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = x_2\omega + b + \xi_2 - y_2 + \varepsilon \geq 0 \\
g_9(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = x_3\omega + b + \xi_3 - y_3 + \varepsilon \geq 0 \\
g_{10}(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = -x_1\omega - b + \xi_1^* + y_1 + \varepsilon \geq 0 \\
g_{11}(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = -x_2\omega - b + \xi_2^* + y_2 + \varepsilon \geq 0 \\
g_{12}(b, \omega, \xi_1, \xi_2, \xi_3, \xi_1^*, \xi_2^*, \xi_3^*) = -x_3\omega - b + \xi_3^* + y_3 + \varepsilon \geq 0
\end{cases}
$$

This optimization task is equivalent to the following matrix algebra:

$$
Min(b \quad \omega \quad \xi_1 \quad \xi_2 \quad \xi_3 \quad \xi_1^* \quad \xi_2^* \quad \xi_3^*)
\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
\begin{pmatrix}
b \\ \omega \\ \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_1^* \\ \xi_2^* \\ \xi_3^*
\end{pmatrix} +
$$

$$
(0 \quad 0 \quad C \quad C \quad C \quad C \quad C \quad C)
\begin{pmatrix}
b \\ \omega \\ \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_1^* \\ \xi_2^* \\ \xi_3^*
\end{pmatrix}
$$

Subject to

$$
\begin{pmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & x_1 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & x_2 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & x_3 & 0 & 0 & 1 & 0 & 0 & 0 \\
-1 & -x_1 & 0 & 0 & 0 & 1 & 0 & 0 \\
-1 & -x_2 & 0 & 0 & 0 & 0 & 1 & 0 \\
-1 & -x_3 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
b \\ \omega \\ \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_1^* \\ \xi_2^* \\ \xi_3^*
\end{pmatrix}
$$

$$
\geq (0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad y_1 - \varepsilon \quad y_2 - \varepsilon \quad y_3 - \varepsilon \quad -y_1 - \varepsilon \quad -y_2 - \varepsilon \quad -y_3 - \varepsilon)
$$

A system of equations can be set up using KKT conditions.

KKT condition 2 (Stationary):

$$
\nabla f(x^*) - \sum_{i=1}^{m} \mu_i \nabla g_i(x^*) - \sum_{i=1}^{k} \lambda_i \nabla h_i(x^*) = 0
$$

In this example,

$$
x^T = (b \quad \omega \quad \xi_1 \quad \xi_2 \quad \xi_3 \quad \xi_1^* \quad \xi_2^* \quad \xi_3^*), m = 12, k = 0
$$

To be consistent with the Lagrange function of Eq.4, the Lagrange multipliers for inequality will be denoted as below:

$$\mu_i = \beta_i \ (i = 1, 2, 3), \mu_i = \beta_{i-3}^* \ (i = 4, 5, 6)$$

$$\mu_{i-6} = \alpha_i \ (i = 7, 8, 9), \qquad \mu_i = \alpha_{i-9}^* \ (i = 10, 11, 12)$$

Each term of the equation can be expressed in the form of vector as below:

$$\nabla f(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial f}{\partial b} \\ \dfrac{\partial f}{\partial \omega} \\ \dfrac{\partial f}{\partial \xi_1} \\ \dfrac{\partial f}{\partial \xi_2} \\ \dfrac{\partial f}{\partial \xi_3} \\ \dfrac{\partial f}{\partial \xi_1^*} \\ \dfrac{\partial f}{\partial \xi_2^*} \\ \dfrac{\partial f}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ \omega \\ C \\ C \\ C \\ C \\ C \\ C \end{pmatrix}, \nabla g_1(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_1}{\partial b} \\ \dfrac{\partial g_1}{\partial \omega} \\ \dfrac{\partial g_1}{\partial \xi_1} \\ \dfrac{\partial g_1}{\partial \xi_2} \\ \dfrac{\partial g_1}{\partial \xi_3} \\ \dfrac{\partial g_1}{\partial \xi_1^*} \\ \dfrac{\partial g_1}{\partial \xi_2^*} \\ \dfrac{\partial g_1}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \nabla g_2(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_2}{\partial b} \\ \dfrac{\partial g_2}{\partial \omega} \\ \dfrac{\partial g_2}{\partial \xi_1} \\ \dfrac{\partial g_2}{\partial \xi_2} \\ \dfrac{\partial g_2}{\partial \xi_3} \\ \dfrac{\partial g_2}{\partial \xi_1^*} \\ \dfrac{\partial g_2}{\partial \xi_2^*} \\ \dfrac{\partial g_2}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\nabla g_3(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_3}{\partial b} \\ \dfrac{\partial g_3}{\partial \omega} \\ \dfrac{\partial g_3}{\partial \xi_1} \\ \dfrac{\partial g_3}{\partial \xi_2} \\ \dfrac{\partial g_3}{\partial \xi_3} \\ \dfrac{\partial g_3}{\partial \xi_1^*} \\ \dfrac{\partial g_3}{\partial \xi_2^*} \\ \dfrac{\partial g_3}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \nabla g_4(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_4}{\partial b} \\ \dfrac{\partial g_4}{\partial \omega} \\ \dfrac{\partial g_4}{\partial \xi_1} \\ \dfrac{\partial g_4}{\partial \xi_2} \\ \dfrac{\partial g_4}{\partial \xi_3} \\ \dfrac{\partial g_4}{\partial \xi_1^*} \\ \dfrac{\partial g_4}{\partial \xi_2^*} \\ \dfrac{\partial g_4}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \nabla g_5(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_5}{\partial b} \\ \dfrac{\partial g_5}{\partial \omega} \\ \dfrac{\partial g_5}{\partial \xi_1} \\ \dfrac{\partial g_5}{\partial \xi_2} \\ \dfrac{\partial g_5}{\partial \xi_3} \\ \dfrac{\partial g_5}{\partial \xi_1^*} \\ \dfrac{\partial g_5}{\partial \xi_2^*} \\ \dfrac{\partial g_5}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$\nabla g_6(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_6}{\partial b} \\[4pt] \dfrac{\partial g_6}{\partial \omega} \\[4pt] \dfrac{\partial g_6}{\partial \xi_1} \\[4pt] \dfrac{\partial g_6}{\partial \xi_2} \\[4pt] \dfrac{\partial g_6}{\partial \xi_3} \\[4pt] \dfrac{\partial g_6}{\partial \xi_1^*} \\[4pt] \dfrac{\partial g_6}{\partial \xi_2^*} \\[4pt] \dfrac{\partial g_6}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \nabla g_7(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_7}{\partial b} \\[4pt] \dfrac{\partial g_7}{\partial \omega} \\[4pt] \dfrac{\partial g_7}{\partial \xi_1} \\[4pt] \dfrac{\partial g_7}{\partial \xi_2} \\[4pt] \dfrac{\partial g_7}{\partial \xi_3} \\[4pt] \dfrac{\partial g_7}{\partial \xi_1^*} \\[4pt] \dfrac{\partial g_7}{\partial \xi_2^*} \\[4pt] \dfrac{\partial g_7}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} 1 \\ x_1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \nabla g_8(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_8}{\partial b} \\[4pt] \dfrac{\partial g_8}{\partial \omega} \\[4pt] \dfrac{\partial g_8}{\partial \xi_1} \\[4pt] \dfrac{\partial g_8}{\partial \xi_2} \\[4pt] \dfrac{\partial g_8}{\partial \xi_3} \\[4pt] \dfrac{\partial g_8}{\partial \xi_1^*} \\[4pt] \dfrac{\partial g_8}{\partial \xi_2^*} \\[4pt] \dfrac{\partial g_8}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} 1 \\ x_2 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\nabla g_9(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_9}{\partial b} \\[4pt] \dfrac{\partial g_9}{\partial \omega} \\[4pt] \dfrac{\partial g_9}{\partial \xi_1} \\[4pt] \dfrac{\partial g_9}{\partial \xi_2} \\[4pt] \dfrac{\partial g_9}{\partial \xi_3} \\[4pt] \dfrac{\partial g_9}{\partial \xi_1^*} \\[4pt] \dfrac{\partial g_9}{\partial \xi_2^*} \\[4pt] \dfrac{\partial g_9}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} 1 \\ x_3 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \nabla g_{10}(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_{10}}{\partial b} \\[4pt] \dfrac{\partial g_{10}}{\partial \omega} \\[4pt] \dfrac{\partial g_{10}}{\partial \xi_1} \\[4pt] \dfrac{\partial g_{10}}{\partial \xi_2} \\[4pt] \dfrac{\partial g_{10}}{\partial \xi_3} \\[4pt] \dfrac{\partial g_{10}}{\partial \xi_1^*} \\[4pt] \dfrac{\partial g_{10}}{\partial \xi_2^*} \\[4pt] \dfrac{\partial g_{10}}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} -1 \\ -x_1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \nabla g_{11}(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_{11}}{\partial b} \\[4pt] \dfrac{\partial g_{11}}{\partial \omega} \\[4pt] \dfrac{\partial g_{11}}{\partial \xi_1} \\[4pt] \dfrac{\partial g_{11}}{\partial \xi_2} \\[4pt] \dfrac{\partial g_{11}}{\partial \xi_3} \\[4pt] \dfrac{\partial g_{11}}{\partial \xi_1^*} \\[4pt] \dfrac{\partial g_{11}}{\partial \xi_2^*} \\[4pt] \dfrac{\partial g_{11}}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} -1 \\ -x_2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$\nabla g_{12}(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_{12}}{\partial b} \\[2mm] \dfrac{\partial g_{12}}{\partial \omega} \\[2mm] \dfrac{\partial g_{12}}{\partial \xi_1} \\[2mm] \dfrac{\partial g_{12}}{\partial \xi_2} \\[2mm] \dfrac{\partial g_{12}}{\partial \xi_3} \\[2mm] \dfrac{\partial g_{12}}{\partial \xi_1^*} \\[2mm] \dfrac{\partial g_{12}}{\partial \xi_2^*} \\[2mm] \dfrac{\partial g_{12}}{\partial \xi_3^*} \end{pmatrix} = \begin{pmatrix} -1 \\ -x_3 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Plugging in the corresponding vectors to the equation, eight equations are derived from the six rows, respectively.

$$-\alpha_1 - \alpha_2 - \alpha_3 + \alpha_1^* + \alpha_2^* + \alpha_3^* = 0$$

$$\omega - x_1\alpha_1 - x_2\alpha_2 - x_3\alpha_3 + x_1\alpha_1^* + x_2\alpha_2^* + x_3\alpha_3^* = 0$$

$$C - \beta_1 - \alpha_1 = 0$$

$$C - \beta_2 - \alpha_2 = 0$$

$$C - \beta_3 - \alpha_3 = 0$$

$$C - \beta_1^* - \alpha_1^* = 0$$

$$C - \beta_2^* - \alpha_2^* = 0$$

$$C - \beta_3^* - \alpha_3^* = 0$$

KKT condition 3

$$\mu_i g_i(\boldsymbol{x}^*) = 0$$

The following twelve equations are set up as a result of applying the KKT condition 3

$$\mu_1 g_1(x^*) = \beta_1 \xi_1 = 0$$

$$\mu_2 g_2(x^*) = \beta_2 \xi_2 = 0$$

$$\mu_3 g_3(x^*) = \beta_3 \xi_3 = 0$$

$$\mu_4 g_4(x^*) = \beta_1^* \xi_1^* = 0$$

$$\mu_5 g_5(x^*) = \beta_2^* \xi_2^* = 0$$

$$\mu_6 g_6(x^*) = \beta_3^* \xi_3^* = 0$$

$$\mu_7 g_7(x^*) = \alpha_1(x_1 \omega + b + \xi_1 + \varepsilon - y_1) = 0$$

$$\mu_8 g_8(x^*) = \alpha_2(x_2 \omega + b + \xi_2 + \varepsilon - y_2) = 0$$

$$\mu_9 g_9(x^*) = \alpha_3(x_3 \omega + b + \xi_3 + \varepsilon - y_3) = 0$$

$$\mu_{10} g_{10}(x^*) = \alpha_1^*(-x_1 \omega - b + \xi_1^* + \varepsilon + y_1) = 0$$

$$\mu_{11} g_{11}(x^*) = \alpha_2^*(-x_2 \omega - b + \xi_2^* + \varepsilon + y_2) = 0$$

$$\mu_{i2} g_{12}(x^*) = \alpha_3^*(-x_3 \omega - b + \xi_3^* + \varepsilon + y_2) = 0$$

Solving the 20 equations simultaneously, the components of x^*, which are b, ω, ξ_1, ξ_2, ξ_3, ξ_1^*, ξ_2^*, and ξ_3^*, are identified.

However, solving the system of 20 equations by hand is extremely difficult and time-consuming. Thus, the solution to this system of equations is calculated with sample training data and predetermined hyperparameters by the quadratic programming solver, Quadprog [56]. The solution obtained from the Quadprog is checked with the KKT

conditions 2 and 3. Since the simultaneous equations are from the KKT conditions, if the solution satisfies all KKT conditions, it will automatically all the simultaneous equations.

A.2.2 Solving SVM Optimization Task Using a Quadratic Solver

The input and output of the Quadprog are as follows:

(Input)

Hyperparameters: $C = 4$, $\varepsilon = 0.1$

Training Data: $(x1,y1) = (1,2)$, $(x2,y2) = (2,4)$, $(x3,y3) = (3,5)$
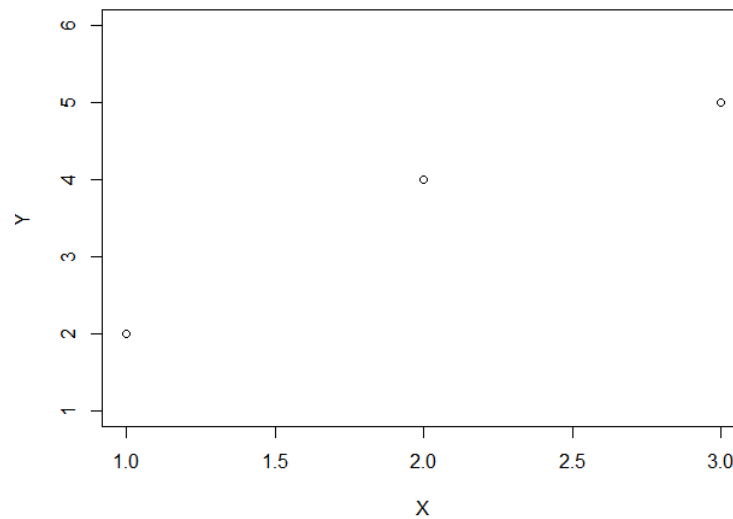
Figure A 10 - Three Sample Points

(Output from the Quadprog)

Solution: $x^T = (b \quad \omega \quad \xi_1 \quad \xi_2 \quad \xi_3 \quad \xi_1^* \quad \xi_2^* \quad \xi_3^*) =$

$(0.6 \quad 1.5 \quad 0 \quad 0.3 \quad 0 \quad 0 \quad 0 \quad 0)$

Lagrange Multipliers:

$\alpha_1 = 0, \alpha_2 = 4, \alpha_3 = 0, \alpha_1^* = 2.75, \alpha_2^* = 0, \alpha_3^* = 1.25$

$\beta_1 = 4, \beta_2 = 0, \beta_3 = 4, \beta_1^* = 1.25, \beta_2^* = 4, \beta_3^* = 2.75$

The resultant plot is:

$f(x) = \omega x + b = 1.5x + 0.6$
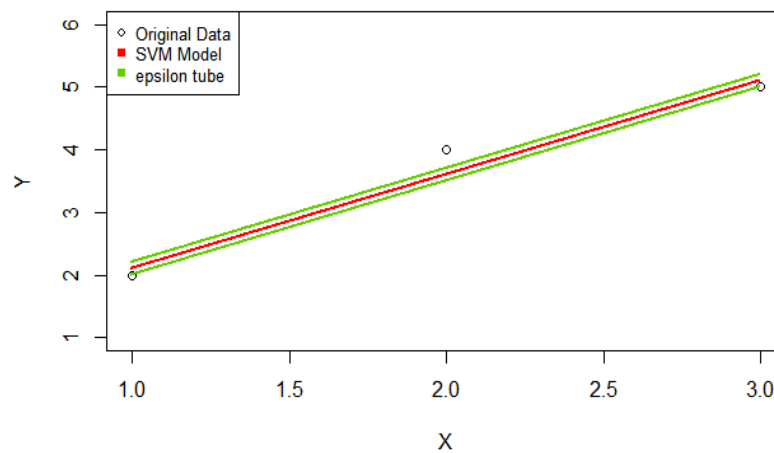
Figure A 11 - Linear (Primal) SVM from the Three Sample Points

A.2.3 Validation of the Solution in Reference to KKT Conditions

The output from the Quadprog is validated by checking if the output satisfies all KKT conditions.

1. Feasible Constraints

$g_i(x^*) \geq 0, h_i(x^*) = 0$

$g_1 = \xi_1 = 0 \geq , g_2 = \xi_2 = 0.3 \geq 0, g_3 = \xi_3 = 0 \geq 0, g_4 = \xi_1^* = 0 \geq 0$

$g_5 = \xi_2^* = 0 \geq 0, g_6 = \xi_3^* = 0 \geq 0$

$g_7 = x_1\omega + b + \xi_1 + \varepsilon - y_1 = 1 \times 1.5 + 0.6 + 0 + 0.1 - 2 \geq 0$

$g_8 = x_2\omega + b + \xi_2 + \varepsilon - y_2 = 2 \times 1.5 + 0.6 + 0.3 + 0.1 - 4 \geq 0$

$g_9 = x_3\omega + b + \xi_3 + \varepsilon - y_3 = 3 \times 1.5 + 0.6 + 0 + 0.1 - 5 \geq 0$

$g_{10} = -x_1\omega - b + \xi_1^* + \varepsilon + y_1 = -1 \times 1.5 - 0.6 + 0 + 0.1 + 2 \geq 0$

$g_{11} = -x_2\omega - b + \xi_2^* + \varepsilon + y_2 = -2 \times 1.5 - 0.6 + 0 + 0.1 + 4 \geq 0$

$g_{12} = -x_3\omega - b + \xi_3^* + \varepsilon + y_3 = -3 \times 1.5 - 0.6 + 0 + 0.1 + 5 \geq 0$

2. Stationary

$$\nabla f(x^*) - \sum_{i=1}^{m} \mu_i \nabla g_i(x^*) - \sum_{i=1}^{k} \lambda_i \nabla h_i(x^*) = 0$$

Each row of the vector equation makes one scalar equation. Thus, scalar six equations are established as below. Plugging in the Lagrange Multipliers to each of corresponding equations, you see that all equations are satisfied.

$$-\alpha_1 - \alpha_2 - \alpha_3 + \alpha_1^* + \alpha_2^* + \alpha_3^* = -0 - 4 - 0 + 2.75 + 0 + 1.25 = 0$$

$$\omega - x_1\alpha_1 - x_2\alpha_2 - x_3\alpha_3 + x_1\alpha_1^* + x_2\alpha_2^* + x_3\alpha_3^*$$

$$= 1.5 - 1 \times 0 - 2 \times 4 - 3 \times 0 + 1 \times 2.75 + 2 \times 0 + 3 \times 1.25 = 0$$

$$C - \beta_1 - \alpha_1 = 4 - 4 - 0 = 0$$

$$C - \beta_2 - \alpha_2 = 4 - 0 - 4 = 0$$

$$C - \beta_3 - \alpha_3 = 4 - 4 - 0 = 0$$

$$C - \beta_1^* - \alpha_1^* = 4 - 1.25 - 2.75 = 0$$

$$C - \beta_2^* - \alpha_2^* = 4 - 4 - 0 = 0$$

$$C - \beta_3^* - \alpha_3^* = 4 - 2.75 - 1.25 = 0$$

3 Complementary Slackness

$$\mu_i\big(\nabla g_i(\boldsymbol{x}^*)\big) = 0$$

$$\mu_1 g_1(x^*) = \beta_1 \xi_1 = 4 \times 0 = 0, \ \mu_2 g_2(x^*) = \beta_2 \xi_2 = 0 \times 0.3 = 0$$

$$\mu_3 g_3(x^*) = \beta_3 \xi_3 = 4 \times 0 = 0, \ \mu_4 g_4(x^*) = \beta_1^* \xi_1^* = 1.25 \times 0 = 0$$

$$\mu_5 g_5(x^*) = \beta_2^* \xi_2^* = 4 \times 0 = 0, \ \mu_6 g_6(x^*) = \beta_3^* \xi_3^* = 2.75 \times 0 = 0$$

$$\mu_7 g_7(x^*) = \alpha_1(x_1\omega + b + \xi_1 + \varepsilon - y_1) = 0 \times (1 \times 1.5 + 0.6 + 0 + 0.1 - 2) = 0$$

$$\mu_8 g_8(x^*) = \alpha_2(x_2\omega + b + \xi_2 + \varepsilon - y_2) = 4 \times (2 \times 1.5 + 0.6 + 0.3 + 0.1 - 4) = 0$$

$$\mu_9 g_9(x^*) = \alpha_3(x_3\omega + b + \xi_3 + \varepsilon - y_3) = 0 \times (3 \times 1.5 + 0.6 + 0 + 0.1 - 5) = 0$$

$$\mu_{10} g_{10}(x^*) = \alpha_1^*(-x_1\omega - b + \xi_1^* + \varepsilon + y_1) = 2.75 \times (-1 \times 1.5 - 0.6 + 0 + 0.1 + 2) = 0$$

$$\mu_{11} g_{11}(x^*) = \alpha_2^*(-x_2\omega - b + \xi_2^* + \varepsilon + y_2) = 0 \times (-2 \times 1.5 - 0.6 + 0 + 0.1 + 4) = 0$$

$$\mu_{i2} g_{12}(x^*) = \alpha_3^*(-x_3\omega - b + \xi_3^* + \varepsilon + y_2) = 1.25 \times (-3 \times 1.5 - 0.6 + 0 + 0.1 + 5) = 0$$

4. Positive Lagrange Multipliers (for inequality constraints)

$$\mu_i \geq 0$$

All Lagrange multipliers are not negative as below.

$\alpha_1 = 0, \alpha_2 = 4, \alpha_3 = 0, \alpha_1^* = 2.75, \alpha_2^* = 0, \alpha_3^* = 1.25$

$\beta_1 = 4, \beta_2 = 0, \beta_3 = 4, \beta_1^* = 1.25, \beta_2^* = 4, \beta_3^* = 2.75$

The solution and Lagrangian multipliers satisfy all KKT conditions. Hence, they are the optimal solution to the given SVM task.

A.3 An Example of Solving the Dual Form of SVM Regression

A.3.1 Establishment of Simultaneous Equations

Here is an example of setting up a system of equations applying KKT conditions to the SVM task that has only three points to train. The training data is assumed to be two-dimensional: (x1,y1), (x2,y2), (x3,y3)

Optimization task:

Maximize $f(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = -\frac{1}{2}\sum_{i=1}^{3}\sum_{j=1}^{3}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)K(x_i \cdot x_j) +$

$\sum_{i=1}^{3}(\alpha_i - \alpha_i^*)y_i - \sum_{i=1}^{3}(\alpha_i + \alpha_i^*)\varepsilon$

subject to

$$\begin{cases} h_1(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = \alpha_1 + \alpha_2 + \alpha_3 - \alpha_1^* - \alpha_2^* - \alpha_3^* = 0 \\ g_1(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = \alpha_1 \geq 0 \\ g_2(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = \alpha_2 \geq 0 \\ g_3(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = \alpha_3 \geq 0 \\ g_4(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = \alpha_1^* \geq 0 \\ g_5(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = \alpha_2^* \geq 0 \\ g_6(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = \alpha_3^* \geq 0 \\ g_7(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = -\alpha_1 + C \geq 0 \\ g_8(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = -\alpha_2 + C \geq 0 \\ g_9(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = -\alpha_3 + C \geq 0 \\ g_{10}(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = -\alpha_1^* + C \geq 0 \\ g_{11}(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = -\alpha_2^* + C \geq 0 \\ g_{12}(\alpha_1, \alpha_2, \alpha_3, \alpha_1^*, \alpha_2^*, \alpha_3^*) = -\alpha_3^* + C \geq 0 \end{cases}$$

This optimization task is equivalent to the following matrix algebra:

Max

$$-\frac{1}{2}(\alpha_1 \quad \alpha_2 \quad \alpha_3 \quad \alpha_1^* \quad \alpha_2^* \quad \alpha_3^*)\begin{pmatrix} k(x_1 \cdot x_1) & k(x_1 \cdot x_2) & k(x_1 \cdot x_3) & -k(x_1 \cdot x_1) & -k(x_1 \cdot x_2) & -k(x_1 \cdot x_3) \\ k(x_2 \cdot x_1) & k(x_2 \cdot x_2) & k(x_2 \cdot x_3) & -k(x_2 \cdot x_1) & -k(x_2 \cdot x_2) & -k(x_2 \cdot x_3) \\ k(x_3 \cdot x_1) & k(x_3 \cdot x_2) & k(x_3 \cdot x_3) & -k(x_3 \cdot x_1) & -k(x_3 \cdot x_2) & -k(x_3 \cdot x_3) \\ -k(x_1 \cdot x_1) & -k(x_1 \cdot x_2) & -k(x_1 \cdot x_3) & k(x_1 \cdot x_1) & k(x_1 \cdot x_2) & k(x_1 \cdot x_3) \\ -k(x_2 \cdot x_1) & -k(x_2 \cdot x_2) & -k(x_2 \cdot x_3) & k(x_2 \cdot x_1) & k(x_2 \cdot x_2) & k(x_2 \cdot x_3) \\ -k(x_3 \cdot x_1) & -k(x_3 \cdot x_2) & -k(x_3 \cdot x_3) & k(x_3 \cdot x_1) & k(x_3 \cdot x_2) & k(x_3 \cdot x_3) \end{pmatrix}\begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_1^* \\ \alpha_2^* \\ \alpha_3^* \end{pmatrix}$$

$$+ (y_1 - \varepsilon \quad y_2 - \varepsilon \quad y_3 - \varepsilon \quad -y_1 - \varepsilon \quad -y_2 - \varepsilon \quad -y_3 - \varepsilon)\begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_1^* \\ \alpha_2^* \\ \alpha_3^* \end{pmatrix}$$

Subject to

$$(1 \quad 1 \quad 1 \quad -1 \quad -1 \quad -1)\begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_1^* \\ \alpha_2^* \\ \alpha_3^* \end{pmatrix} = (0)$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix}\begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_1^* \\ \alpha_2^* \\ \alpha_3^* \end{pmatrix} \geq (0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad -C \quad -C \quad -C \quad -C \quad -C \quad -C)$$

A system of equations can be set up using KKT conditions.

1. KKT condition 1 (Feasible Constraints):

$g_i(x^*) \geq 0, i = 1, \dots, m, \ h_i(x^*) = 0, i = 1, \dots, k$

in this example,

$$\boldsymbol{x}^T = (\alpha_1 \quad \alpha_2 \quad \alpha_3 \quad \alpha_1^* \quad \alpha_2^* \quad \alpha_3^*), \text{m} = 12, \text{k} = 1$$

The following one equation is established by the KKT condition 1.

$$h_1 = \alpha_1 + \alpha_2 + \alpha_3 - \alpha_1^* - \alpha_2^* - \alpha_3^* = 0$$

KKT condition 2 (Stationary):

$$\nabla f(\boldsymbol{x}^*) + \sum_{i=1}^{m} \mu_i \nabla g_i(\boldsymbol{x}^*) - \sum_{i=1}^{k} \lambda_i \nabla h_i(\boldsymbol{x}^*) = 0$$

Each term of the equation can be expressed in the form of vector as below.

$$\nabla f(\boldsymbol{x}^*) = \begin{pmatrix} \frac{\partial f}{\partial \alpha_1} \\ \frac{\partial f}{\partial \alpha_2} \\ \frac{\partial f}{\partial \alpha_3} \\ \frac{\partial f}{\partial \alpha_1^*} \\ \frac{\partial f}{\partial \alpha_2^*} \\ \frac{\partial f}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} -k(x_1,x_1)(\alpha_1 - \alpha_1^*) - k(x_1,x_2)(\alpha_2 - \alpha_2^*) - k(x_1,x_3)(\alpha_3 - \alpha_3^*) + y_1 - \varepsilon \\ -k(x_2,x_1)(\alpha_1 - \alpha_1^*) - k(x_2,x_2)(\alpha_2 - \alpha_2^*) - k(x_2,x_3)(\alpha_3 - \alpha_3^*) + y_2 - \varepsilon \\ -k(x_3,x_1)(\alpha_1 - \alpha_1^*) - k(x_3,x_2)(\alpha_2 - \alpha_2^*) - k(x_3,x_3)(\alpha_3 - \alpha_3^*) + y_3 - \varepsilon \\ k(x_1,x_1)(\alpha_1 - \alpha_1^*) + k(x_1,x_2)(\alpha_2 - \alpha_2^*) + k(x_1,x_3)(\alpha_3 - \alpha_3^*) - y_1 - \varepsilon \\ k(x_2,x_1)(\alpha_1 - \alpha_1^*) + k(x_2,x_2)(\alpha_2 - \alpha_2^*) + k(x_2,x_3)(\alpha_3 - \alpha_3^*) - y_2 - \varepsilon \\ k(x_3,x_1)(\alpha_1 - \alpha_1^*) + k(x_3,x_2)(\alpha_2 - \alpha_2^*) + k(x_3,x_3)(\alpha_3 - \alpha_3^*) - y_3 - \varepsilon \end{pmatrix}$$

$$\nabla g_1(\boldsymbol{x}^*) = \begin{pmatrix} \frac{\partial g_1}{\partial \alpha_1} \\ \frac{\partial g_1}{\partial \alpha_2} \\ \frac{\partial g_1}{\partial \alpha_3} \\ \frac{\partial g_1}{\partial \alpha_1^*} \\ \frac{\partial g_1}{\partial \alpha_2^*} \\ \frac{\partial g_1}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \nabla g_2(\boldsymbol{x}^*) = \begin{pmatrix} \frac{\partial g_2}{\partial \alpha_1} \\ \frac{\partial g_2}{\partial \alpha_2} \\ \frac{\partial g_2}{\partial \alpha_3} \\ \frac{\partial g_2}{\partial \alpha_1^*} \\ \frac{\partial g_2}{\partial \alpha_2^*} \\ \frac{\partial g_2}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \nabla g_3(\boldsymbol{x}^*) = \begin{pmatrix} \frac{\partial g_3}{\partial \alpha_1} \\ \frac{\partial g_3}{\partial \alpha_2} \\ \frac{\partial g_3}{\partial \alpha_3} \\ \frac{\partial g_3}{\partial \alpha_1^*} \\ \frac{\partial g_3}{\partial \alpha_2^*} \\ \frac{\partial g_3}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \nabla g_4(\boldsymbol{x}^*) = \begin{pmatrix} \frac{\partial g_4}{\partial \alpha_1} \\ \frac{\partial g_4}{\partial \alpha_2} \\ \frac{\partial g_4}{\partial \alpha_3} \\ \frac{\partial g_4}{\partial \alpha_1^*} \\ \frac{\partial g_4}{\partial \alpha_2^*} \\ \frac{\partial g_4}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$\nabla g_5(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_5}{\partial \alpha_1} \\ \dfrac{\partial g_5}{\partial \alpha_2} \\ \dfrac{\partial g_5}{\partial \alpha_3} \\ \dfrac{\partial g_5}{\partial \alpha_1^*} \\ \dfrac{\partial g_5}{\partial \alpha_2^*} \\ \dfrac{\partial g_5}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \nabla g_6(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_6}{\partial \alpha_1} \\ \dfrac{\partial g_6}{\partial \alpha_2} \\ \dfrac{\partial g_6}{\partial \alpha_3} \\ \dfrac{\partial g_6}{\partial \alpha_1^*} \\ \dfrac{\partial g_6}{\partial \alpha_2^*} \\ \dfrac{\partial g_6}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \nabla g_7(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_7}{\partial \alpha_1} \\ \dfrac{\partial g_7}{\partial \alpha_2} \\ \dfrac{\partial g_7}{\partial \alpha_3} \\ \dfrac{\partial g_7}{\partial \alpha_1^*} \\ \dfrac{\partial g_7}{\partial \alpha_2^*} \\ \dfrac{\partial g_7}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \nabla g_8(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_8}{\partial \alpha_1} \\ \dfrac{\partial g_8}{\partial \alpha_2} \\ \dfrac{\partial g_8}{\partial \alpha_3} \\ \dfrac{\partial g_8}{\partial \alpha_1^*} \\ \dfrac{\partial g_8}{\partial \alpha_2^*} \\ \dfrac{\partial g_8}{\partial \alpha_3^*} \end{pmatrix}$$

$$= \begin{pmatrix} 0 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\nabla g_9(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_9}{\partial \alpha_1} \\ \dfrac{\partial g_9}{\partial \alpha_2} \\ \dfrac{\partial g_9}{\partial \alpha_3} \\ \dfrac{\partial g_9}{\partial \alpha_1^*} \\ \dfrac{\partial g_9}{\partial \alpha_2^*} \\ \dfrac{\partial g_9}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \nabla g_{10}(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_{10}}{\partial \alpha_1} \\ \dfrac{\partial g_{10}}{\partial \alpha_2} \\ \dfrac{\partial g_{10}}{\partial \alpha_3} \\ \dfrac{\partial g_{10}}{\partial \alpha_1^*} \\ \dfrac{\partial g_{10}}{\partial \alpha_2^*} \\ \dfrac{\partial g_{10}}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \end{pmatrix}, \nabla g_{11}(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_{11}}{\partial \alpha_1} \\ \dfrac{\partial g_{11}}{\partial \alpha_2} \\ \dfrac{\partial g_{11}}{\partial \alpha_3} \\ \dfrac{\partial g_{11}}{\partial \alpha_1^*} \\ \dfrac{\partial g_{11}}{\partial \alpha_2^*} \\ \dfrac{\partial g_{11}}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 0 \end{pmatrix}$$

$$\nabla g_{12}(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial g_{12}}{\partial \alpha_1} \\ \dfrac{\partial g_{12}}{\partial \alpha_2} \\ \dfrac{\partial g_{12}}{\partial \alpha_3} \\ \dfrac{\partial g_{12}}{\partial \alpha_1^*} \\ \dfrac{\partial g_{12}}{\partial \alpha_2^*} \\ \dfrac{\partial g_{12}}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \end{pmatrix}, \nabla h_1(\boldsymbol{x}^*) = \begin{pmatrix} \dfrac{\partial h_1}{\partial \alpha_1} \\ \dfrac{\partial h_1}{\partial \alpha_2} \\ \dfrac{\partial h_1}{\partial \alpha_3} \\ \dfrac{\partial h_1}{\partial \alpha_1^*} \\ \dfrac{\partial h_1}{\partial \alpha_2^*} \\ \dfrac{\partial h_1}{\partial \alpha_3^*} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ -1 \end{pmatrix}$$

Plugging in the corresponding vectors to the equation, six equations are derived from the six rows, respectively.

$$-k(x_1, x_1)(\alpha_1 - \alpha_1^*) - k(x_1, x_2)(\alpha_2 - \alpha_2^*) - k(x_1, x_3)(\alpha_3 - \alpha_3^*) + y_1 - \varepsilon + \mu_1 - \mu_7 - \lambda_1 = 0$$

$$-k(x_2, x_1)(\alpha_1 - \alpha_1^*) - k(x_2, x_2)(\alpha_2 - \alpha_2^*) - k(x_2, x_3)(\alpha_3 - \alpha_3^*) + y_2 - \varepsilon + \mu_2 - \mu_8 - \lambda_1 = 0$$

$$-k(x_3, x_1)(\alpha_1 - \alpha_1^*) - k(x_3, x_2)(\alpha_2 - \alpha_2^*) - k(x_3, x_3)(\alpha_3 - \alpha_3^*) + y_3 - \varepsilon + \mu_3 - \mu_9 - \lambda_1 = 0$$

$$k(x_1, x_1)(\alpha_1 - \alpha_1^*) + k(x_1, x_2)(\alpha_2 - \alpha_2^*) + k(x_1, x_3)(\alpha_3 - \alpha_3^*) - y_1 - \varepsilon + \mu_4 - \mu_{10} + \lambda_1 = 0$$

$$k(x_2, x_1)(\alpha_1 - \alpha_1^*) + k(x_2, x_2)(\alpha_2 - \alpha_2^*) + k(x_2, x_3)(\alpha_3 - \alpha_3^*) - y_2 - \varepsilon + \mu_5 - \mu_{11} + \lambda_1 = 0$$

$$k(x_3, x_1)(\alpha_1 - \alpha_1^*) + k(x_3, x_2)(\alpha_2 - \alpha_2^*) + k(x_3, x_3)(\alpha_3 - \alpha_3^*) - y_3 - \varepsilon + \mu_6 - \mu_{12} + \lambda_1 = 0$$

KKT condition 3 (Complementary Slackness):

$$\mu_i g_i(x^*) = 0$$

The following twelve equations are set up by the KKT condition 3.

$$\mu_1 g_1(x^*) = \mu_1 \alpha_1 = 0$$

$$\mu_2 g_2(x^*) = \mu_2 \alpha_2 = 0$$

$$\mu_3 g_3(x^*) = \mu_3 \alpha_3 = 0$$

$$\mu_4 g_4(x^*) = \mu_4 \alpha_1^* = 0$$

$$\mu_5 g_5(x^*) = \mu_5 \alpha_2^* = 0$$

$$\mu_6 g_6(x^*) = \mu_6 \alpha_3^* = 0$$

$$\mu_7 g_7(x^*) = \mu_7(-\alpha_1 + C) = 0$$

$$\mu_8 g_8(x^*) = \mu_8(-\alpha_2 + C) = 0$$

$$\mu_9 g_9(x^*) = \mu_9(-\alpha_3 + C) = 0$$

$$\mu_{10} g_{10}(x^*) = \mu_{10}(-\alpha_1^* + C) = 0$$

$$\mu_{11} g_{11}(x^*) = \mu_{11}(-\alpha_2^* + C) = 0$$

$$\mu_{12} g_{12}(x^*) = \mu_{12}(-\alpha_3^* + C) = 0$$

Solving the 19 equations simultaneously, the components of $\boldsymbol{x}^*$, which are α_1, α_2, α_3, $\alpha_1{}^*$, $\alpha_2{}^*$ and $\alpha_3{}^*$, are identified.

As is done to solve the primal form of SVM regression, Quadprog is used to solve the simultaneous problem. The solution obtained from the Quadprog is to be checked with the KKT conditions. Once the solution to the system of equations are confirmed, it will be double checked with the result from the SVM package 'e1071' to validate the whole hand calculation approach to the SVM regression. This process will be carried out on both linear and RBF kernel SVM.

A.3.2 Solving SVM Optimization Task Using a Quadratic Solver

The input and output of the Quadprog are as follows:

(Input)

Hyperparameters: $C = 4$, $\varepsilon = 0.1$, $\gamma = 1$ (only for RBF Kernel)

Training Data: $(x1,y1) = (1,2)$, $(x2,y2) = (2,4)$, $(x3,y3) = (3,5)$
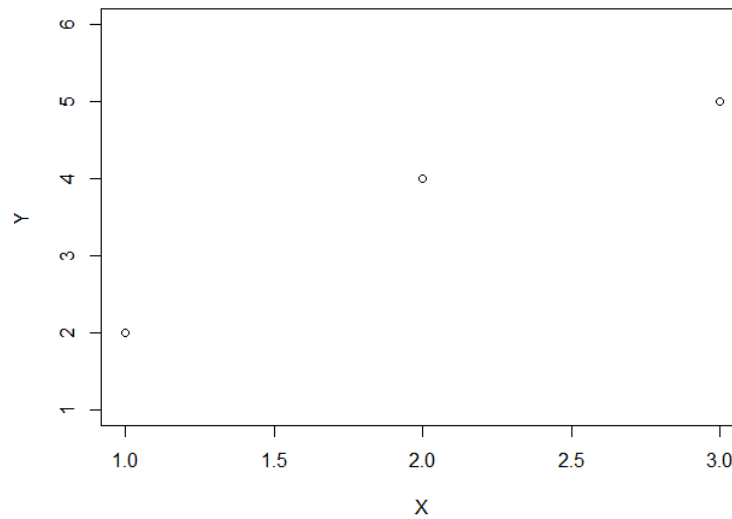
Figure A 12 - Three Sample Points

(Output from the Quadprog)

Case A. Linear SVM regression

Solution: $x^T = (\alpha_1 \quad \alpha_2 \quad \alpha_3 \quad \alpha_1^* \quad \alpha_2^* \quad \alpha_3^*) = (0 \quad 4 \quad 0 \quad 2.75 \quad 0 \quad 1.25)$

Lagrange Multipliers:

$\lambda_i = 0.6 \; (i = 1)$

$\mu_i = 0.2, 0, 0.2, 0, 0.5, 0, 0, 0.3, 0, 0, 0, 0 \; (i = 1, \ldots, 12)$

ω is found by using Eq. 5.1

$$\omega = \sum_{i=1}^{3} (\alpha_i - \alpha_i^*)x_i = (0 - 2.75) \times 1 + (4 - 0) \times 2 + (0 - 1.25) \times 3 = 1.5$$

There are two unbound support vectors, α_1^*, α_3^*. The value of b can be found by applying

any of them to the KKT condition 3.

$$b = \varepsilon + y_1 - x_1 \cdot \omega = 0.1 + 2 - 1 \times 1.5 = 0.6$$

The resultant plot is:

$$f(x) = \omega x + b = 1.5x + 0.6$$
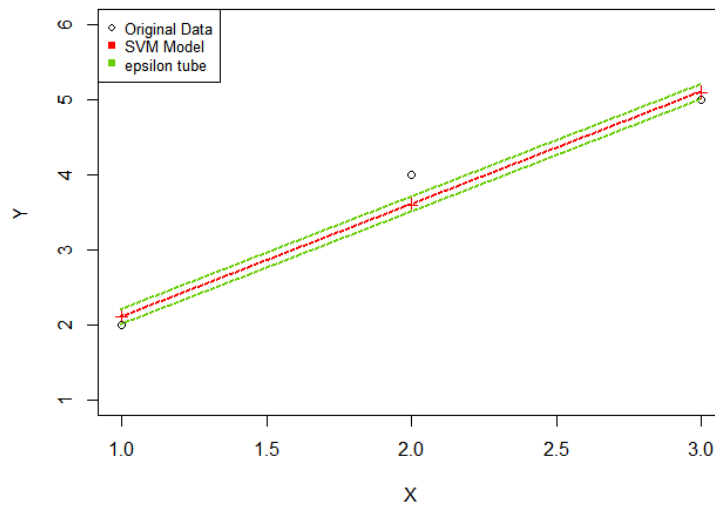
Figure A 13 - Linear (Dual) SVM from the Three Sample Points

Note that the result is the same as the case where the primal form of SVM is solved.

Case B. RBF Kernel SVM regression

Solution: $x^T = (\alpha_1 \quad \alpha_2 \quad \alpha_3 \quad \alpha_1^* \quad \alpha_2^* \quad \alpha_3^*) = (0 \quad 0.52 \quad 1.17 \quad 1.69 \quad 0 \quad 0)$

Lagrange Multipliers:

$\lambda_i = 3.57 \ (i = 1)$

$\mu_i = 0.2, 0, 0, 0, 0.2, 0.2, 0, 0, 0, 0, 0, 0 \ (i = 1, \ldots, 12)$

There are three unbound support vectors, α_1^*, α_2^*, α_3^*. The value of b can be found by applying any of them to the KKT condition 3.

$b = -\varepsilon + y_2 - \sum_{j=1}^{3}(\alpha_j - \alpha_j^*) e^{-(x_j - x_2)^2} = 3.57$

The resultant plot is:

$$f(x) = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)k(x_i \cdot x) + b = -1.69e^{-(x-1)^2} + 0.52e^{-(x-2)^2} + 1.17e^{-(x-3)^2} + 3.57$$
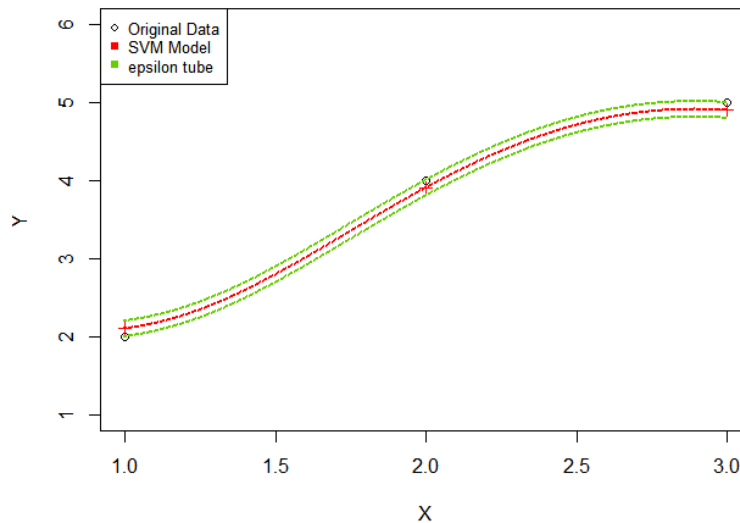
Figure A 14 - RBF Kernel (Dual) SVM from the Three Sample Points

A.3.3 Validation of the Solution in Reference to KKT Conditions

The output from the Quadprog is validated by checking if the output satisfies all KKT

conditions.

Case A. Linear SVM regression

1. Feasible Constraints

$$g_i(\pmb{x}^*) \geq 0, \ h_i(\pmb{x}^*) = 0$$

$$h_1 = \alpha_1 + \alpha_2 + \alpha_3 - \alpha_1^* - \alpha_2^* - \alpha_3^* = 0 + 4 + 0 - 2.75 - 0 - 1.25 = 0$$

$$g_1 = \alpha_1 = 0 \geq 0, g_2 = \alpha_2 = 4 \geq 0, g_3 = \alpha_3 = 0 \geq 0, g_4 = \alpha_1^* = 2.75 \geq 0,$$

$$g_5 = \alpha_2^* = 0 \geq 0, g_6 = \alpha_3^* = 1.25 \geq 0, g_7 = -\alpha_1 + C = 4 \geq 0, g_8 = -\alpha_2 + C = 0 \geq$$

$$0, g_9 = -\alpha_3 + C = 4 \geq 0, g_{10} = -\alpha_1^* + C = 1.25 \geq 0, g_{11} = -\alpha_2^* + C = 4 \geq 0,$$

$$g_{12} = -\alpha_3^* + C = 2.75 \geq 0$$

2. Stationary

$$\nabla f(\pmb{x}^*) - \sum_{i=1}^{m} \mu_i \nabla g_i(\pmb{x}^*) - \sum_{i=1}^{k} \lambda_i \nabla h_i(\pmb{x}^*) = 0$$

$$-x_1, x_1(\alpha_1 - \alpha_1^*) - x_1, x_2(\alpha_2 - \alpha_2^*) - x_1, x_3(\alpha_3 - \alpha_3^*) + y_1 - \varepsilon + \mu_1 - \mu_7 - \lambda_1$$

$$= -1 \times (0 - 2.75) - 2 \times (4 - 0) - 3 \times (0 - 1.25) + 2 - 0.1 + 0.2 - 0 - 0.6 = 0$$

$$-x_2, x_1(\alpha_1 - \alpha_1^*) - x_2, x_2(\alpha_2 - \alpha_2^*) - x_2, x_3(\alpha_3 - \alpha_3^*) + y_2 - \varepsilon + \mu_2 - \mu_8 - \lambda_1 = 0$$

$$= -2 \times (0 - 2.75) - 4 \times (4 - 0) - 6 \times (0 - 1.25) + 4 - 0.1 + 0 - 0.3 - 0.6 = 0$$

$$-x_3, x_1(\alpha_1 - \alpha_1^*) - x_3, x_2(\alpha_2 - \alpha_2^*) - x_3, x_3(\alpha_3 - \alpha_3^*) + y_3 - \varepsilon + \mu_3 - \mu_9 - \lambda_1 = 0$$

$$= -3 \times (0 - 2.75) - 6 \times (4 - 0) - 9 \times (0 - 1.25) + 5 - 0.1 + 0.2 - 0 - 0.6 = 0$$

$$x_1, x_1(\alpha_1 - \alpha_1^*) + x_1, x_2(\alpha_2 - \alpha_2^*) + x_1, x_3(\alpha_3 - \alpha_3^*) - y_1 - \varepsilon + \mu_4 - \mu_{10} + \lambda_1 = 0$$

$$= 1 \times (0 - 2.75) + 2 \times (4 - 0) + 3 \times (0 - 1.25) - 2 - 0.1 + 0 - 0 - 0.6 = 0$$

$$x_2, x_1(\alpha_1 - \alpha_1^*) + x_2, x_2(\alpha_2 - \alpha_2^*) + x_2, x_3(\alpha_3 - \alpha_3^*) - y_2 - \varepsilon + \mu_5 - \mu_{11} + \lambda_1 = 0$$

$$= 2 \times (0 - 2.75) + 4 \times (4 - 0) + 6 \times (0 - 1.25) - 4 - 0.1 + 0.5 - 0 - 0.6 = 0$$

$$x_3, x_1(\alpha_1 - \alpha_1^*) + x_3, x_2(\alpha_2 - \alpha_2^*) + x_3, x_3(\alpha_3 - \alpha_3^*) - y_3 - \varepsilon + \mu_6 - \mu_{12} + \lambda_1 = 0$$

$$= 3 \times (0 - 2.75) + 6 \times (4 - 0) + 9 \times (0 - 1.25) - 5 - 0.1 + 0 - 0 - 0.6 = 0$$

3 Complementary Slackness

$$\mu_i\big(\nabla g_i(x^*)\big) = 0$$

$$\mu_1 g_1(x^*) = \mu_1 \alpha_1 = 0.2 \times 0 = 0, \quad \mu_2 g_2(x^*) = \mu_2 \alpha_2 = 0 \times 4 = 0$$

$$\mu_3 g_3(x^*) = \mu_3 \alpha_3 = 0.2 \times 0 = 0, \quad \mu_4 g_4(x^*) = \mu_4 \alpha_1^* = 0 \times 2.75 = 0$$

$$\mu_5 g_5(x^*) = \mu_5 \alpha_2^* = 0.5 \times 0 = 0, \quad \mu_6 g_6(x^*) = \mu_6 \alpha_3^* = 0 \times 1.25 = 0$$

$$\mu_7 g_7(x^*) = \mu_7(-\alpha_1 + C) = 0 \times (-0 + 4) = 0$$

$$\mu_8 g_8(x^*) = \mu_8(-\alpha_2 + C) = 0.3 \times (-4 + 4) = 0$$

$$\mu_9 g_9(x^*) = \mu_9(-\alpha_3 + C) = 0 \times (-0 + 4) = 0$$

$$\mu_{10} g_{10}(x^*) = \mu_{10}(-\alpha_1^* + C) = 0 \times (-2.75 + 4) = 0$$

$$\mu_{11} g_{11}(x^*) = \mu_{11}(-\alpha_2^* + C) = 0 \times (-0 + 4) = 0$$

$$\mu_{12} g_{12}(x^*) = \mu_{12}(-\alpha_3^* + C) = 0 \times (-1.25 + 4) = 0$$

4. Positive Lagrange Multipliers (for inequality constraints)

$$\mu_i \geq 0$$

All Lagrange multiplies calculated from Quadprog are not negative as below.

$$\mu_i = 0.2, 0, 0.2, 0, 0.5, 0, 0, 0.3, 0, 0, 0, 0 \ (i = 1, \ldots, 12)$$

The solution and Lagrangian multipliers satisfy all KKT conditions. Hence, they are the optimal solution to the given SVM task.

Case B. RBF Kernel SVM regression

1. Feasible Constraints

$$g_i(\boldsymbol{x}^*) \geq 0, \ h_i(\boldsymbol{x}^*) = 0$$

$$h_1 = \alpha_1 + \alpha_2 + \alpha_3 - \alpha_1^* - \alpha_2^* - \alpha_3^* = 0 + 0.52 + 1.17 - 1.69 - 0 - 0 = 0$$

$$g_1 = \alpha_1 = 0 \geq 0, g_2 = \alpha_2 = 0.52 \geq 0, g_3 = \alpha_3 = 1.17 \geq 0, g_4 = \alpha_1^* = 1.69 \geq 0$$

$$g_5 = \alpha_2^* = 0 \geq 0, g_6 = \alpha_3^* = 0 \geq 0, g_7 = -\alpha_1 + C = 4 \geq 0, g_8 = -\alpha_2 + C = 3.48 \geq 0$$

$$g_9 = -\alpha_3 + C = 2.83 \geq 0, g_{10} = -\alpha_1^* + C = 2.31 \geq 0, g_{11} = -\alpha_2^* + C = 4 \geq 0$$

$$g_{12} = -\alpha_3^* + C = 4 \geq 0$$

2. Stationary

$$\nabla f(x^*) - \sum_{i=1}^{m} \mu_i \nabla g_i(x^*) - \sum_{i=1}^{k} \lambda_i \nabla h_i(x^*) = 0$$

$-k(x_1, x_1)(\alpha_1 - \alpha_1^*) - k(x_1, x_2)(\alpha_2 - \alpha_2^*) - k(x_1, x_3)(\alpha_3 - \alpha_3^*) + y_1 - \varepsilon + \mu_1 - \mu_7 - \lambda_1$

$= -e^{-(1-1)^2} \times (0 - 1.69) - e^{-(1-2)^2} \times (0.52 - 0) - e^{-(1-3)^2} \times (1.17 - 0) + 2 - 0.1 + 0.2 - 0 - 3.57 \cong 0$

$-k(x_2, x_1)(\alpha_1 - \alpha_1^*) - k(x_2, x_2)(\alpha_2 - \alpha_2^*) - k(x_2, x_3)(\alpha_3 - \alpha_3^*) + y_2 - \varepsilon + \mu_2 - \mu_8 - \lambda_1 = 0$

$= -e^{-(2-1)^2} \times (0 - 1.69) - e^{-(2-2)^2} \times (0.52 - 0) - e^{-(2-3)^2} \times (1.17 - 0) + 4 - 0.1 + 0 - 0 - 3.57 \cong 0$

$-k(x_3, x_1)(\alpha_1 - \alpha_1^*) - k(x_3, x_2)(\alpha_2 - \alpha_2^*) - k(x_3, x_3)(\alpha_3 - \alpha_3^*) + y_3 - \varepsilon + \mu_3 - \mu_9 - \lambda_1 = 0$

$= -e^{-(3-1)^2} \times (0 - 1.69) - e^{-(3-2)^2} \times (0.52 - 0) - e^{-(3-3)^2} \times (1.17 - 0) + 5 - 0.1 + 0 - 0 - 3.57 \cong 0$

$k(x_1, x_1)(\alpha_1 - \alpha_1^*) + k(x_1, x_2)(\alpha_2 - \alpha_2^*) + k(x_1, x_3)(\alpha_3 - \alpha_3^*) - y_1 - \varepsilon + \mu_4 - \mu_{10} + \lambda_1 = 0$

$= e^{-(1-1)^2} \times (0 - 1.69) + e^{-(1-2)^2} \times (0.52 - 0) + e^{-(1-3)^2} \times (1.17 - 0) - 2 - 0.1 + 0 - 0 + 3.57 \cong 0$

$k(x_2, x_1)(\alpha_1 - \alpha_1^*) + k(x_2, x_2)(\alpha_2 - \alpha_2^*) + k(x_2, x_3)(\alpha_3 - \alpha_3^*) - y_2 - \varepsilon + \mu_5 - \mu_{11} + \lambda_1 = 0$

$= e^{-(2-1)^2} \times (0 - 1.69) + e^{-(2-2)^2} \times (0.52 - 0) + e^{-(2-3)^2} \times (1.17 - 0) - 4 - 0.1 + 0.2 - 0 + 3.57 \cong 0$

$k(x_3, x_1)(\alpha_1 - \alpha_1^*) + k(x_3, x_2)(\alpha_2 - \alpha_2^*) + k(x_3, x_3)(\alpha_3 - \alpha_3^*) - y_3 - \varepsilon + \mu_6 - \mu_{12} + \lambda_1 = 0$

$= e^{-(3-1)^2} \times (0 - 1.69) + e^{-(3-2)^2} \times (0.52 - 0) + e^{-(3-3)^2} \times (1.17 - 0) - 5 - 0.1 + 0.2 - 0 - 3.57 \cong 0$

The left-hand side of the equation above is not exactly zero because rounded numbers are inserted.

3 Complementary Slackness

$$\mu_i \big(\nabla g_i(x^*) \big) = 0$$

$\mu_1 g_1(x^*) = \mu_1 \alpha_1 = 0.2 \times 0 = 0, \ \mu_2 g_2(x^*) = \mu_2 \alpha_2 = 0 \times 0.52 = 0$

$\mu_3 g_3(x^*) = \mu_3 \alpha_3 = 0 \times 1.17 = 0, \ \mu_4 g_4(x^*) = \mu_4 \alpha_1^* = 0 \times 1.69 = 0$

$\mu_5 g_5(x^*) = \mu_5 \alpha_2^* = 0.2 \times 0 = 0, \ \mu_6 g_6(x^*) = \mu_6 \alpha_3^* = 0.2 \times 0 = 0$

$\mu_7 g_7(x^*) = \mu_7(-\alpha_1 + C) = 0 \times (-0 + 4) = 0$

$$\mu_8 g_8(x^*) = \mu_8(-\alpha_2 + C) = 0 \times (-0.52 + 4) = 0$$

$$\mu_9 g_9(x^*) = \mu_9(-\alpha_3 + C) = 0 \times (-1.17 + 4) = 0$$

$$\mu_{10} g_{10}(x^*) = \mu_{10}(-\alpha_1^* + C) = 0 \times (-1.69 + 4) = 0$$

$$\mu_{11} g_{11}(x^*) = \mu_{11}(-\alpha_2^* + C) = 0 \times (-0 + 4) = 0$$

$$\mu_{12} g_{12}(x^*) = \mu_{12}(-\alpha_3^* + C) = 0 \times (-0 + 4) = 0$$

4. Positive Lagrange Multipliers (for inequality constraints)

$$\mu_i \geq 0$$

All Lagrange multiplies calculated from Quadprog are not negative as below.

$$\mu_i = 0.2, 0, 0, 0, 0.2, 0.2, 0, 0, 0, 0, 0, 0 \ \ (i = 1, \ldots, 12)$$

.

The solution and Lagrangian multipliers satisfy all KKT conditions. Hence, they are the optimal solution to the given SVM task.

A.3.4 Validation of the Described Hand Calculation Approach in Reference to the SVM Package 'e1071'

The whole hand calculation approached described above is validated by comparing the result with the output from SVM package e1071.

'

Case A. Linear SVM regression

The result of training linear a SVM model on the same training data with the same hyperparameters using the SVM package 'e1071':

$\alpha_i - \alpha_i^* = -2.75, 4, -1.25$ $(i = 1, 2, 3)$

$b = -0.6$ (the opposite sign convention is used for the bias term in 'e1071')

This result agrees with the earlier one, which was obtained without using the SVM package.

Name	Type	Value
rho	double [1]	-0.6
compprob	logical [1]	FALSE
probA	NULL	Pairlist of length 0
probB	NULL	Pairlist of length 0
sigma	NULL	Pairlist of length 0
coefs	double [3 x 1]	-2.75 4.00 -1.25

Figure A 15 - The result of linear SVM trained from the Three Sample Points with 'e1071'

Case B. RBF Kernel SVM regression

The result of training linear a SVM model on the same training data with the same hyperparameters using the SVM package 'e1071':

$\alpha_i - \alpha_i^* = -1.684, 0.517, 1.167$ $(i = 1, 2, 3)$

$b = -3.573$ (the opposite sign convention is used for the bias term in 'e1071')

This result agrees with the earlier one, which was obtained without using the SVM package.

Name	Type	Value
rho	double [1]	-3.573137
compprob	logical [1]	FALSE
probA	NULL	Pairlist of length 0
probB	NULL	Pairlist of length 0
sigma	NULL	Pairlist of length 0
coefs	double [3 x 1]	-1.684 0.517 1.167

Figure A 16 - The result of RBF kernel SVM trained from the Three Sample Points with 'e1071'

APPENDIX B. CONSIDERATION OF SVM IMPLEMENTATION

B.1 Scaling

Scaling is a statistical and mathematical process to fit given numerical values of data in a desired format. Hsu et al. [57] emphasized the importance of scaling in applying SVR. It prevents attributes with bigger numeric ranges from overshadowing those with smaller numeric ranges. It also helps to reduce numerical complexities in the calculation process. There are two ways of scaling. Min-max scaling and standard scaling. Min-max scaling is to adjust the range of each attribute to the other typical ranges, for example, between -1 and 1, or 0 and 1. It should be noted that the same scaling method and rate have to be applied to both training and testing data. Standard scaling is to rearrange data in such a way that their mean is zero and standard variation is 1. The following is the mathematical expression of the two scaling methods.

1) Min-Max Scaling

$$LB + \frac{X - X_{min}}{X_{max} - X_{min}} (UB - LB) \qquad \text{(EQ.B 1)}$$

2) Standard Scaling

$$\frac{X - X_{mean}}{\sigma_X} \qquad \text{(EQ.B 2)}$$

Crone et al. [58] tested SVM models with different Min-Max scaling and standard scaling. There was not a single model which always showed the best result, the resultant

difference between methods were not significantly large. Thus, the defaultt scaling

option of e1071[46] library, which standard scaling, is used for this research.

B.2 Kernel Selection

Kernel functions map data into a higher dimensional feature space so that the

transformed data can be linearly separated or regressed

$$\varphi: R^n \rightarrow R^m \ (n < m)$$

$$K(x, y) = \varphi(x)^T \varphi(y) \qquad\qquad (\text{EQ.B 3})$$
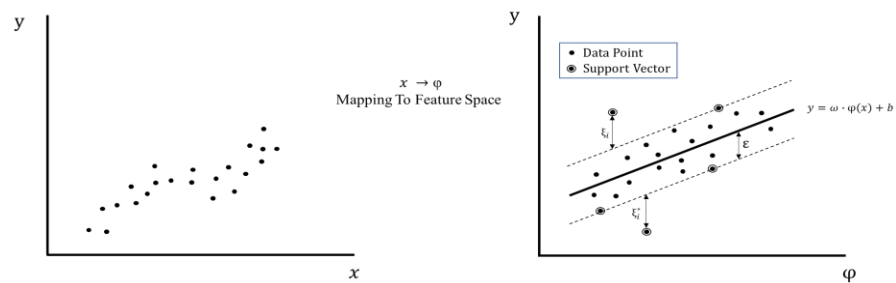
Figure B 1 - Mapping a Non-linear SVR into a Feature Space

As seen in EQ.B1, a kernel function corresponds to an inner product after each data is

transferred to another dimension. Thus, the description of the mapping function φ itself

is not necessarily required. (The only thing needed is replace $(x_i \cdot x_j)$ with $(\varphi(x_i) \cdot$

$\varphi(x_j))$ in a dual equation.

Table B 1 - Summary of Kernel Function Characteristics

	Kernel Function Type	No. of hyperparameters	Description
1	Linear $K(x_i, x_j) = x_i^T x_j$	2 (ε, C)	Actually, this does not involve mapping process. This is used when original data follows linear patterns well. It has the smallest number of hyperparameters to tune.
2	Polynomial $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d$	4 (ε, C, γ, d)	This can capture non-linear patterns. Usually, r is set to zero, and d is set between 1 and 10. Due to relatively larger numbers of hyperparameters to tune, it is not as popular as RBF.
3	Radial Based Function $K(x_i, x_j) = \exp\left(-\gamma \|x_j - x_j\|^2\right)$	3 (ε, C, γ)	This is the most generally used option due to its good performance of catching non-linear patterns and relatively smaller number of hyperparameters to tune. According a practical guide, it is recommend to search C in the range $[2^{-5} : 2^{15}]$ and γ in the range $[2^{-15} : 2^3]$.

Among these, RBF is most widely used due to its capacity to catch non-linear patterns. Compared to polynomial functions, it has less numbers of hyperparameters to tune, thus has less numerical difficulty and complexity [31]. When it comes to the sigmoid kernel, it does not work with certain parameters, not making the inner product of two vectors in feature space [59]. Hence, RBF is to be used as a default kernel function.

B.3 Training and Hyperparameter Optimization

A SVM model is created by using a set of input data including independent variables and a dependent variable in such way that the model fits the input data. The data assigned for generating the model is called training data. In many cases, there would be a big difference between the output values from the first model and the values of dependent variables of the training data set. This difference can be reduced by adjusting hyperparameters of the SVM model such as ε, C, and γ. This adjusting process is called tuning. However, even if the difference is reduced to be small by tuning, it does not necessarily lead to a good prediction when new input data is used. This discrepancy happens when the focus of tuning is only on a decrease in error and not on capturing general patterns of the training data. This is called overfitting. To prevent this, the updated model needs to be validated with other types of data, namely validation data. Once the tuning process is finished with training and validation data, the final step is to test with the data that is never used during the tuning process. It is important to understand the difference between validation data and testing data. Validation data is used for tuning hyperparameters of machine learning models, preventing the risk of overfitting, which may happen when tuning is done only with training data. Testing data is used only for final testing, therefore, should not be used as feedback for tuning process in order to maintain the reliability of the test result.

(a) Properly established model   (b) Overfitted model
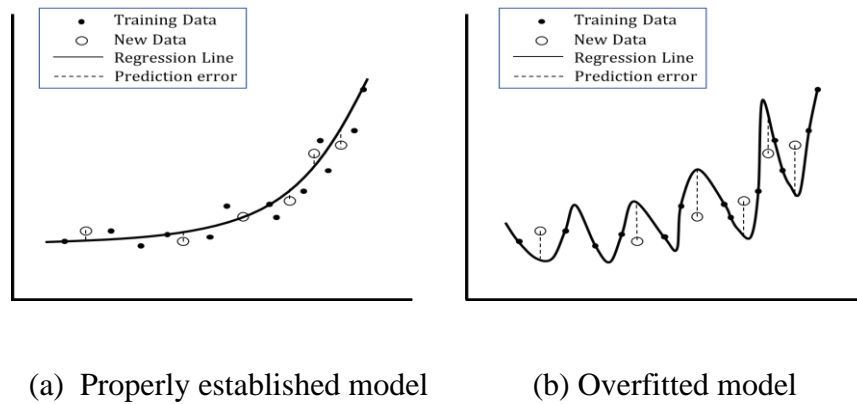
Figure B 2 - Examples of a Properly Established Model and Overfitted Model

B.3.1 Validation Methods

Hold-out method

A portion of data is held out only for validation. In general, the amount of data assigned for validation is smaller than that for training. There is no strict rule for partitioning. Typically, 20% or 30% of data is held out for validation. Quick computational speed is its most noticeable merit. On the contrary, since only a part of data is used for validation, the metrics from the validation data has limited reliability. However, if the quantity of data set is large enough, thus more data can be held out for validation, this drawback can be counteracted.
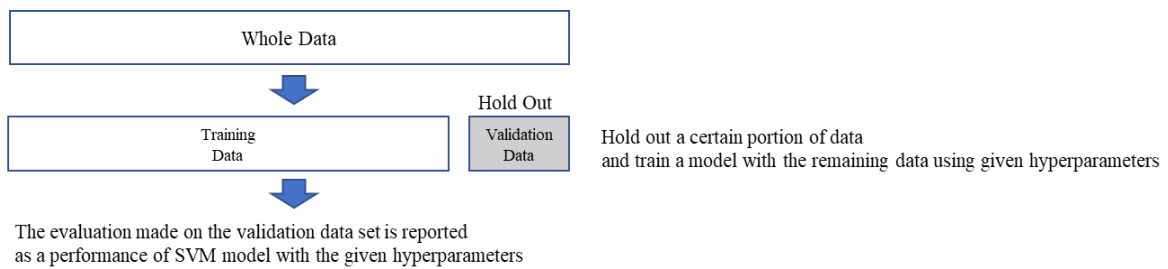
Figure B 3 - Hold-out Method

(Repeated) K-fold cross validation

Except for data for testing, the remaining data is divided into k-folds. For a predefined hyper parameters condition, each fold serves as a validation set one after the other while the remaining k-1 folds data sets are used for training a model. The average of validation metrics from all turns is used to evaluate the validity of the model generated by the predefined hyperparameters. Compared to the hold-out method, which validates its model only with a portion of the data, this validation metrics is more reliable since it validates model with all data set. (except for testing data). Another advantage is that it trains a model with all data set (except for testing data) while the hold-out method cannot use the hold-out portion for training purpose. This merit is manifested when the number of data set if limited. The drawback is that it is computationally more expensive than the hold-out method. There no strict rule defining the optimal number of folds, but 5-fold or 10-fold cross validation is typically used.
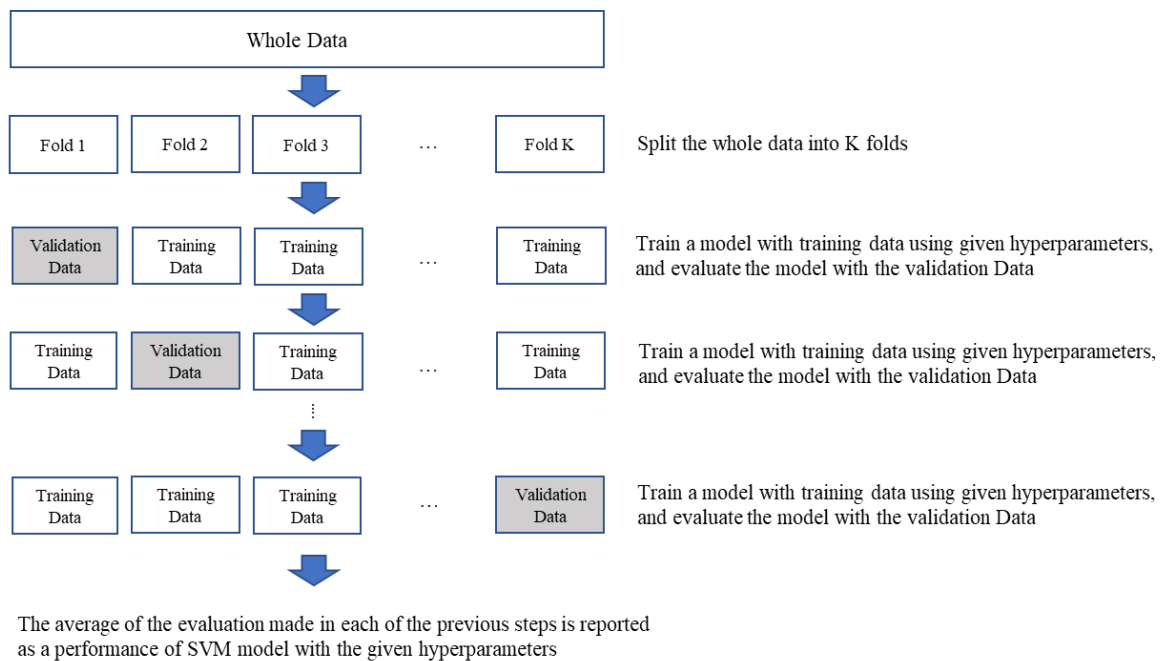
Figure B 4 - *k*-fold Validation

Although the cross validation by itself is a good and robust method for validation, it has a source of uncertainty in its procedure. It is the moment of partitioning the data into k folds that induces inconsistent performance. For the same size of data set, the way it is split into *k* fold can be numerous. This partition is usually carried out by a random algorithm of computer. In the case where the data for training is clean without serious noise, it does not matter how the data is divided into *k* folds. Regardless of the partition, SVM will shows consistent result. However, if this is not the case, the results may vary. To minimize such an inconsistency caused by partition, *k*-fold cross validation is often carried out repeatedly so that the SVM algorithm can multiple cases of partition.

Leave-one out cross validation

This is a special case of *k*-fold validation where the number of folds is equal to the number of data point available for training and validation. Although it brings more stable validation metrics, the computational cost is extremely expensive.
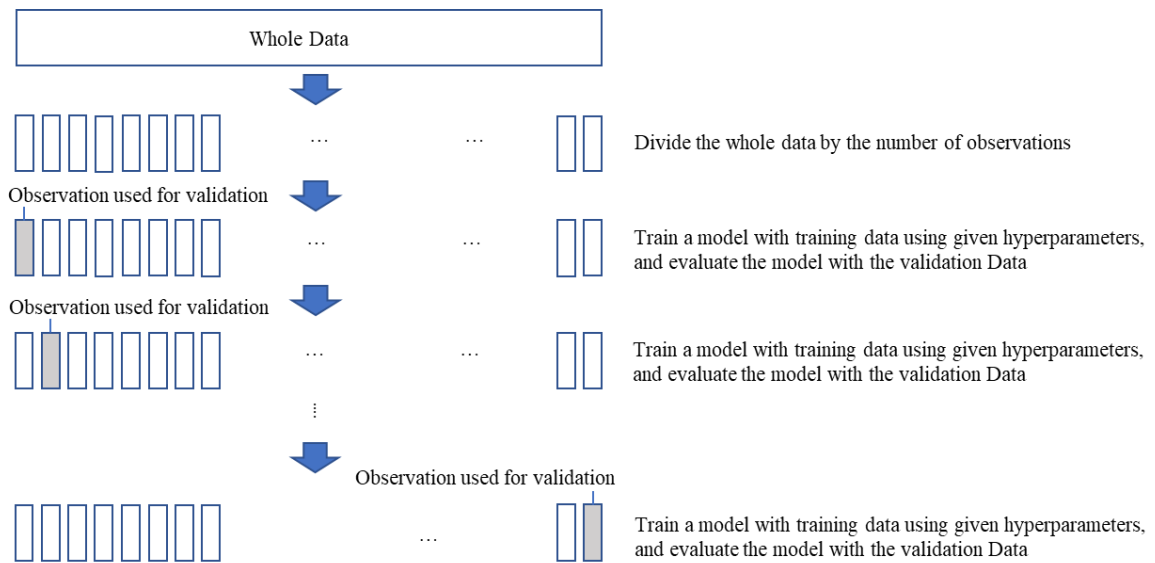
Figure B 5 - Leave-one Out Cross Validation

In this research 5 Fold Cross Validation is used by default and 10 Repeated 5 Fold Cross Validation is used for comparative study.

B.3.2 Influence of Hyperparameters

Hyperparameters are a set of parameters defined by users before machine learning models are trained with training data set. RBF SVM has three hyperparameters: ε, C, and

γ. The parameters that are calculated as a result of training are not called

hyperparameters. For example, in SVM regression function f(x) = $\boldsymbol{\omega} \cdot \boldsymbol{x} + b$, the

coefficient $\boldsymbol{\omega}$ and b are specified after the model is trained with input data, thus they are

not called hyperparameters. In other words, they are given as a result of SVM algorithm

training with predetermined hyperparameters.

Epsilon (ε)

A certain amount of distance from a SVM model. SVM algorithm generates a regression

plot to put as many data within this range as possible. (See Appendix A for the details of

its influence.)

Cost (C)

It determines how much penalty is imposed on the data outside of ε. (See Appendix A

for the details of its influence.)

Gamma(γ) (for RBF)

The inverse of the radius of influence of support vectors. It defines how far a support

vector have its influences on other data set.

- Small γ means a large influence a support vector can have on nearby data. It makes a

regression plot flat.

- Lage γ, means a small influence a support vector can have on nearby data. It makes a graph spiky. Although it increases the accuracy, it increases the risk of overfitting as well.

The impact of gamma would be understood more intuitively by watching an extremely simple case where only one support vector constitutes a SVM model. (See EQ.B4 and Figure B6.)

$$f(x) = e^{-\gamma(x-1)^2}$$
(EQ.B 4)
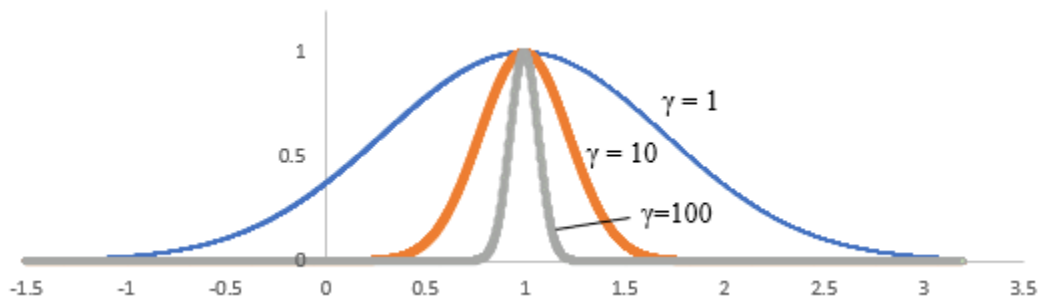
Figure B 6 - Impact of gamma

Figure B6 indicates plot of EQ.B4 with different value of γ. As stated above, smaller gamma makes the influence of support vector wider and vice versa. Figure B7 shows an impact of gamma with more support vectors.
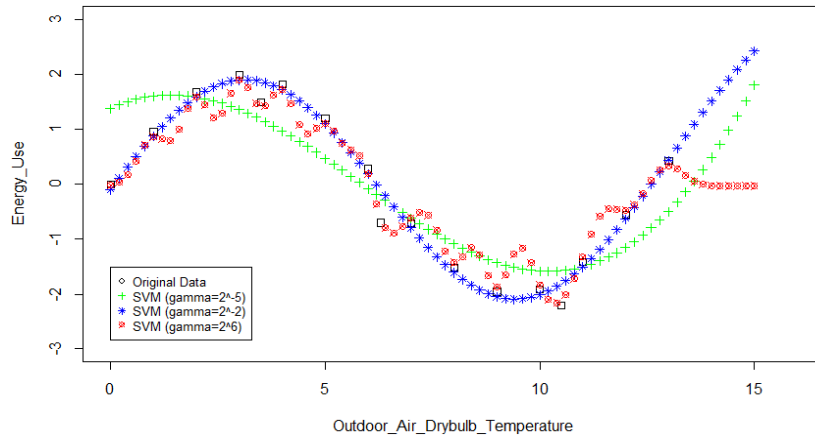
$$\varepsilon = 0.0625, C = 1024$$

Figure B 7 - The Impact of Gamma

B.3.3 Optimization (Tuning) Methods

Many methods have been being developed in an attempt to efficiently search for the best

hyperparameters. Yet there is no 'best' way to make others obsolete. Among them, grid

search (GS) is arguably the most conservative, yet most widely used method in the

selection of the SVR hyperparameters due to its straightforwardness. The following is

the examples of hyperparameter tuning methods.

Exhaustive grid search

Exhaustive grid search is the most simple and accurate technique available [60]. Hence,

as stated above, this is arguably the most generally method. Once the range of

hyperparameters, in which the best ones are expected to exist, are set up, it calculates the

validity of every possible combination of hyperparameters and shows the best one. The downside is its high computational cost.

Random search

As the name suggests, it searches the best hyperparameter sets by randomly chosen combination of hyperparameters. Bergstra and Bengio [61] empirically and theoretically showed that the random search outperforms grid search methods in many cases.
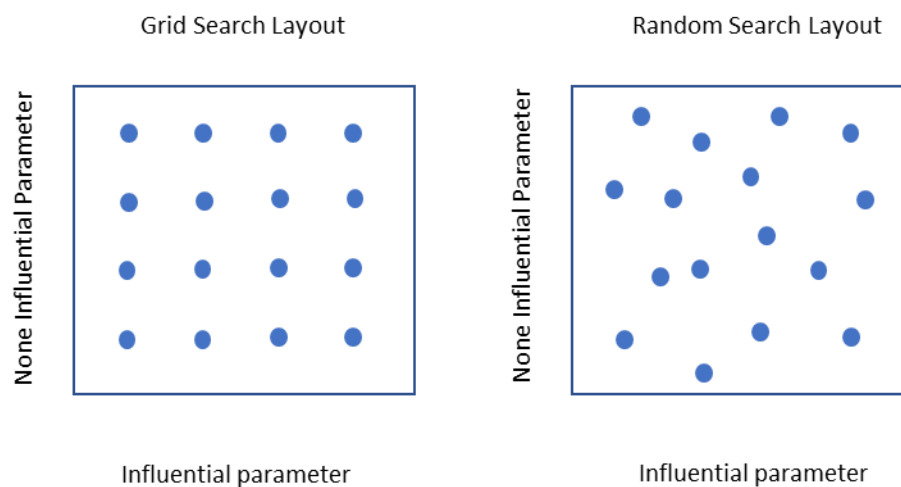
Figure B 8 - Grid search and random search (Adapted from Bergstra and Bengio, 2012 [61])

In general. not all hyperparameters are equally important. Contrary to this phenomenon, the grid search method searches every grid on the premise that all hyperparameters are equally important. This causes the grid search method to waste its computational power

searching for relatively less important hyper parameters. As a result, as seen in Figure B8, for the same number of search, random search method can carry out more detailed test on the important hyperparameters.

Analytical selection technique

Cherkassky et al. [49] proposed an analytical approach to quickly find the parameters C and ε. According to Mattera and Haykin [62] , a good value for C can be considered as equal to the range of output. However, this is vulnerable to outlier. Thus, the EQ.B5 is proposed to tackle the problem. Eq.B6 is developed from an intuitive idea that ε should be proportional to the input noise level and inversely proportional to the number of samples. The coefficient 3 and $\sqrt{\ln n}$ is empirically derived.

$$C = \max\left(\bar{y} + 3\sigma_y , \bar{y} - 3\sigma_y\right) \qquad \text{(EQ.B 5)}$$

$$\varepsilon = 3\sigma \sqrt{\frac{\ln n}{n}} \qquad \text{(EQ.B 6)}$$

Where $\bar{y}$ is the mean of training data, σ_y is the standard deviation of training data, and σ is the standard deviation of input noise. In addition to the equations above, Louw and Steel [51] used empirical equation $\gamma = 1/p$ for the hyper parameter γ. Tang et al. [48] proposed a way to derive an initial γ using a golden section search method. Although these analytical formulas return a quick estimation, their accuracy is not high. Thus, they

are usually used for first estimate and additional estimation needs to be followed. More detained explanation of these combined approaches is to be followed in the next section.

4) Combination of analytical selection techniques and other methodologies.

Tang et al. [48] not only proposed the analytical approach to find an initial γ, but also combined it with one dimensional grid search to find appropriate values of other hyperparameters and to update the value of γ.

Kaneko and Funatsu [50] used Eq. 16 to set an initial C, and used the golden selection search algorithm proposed by Tang et al. [48] to set an initial γ. Optimal ε was found by an one dimensional grid search method while the C and γ kept as the initial values. Optimal C and γ were found consecutively by the same principle. This method shows the same level of accuracy as exhaustive grid search while achieving significant savings in computational time. Tsirikoglou et al. [60] used a combination of analytical approach of Cherkassky et al. [49] and genetic algorithm to show statistically better results compared to the cases where only the genetic algorithm were used.

In this research, two exhaustive grid search and a newly suggested methods are tested to find a more appropriate option for building baseline modeling purpose. The performance is checked in different with daily frequency data considering not only accuracy, but also computational time, The scope of grid search is determined in accordance with the practical SVM guide [57].

Table B 2 - Summary of SVM Implementation Plans

	Consideration	Option	Use
1	Scaling	- Min-Max Scaling - Standard Scaling (Normalization)	Standard Scaling (Normalization) (mean 0, standard deviation 1)
2	Validation strategy	(Options) - Hold-out - (Repeated) K-fold cross validation - Leave one out cross validation - Others	N repeated K-fold cross validation (K = 5, N = 1 by default, 10 for comparative study)
3	Kernel function selection	(Options) - Linear - Polynomial - RBF - Others	RBF
4	Tuning method selection	(Options) - Exhaustive grid search - Random search - Analytic approach - Genetic algorithm - Analytic approach plus other methods - Others	- Grid Search - Newly suggested method (one dimensional grid search with recommended initial values)
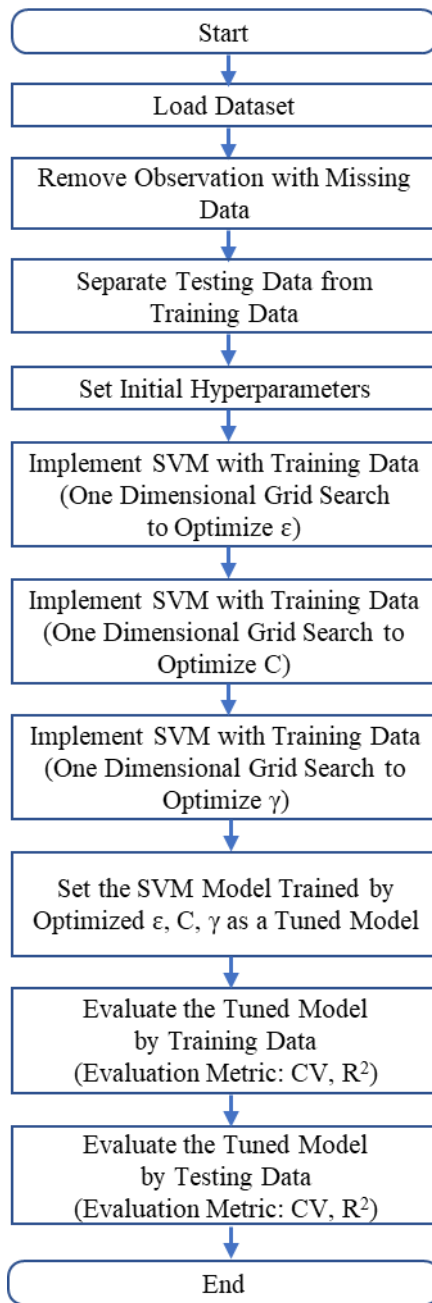
Figure B 9- SVM Implementation (ODGS) Schematic

APPENDIX C. SVM REGRESSION PATTER OBSERVATION

C.1 Linear Input Data

C.1.1 Daily Data

C.1.1.1 Ideal Input
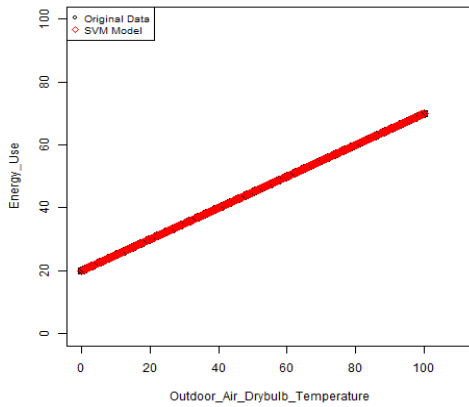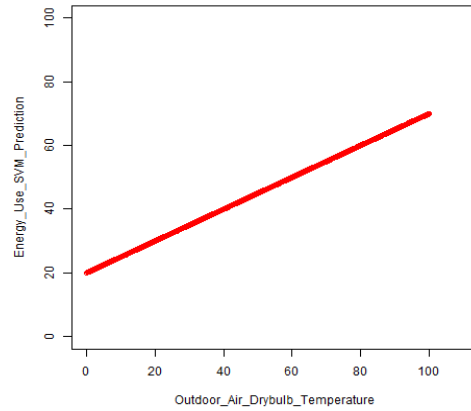
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 0.0086, R^2 = 0.9999, number of support vector = 350

Figure C 1 - SVM Regression Observation (Ideal Linear Daily Data)

C.1.1.2 10% Noisy Input

1) homoscedastic Noise
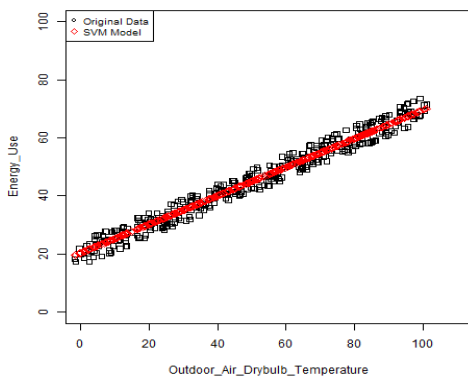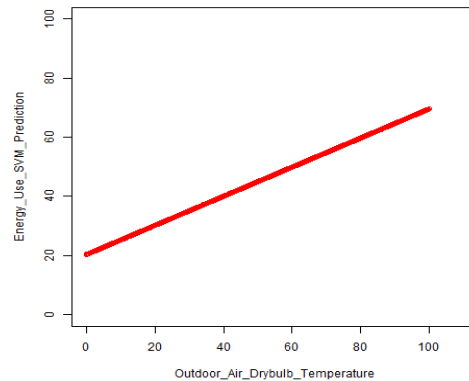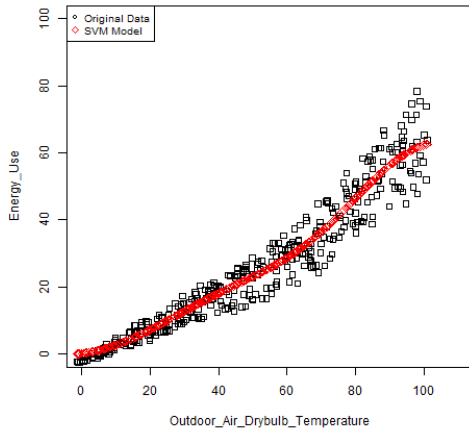

(a) original data and SVM regression points   (b) SVM model continuous plot
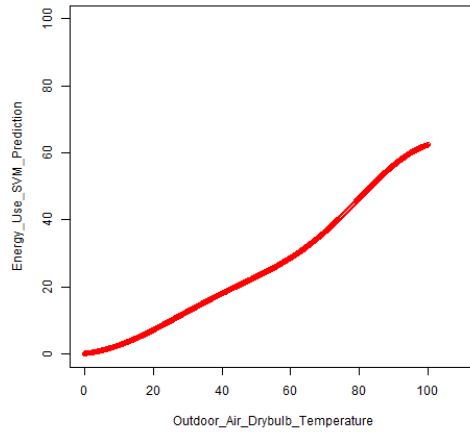
CV (%) = 5.67, R^2 = 0.9703

Figure C 2 - SVM Regression Observation (10% Noisy Linear Daily Data 1)

2) heteroscedastic noise
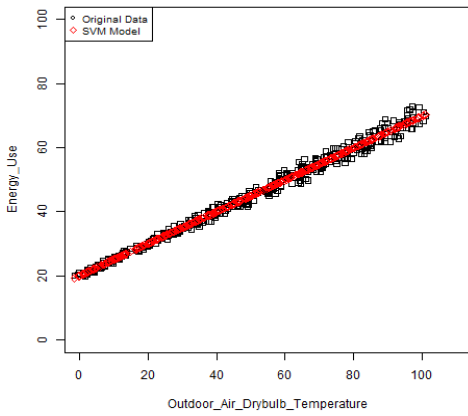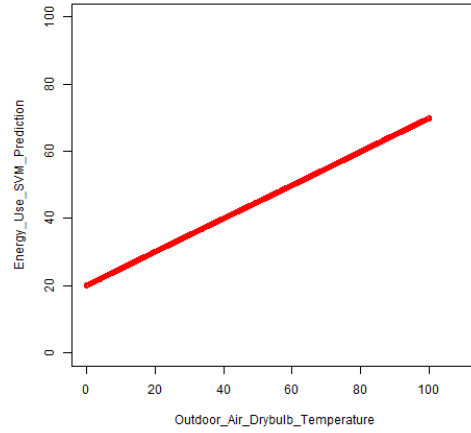
(a) original data and SVM regression points (b) SVM model continuous plot
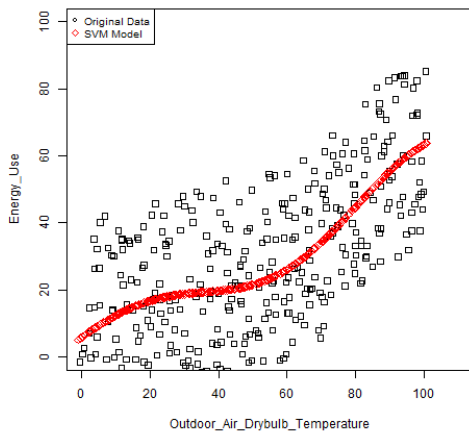
CV (%) = 3.36, R^2 = 0.9893

Figure C 3 - SVM Regression Observation (10% Noisy Linear Daily Data 2)

C.1.1.3 30% Noisy Input

1) homoscedastic Noise
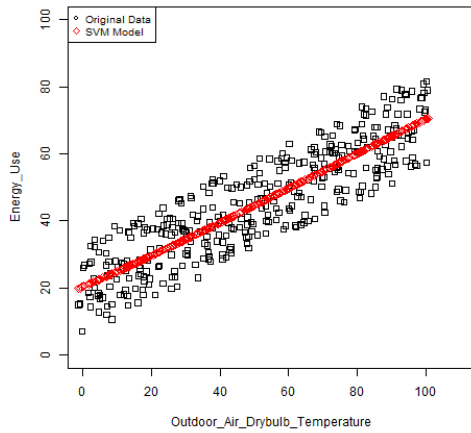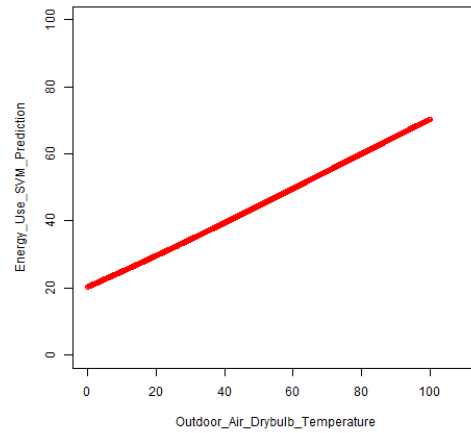
(a) original data and SVM regression points (b) SVM model continuous plot

CV (%) = 17.42, R^2 = 0.7776

Figure C 4 - SVM Regression Observation (30% Noise Linear Daily Data 1)

2) heteroscedastic Noise
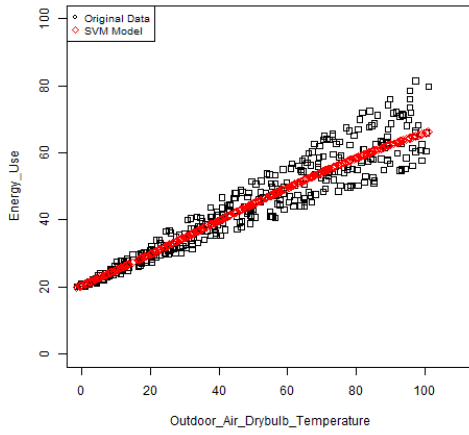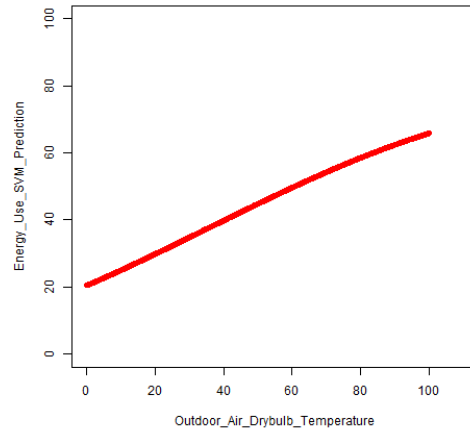
(a) original data and SVM regression points   (b) SVM model continuous plot
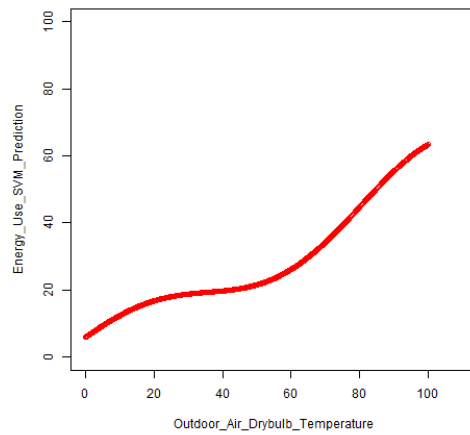
CV (%) = 10.37, R^2 = 0.9058

Figure C 5 - SVM Regression Observation (30% Noise Linear Daily Data 2)

C.1.1.4 50% Noisy Input

1) homoscedastic Noise
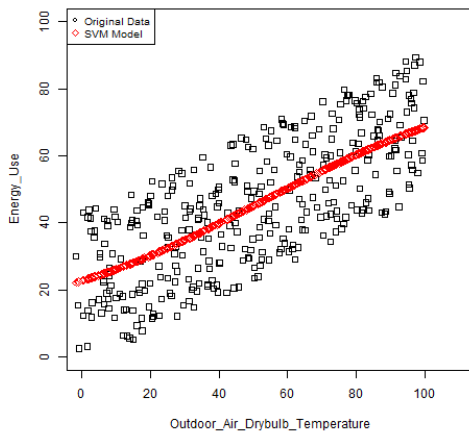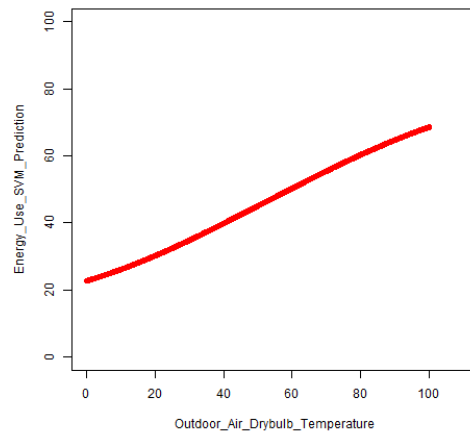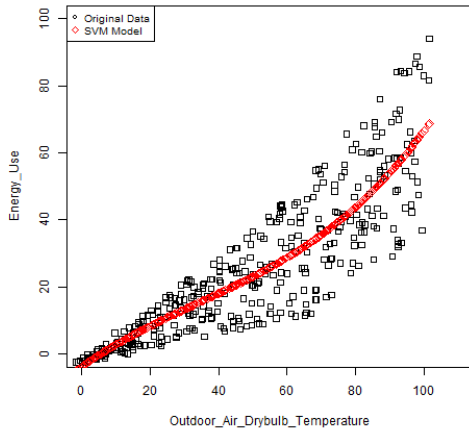

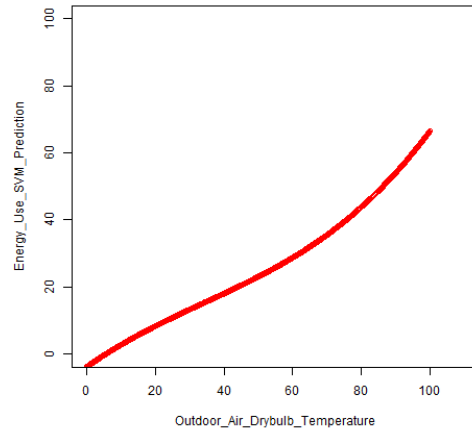(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 29.72, R^2 = 0.5461

Figure C 6 - SVM Regression Observation (50% Noisy Linear Daily Data 1)

2) heteroscedastic Noise
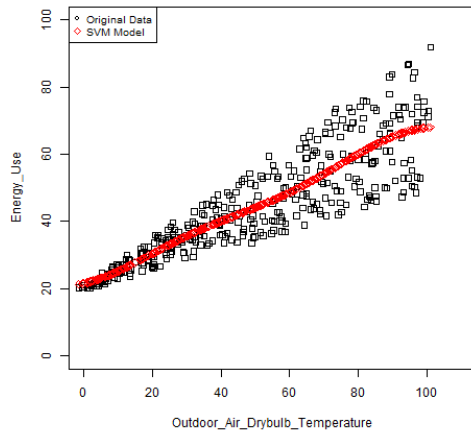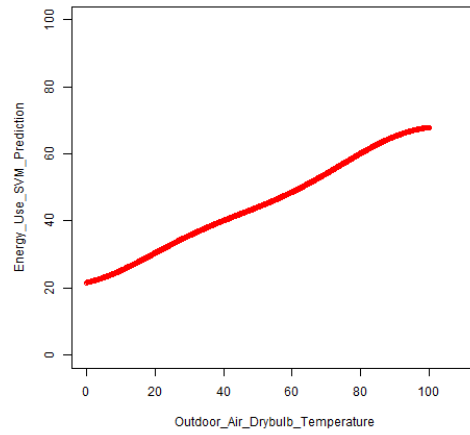
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 16.64, R^2 = 0.7824

Figure C 7 - SVM Regression Observation (50% Noisy Linear Daily Data 2)

C.1.1.5 100% Noisy Input

1) homoscedastic Noise
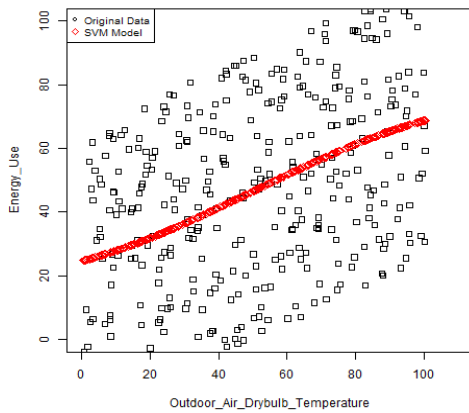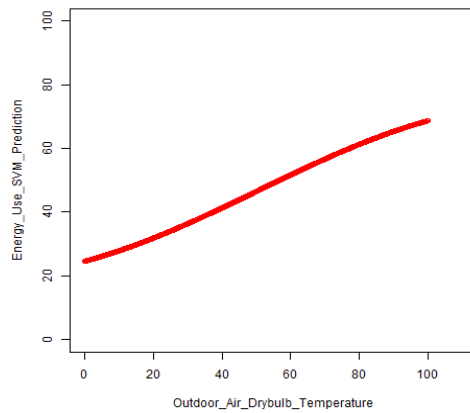
(a) original data and SVM regression point   (b) SVM model continuous plot

CV (%) = 57.90, R^2 = 0.2140

Figure C 8 - SVM Regression Observation (100% Noisy Linear Daily Data 1)

2) heteroscedastic Noise
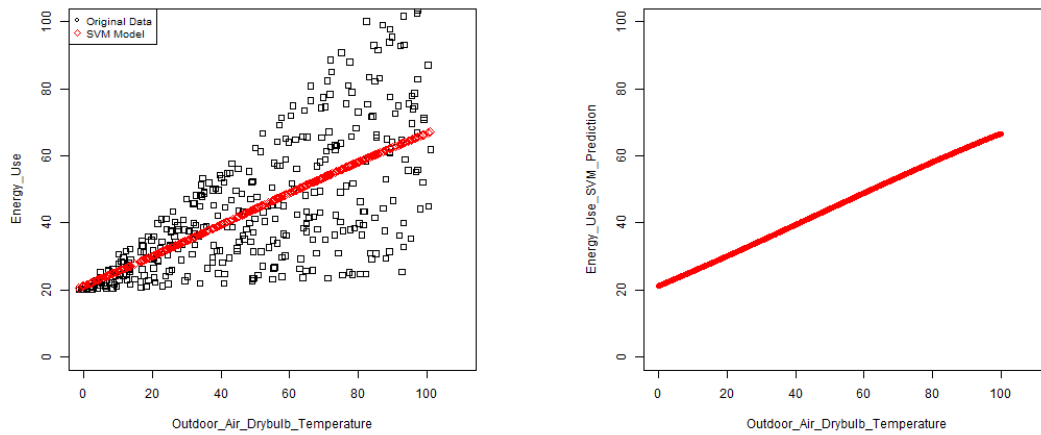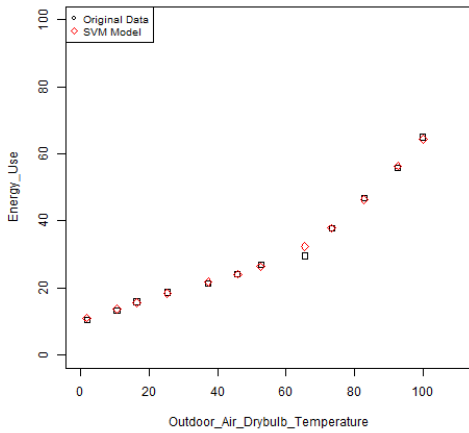

(a) original data and SVM regression points

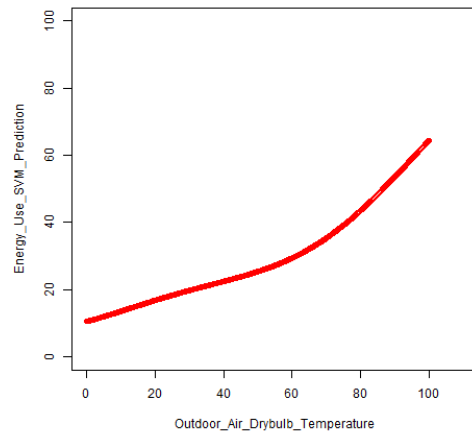(b) SVM model continuous plot

CV (%) = 33.88, R^2 = 0.4695

Figure C 9 - SVM Regression Observation (100% Noisy Linear Daily Data 2)

C.1.2 Monthly Data
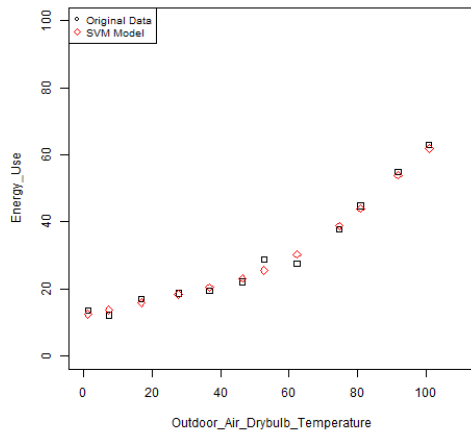
C.1.2.1 Ideal Input
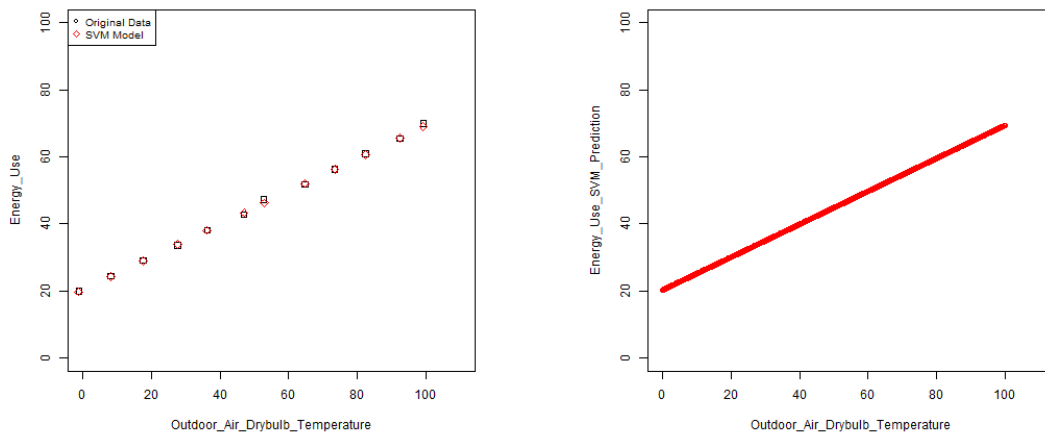
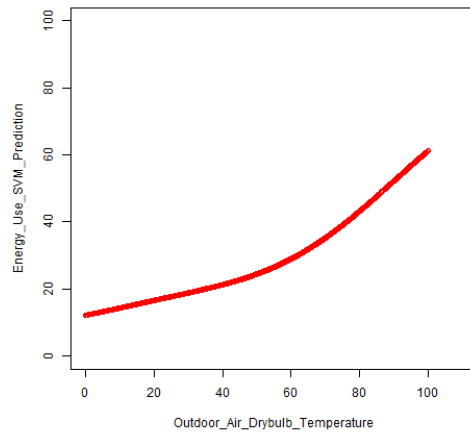(a) original data and SVM regression points

(b) SVM model continuous plot

CV (%) = 1.11, R^2 = 0.9991

Figure C 10 - SVM Regression Observation (Ideal Linear Monthly Data)

C.1.2.2 10% Noisy Input
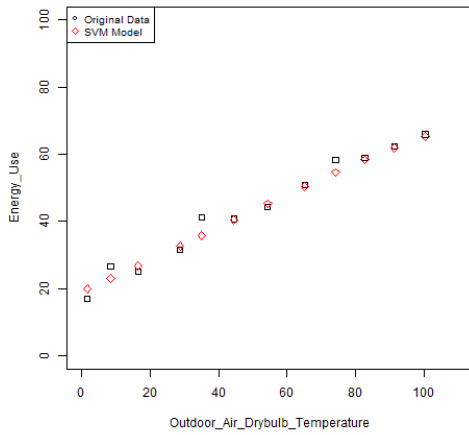
(a) original data and SVM regression points　　　(b) SVM model continuous plot
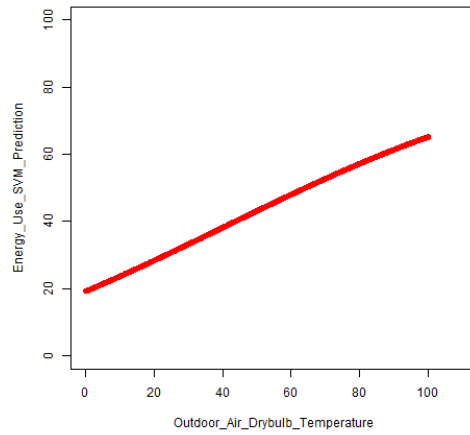
$$CV (\%) = 5.91, R^2 = 0.9781$$

Figure C 11 - SVM Regression Observation (10% Noisy Linear Monthly Data)

C.1.2.3 30% Noisy Input
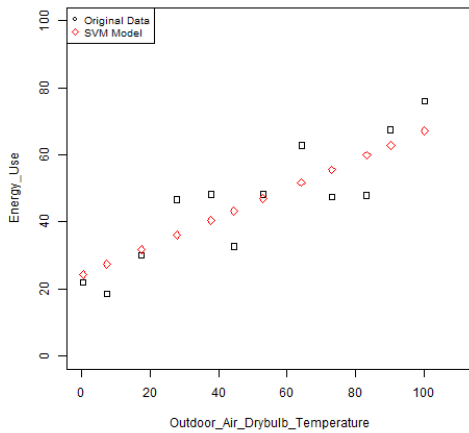
(a) original data and SVM regression points　　　(b) SVM model continuous plot
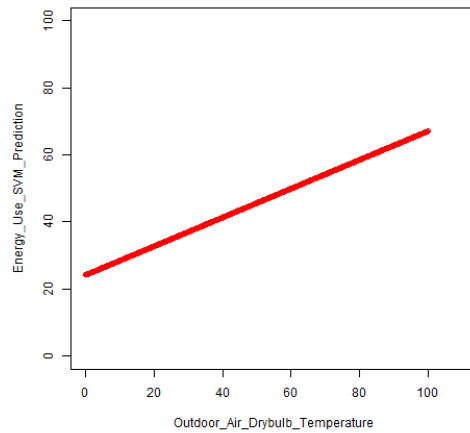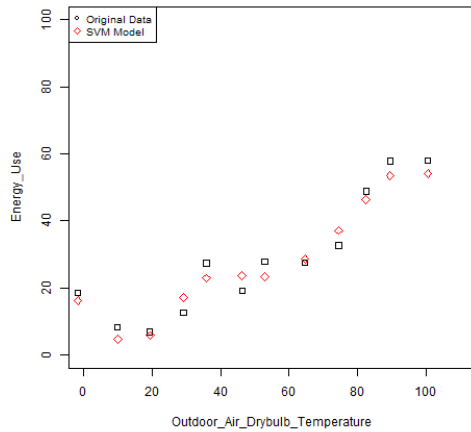
$$CV (\%) = 18.71, R^2 = 0.7708$$

Figure C 12 - SVM Regression Observation (30% Noisy Linear Monthly Data)

C.1.2.4 50% Noisy Input
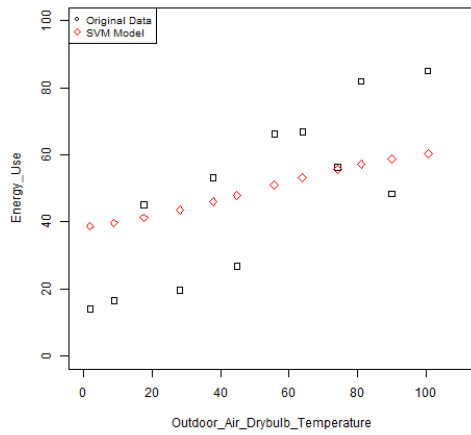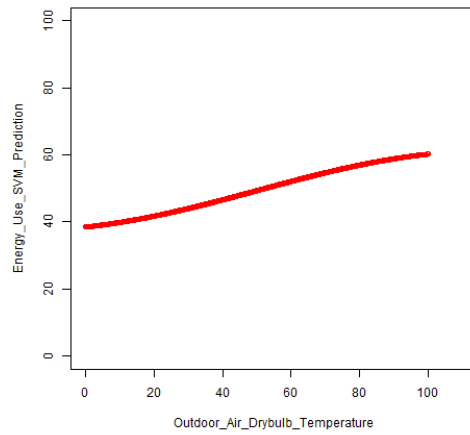
(a) original data and SVM regression points
(b) SVM model continuous plot

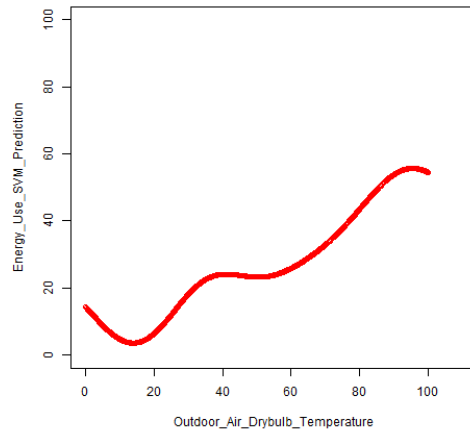CV (%) = 38.5, R^2 = 0.6609

Figure C 13 - SVM Regression Observation (50% Noisy Linear Monthly Data)

C.1.2.5 100% Noisy Input
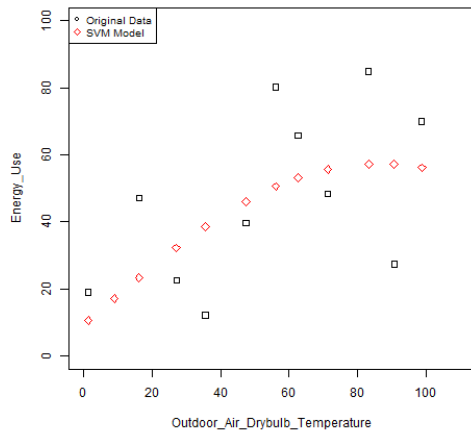
(a) original data and SVM regression points
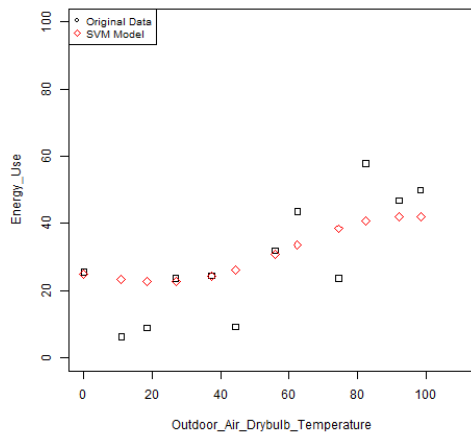(b) SVM model continuous plot
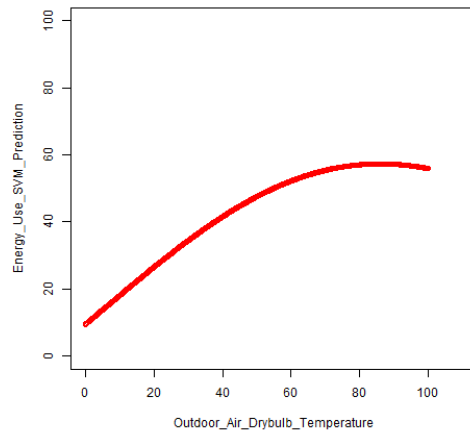
CV (%) = 52.72, R^2 = 0.4710

Figure C 14 - SVM Regression Observation (100% Noisy Linear Monthly Data)

C.1.3 Daily Data (with linear SVM)

C.1.3.1 Ideal Input
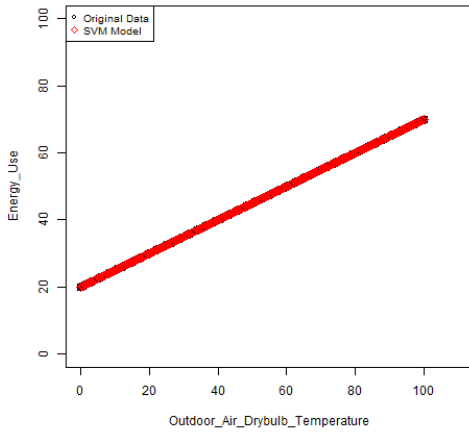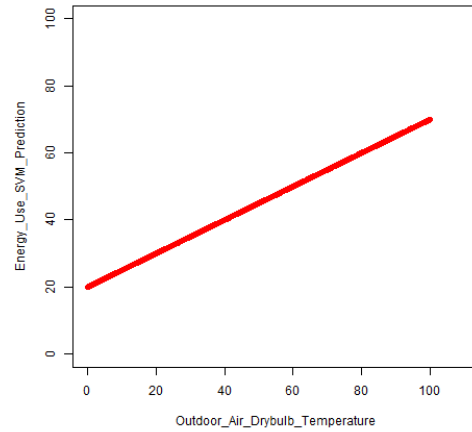
(a) original data and SVM regression points	(b) SVM model continuous plot

CV (%) = 1.11, R^2 = 0.9991, number of support vector = 2

Figure C 15 - Linear SVM Regression Observation (Ideal Linear Daily Data)

C.1.3.2 10% Noisy Input
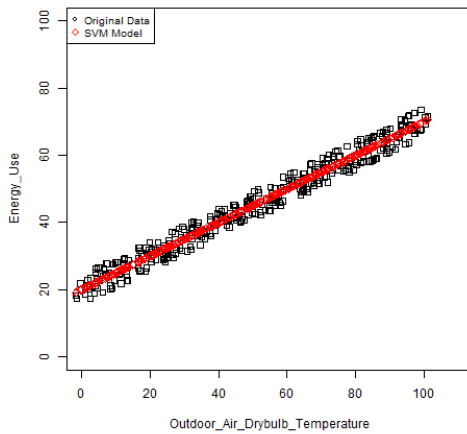
(a) original data and SVM regression points	(b) SVM model continuous plot
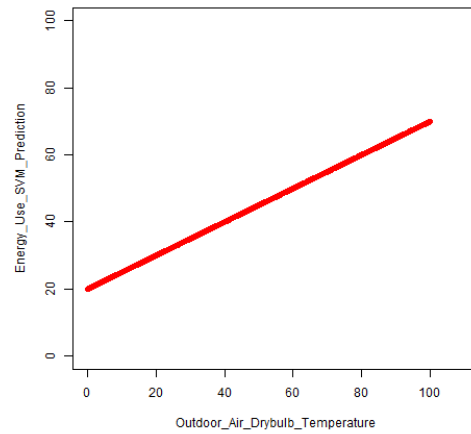
CV (%) = 5.91, R^2 = 0.9781

Figure C 16 - Linear SVM Regression Observation (10% Noisy Linear Daily Data)

C.1.3.3 30% Noisy Input
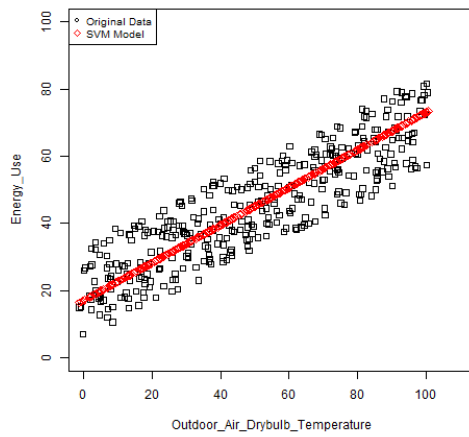
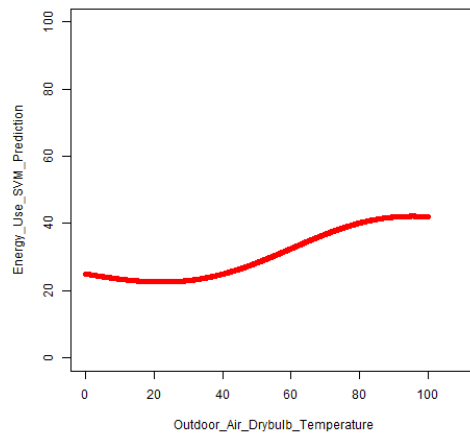(a) original data and SVM regression points   (b) SVM model continuous plot
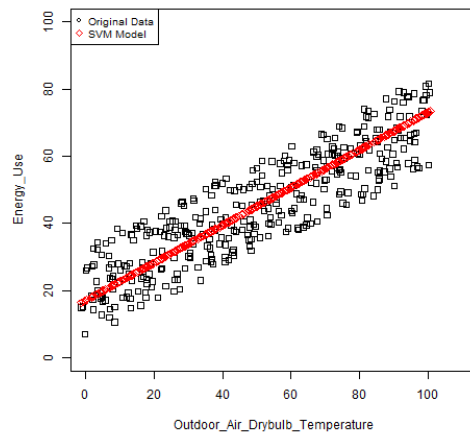
CV (%) = 18.71, R^2 = 0.7708

Figure C 17 - Linear SVM Regression Observation (30% Noisy Linear Daily Data)

C.1.3.4 50% Noisy Input
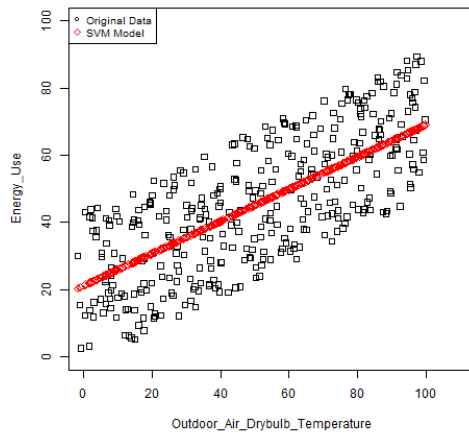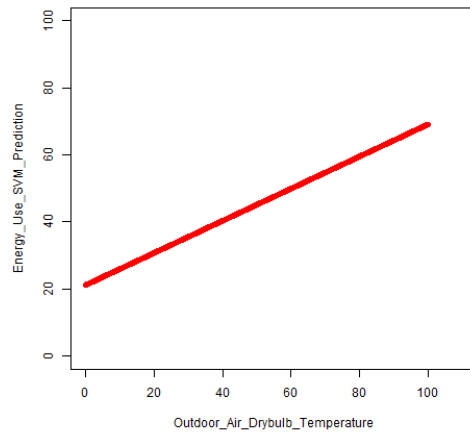
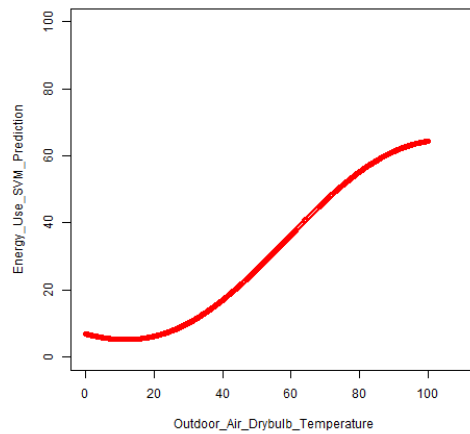(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 38.50, R^2 = 0.6609

Figure C 18 - Linear SVM Regression Observation (50% Noisy Linear Daily Data)

178

C.1.3.4 100% Noisy Input
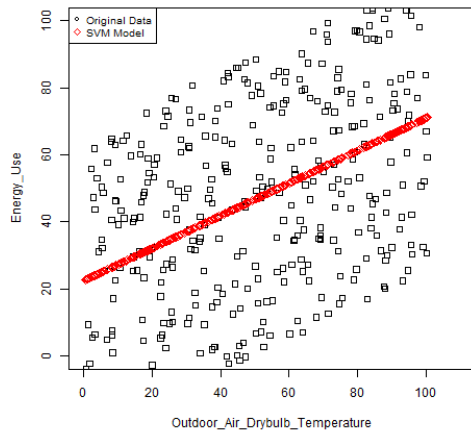
(a) original data and SVM regression points (b) SVM model continuous plot
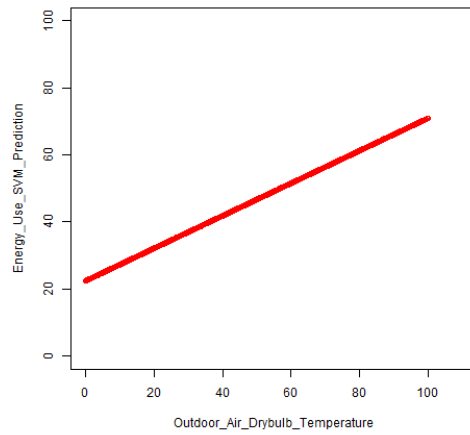
CV (%) = 52.72, R^2 = 0.4710

Figure C 19 - Linear SVM Regression Observation (100% Noisy Linear Daily Data)

C.1.4 Monthly Data (with linear SVM)

C.1.4.1 Ideal Input
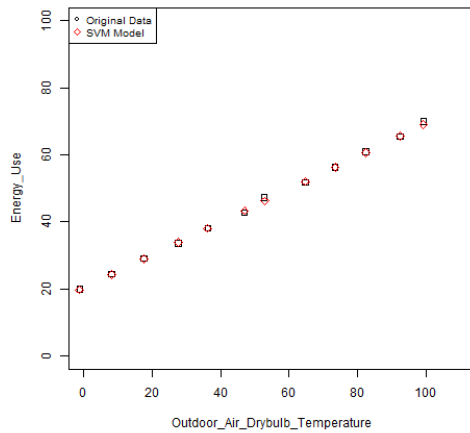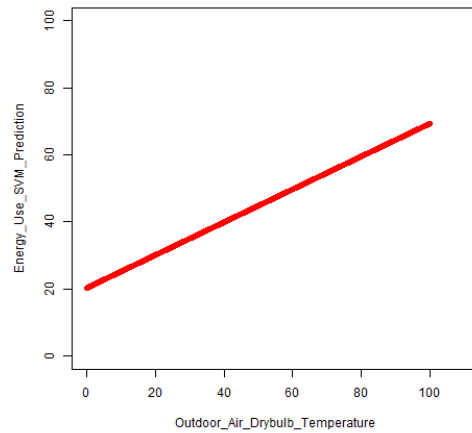
(a) original data and SVM regression points (b) SVM model continuous plot

CV (%) = 1.13, R^2 = 0.9991

Figure C 20 - Linear SVM Regression Observation (Ideal Linear Monthly Data)

C.1.4.2 10% Noisy Input
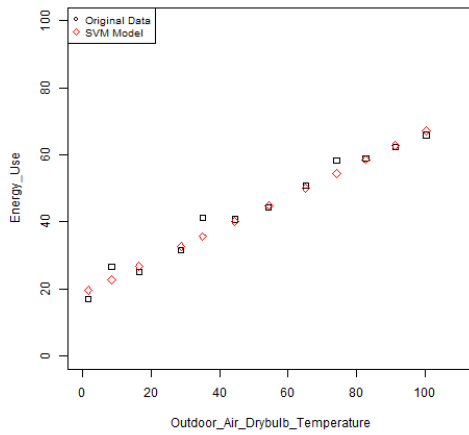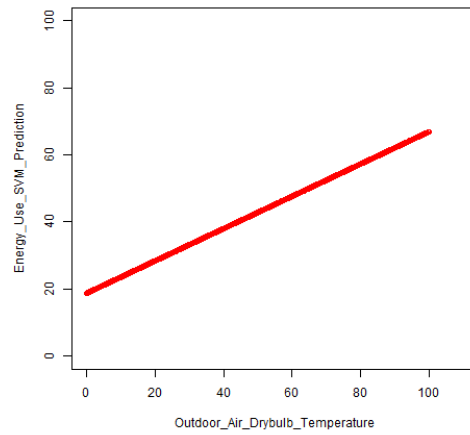
(a) original data and SVM regression points　　(b) SVM model continuous plot
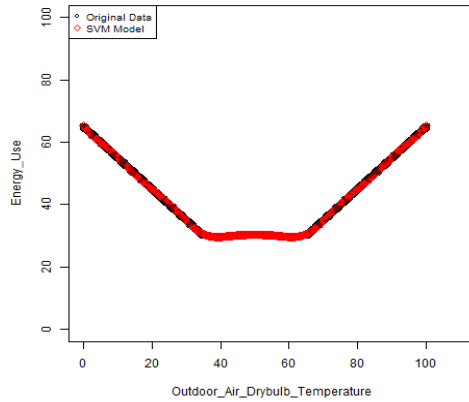
CV (%) = 5.95, R^2 = 0.9764

Figure C 21 - Linear SVM Regression Observation (10% Noise Linear Monthly Data)

C.1.4.3 30% Noisy Input
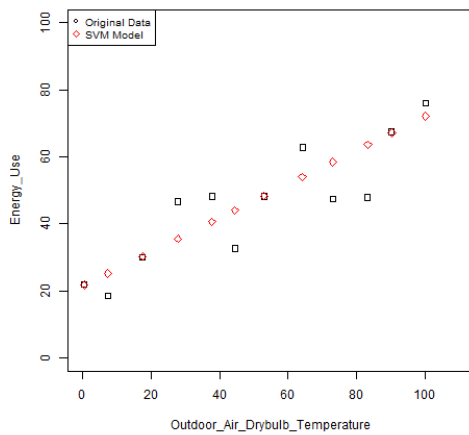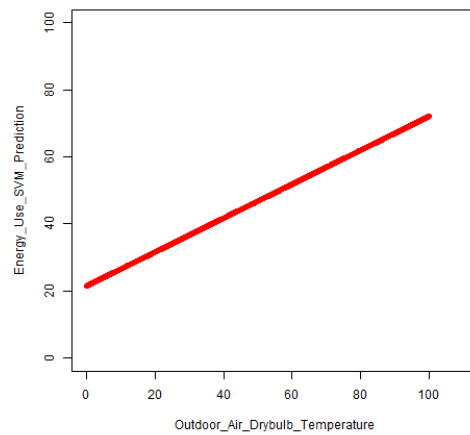
(a) original data and SVM regression points　　(b) SVM model continuous plot

CV (%) = 18.39, R^2 = 0.7709

Figure C 22 - Linear SVM Regression Observation (30% Noise Linear Monthly Data)

C.1.4.4 50% Noisy Input
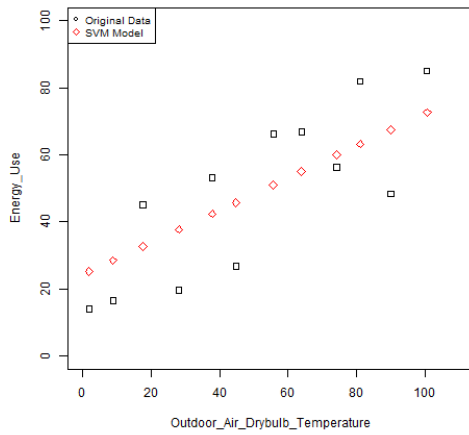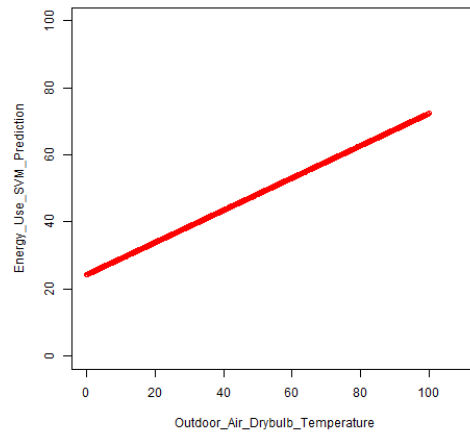
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 30.95, R^2 = 0.6634

Figure C 23 - Linear SVM Regression Observation (50% Noise Linear Monthly Data)

C.1.4.5 100% Noisy Input
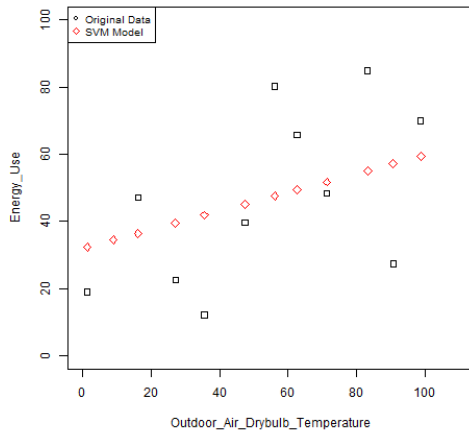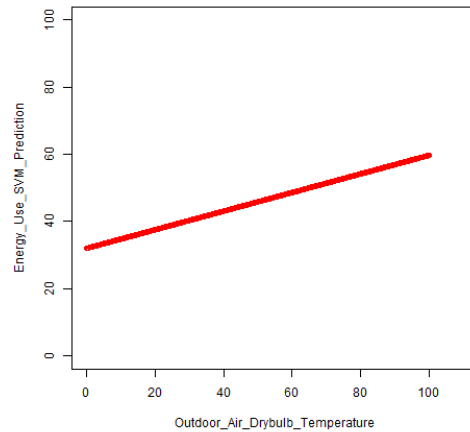
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 54.73, R^2 = 0.4226

Figure C 24 - Linear SVM Regression Observation (100% Noise Linear Monthly Data)

C.2 Piecewise Linear Input Data (Two pieces)

C.2.1 Daily Data (Type 1)

C.2.1.1 Ideal Input
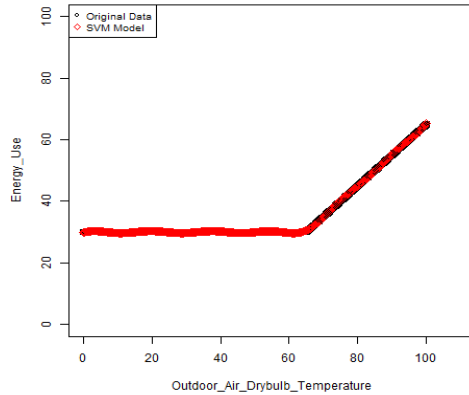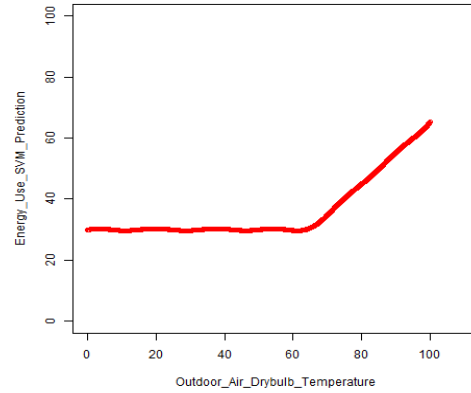
(a) original data and SVM regression   (b) SVM model continuous plot

$$CV (\%) = 0.63, R^2 = 0.9995$$

Figure C 25 - SVM Regression Observation (Ideal Piecewise (3P) Linear Daily Data)

C.2.1.2 10% Noisy Input

1*)* homoscedastic noise
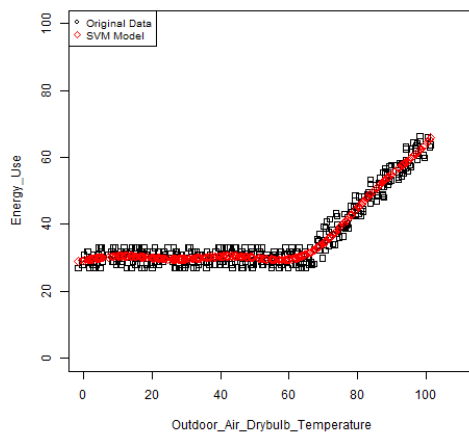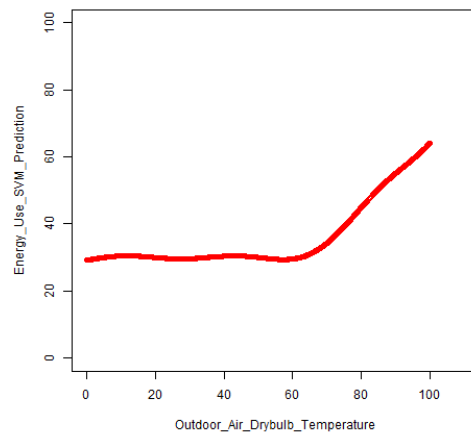
(a) original data and SVM regression points   (b) SVM model continuous plot

$$CV (\%) = 6.02, R^2 = 0.9568$$

Figure C 26 - SVM Regression Observation (10% Noise Piecewise (3P) Linear Daily Data 1)

2*)* heteroscedastic noise
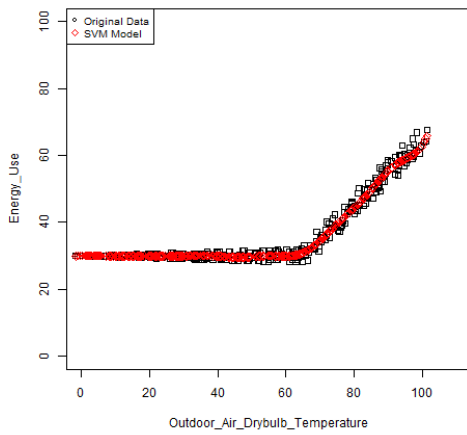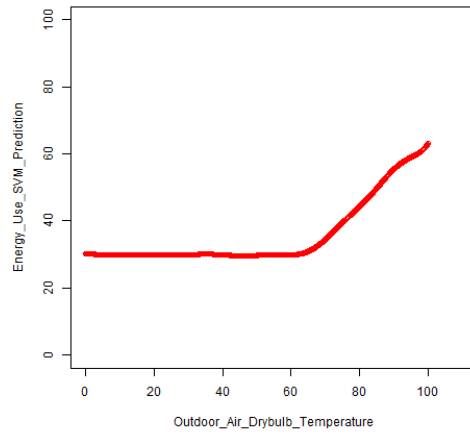
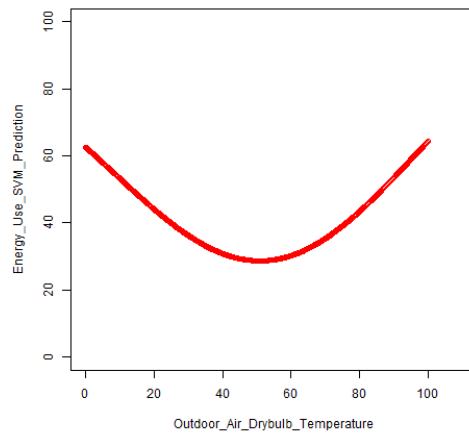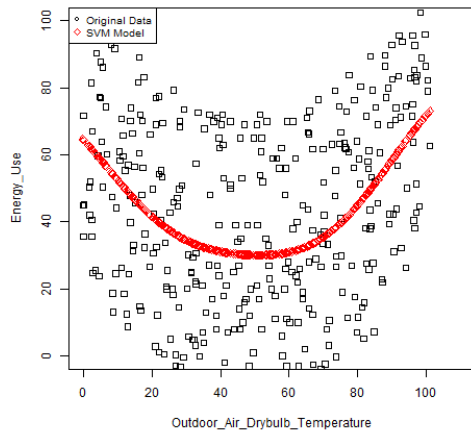(a) original data and SVM regression points   (b) SVM model continuous plot

$CV (\%) = 3.66, R^2 = 0.9841$

Figure C 27 - SVM Regression Observation (10% Noise Piecewise (3P) Linear Daily Data 2)

C.2.1.3 30% Noisy Input

1*)* homoscedastic noise
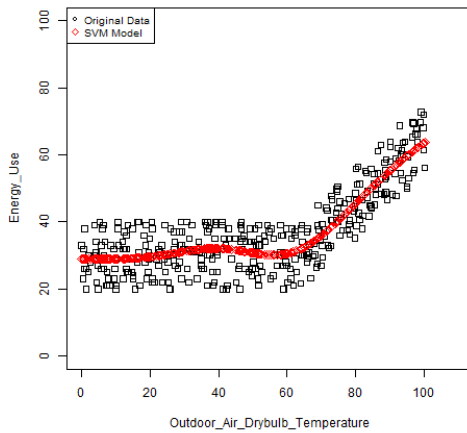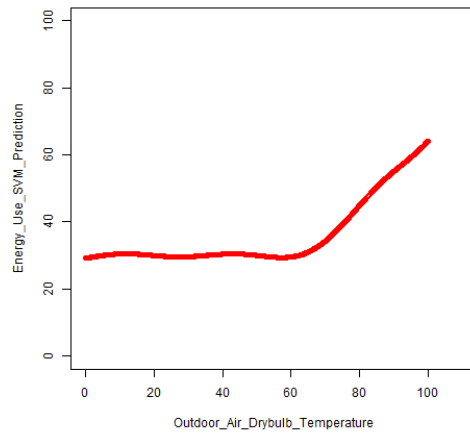
(a) original data and SVM regression points   (b) SVM model continuous plot

$CV (\%) = 16.27, R^2 = 0.7513$

Figure C 28 - SVM Regression Observation (30% Noise Piecewise (3P) Linear Daily Data 1)

2*)* heteroscedastic noise
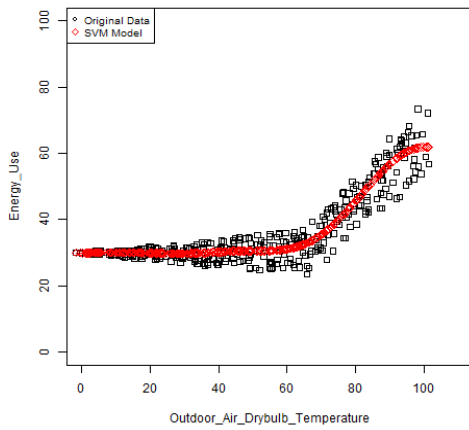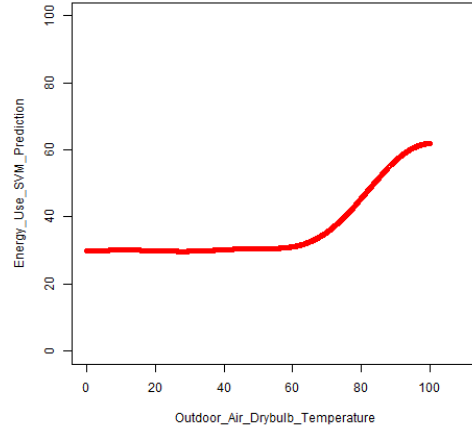
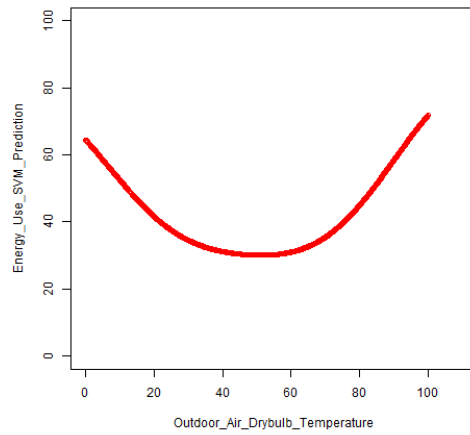(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 9.89, R^2 = 0.8884

Figure C 29 - SVM Regression Observation (30% Noise Piecewise (3P) Linear Daily Data 2)

C.2.1.4 50% Noisy Input

1*)* homoscedastic noise
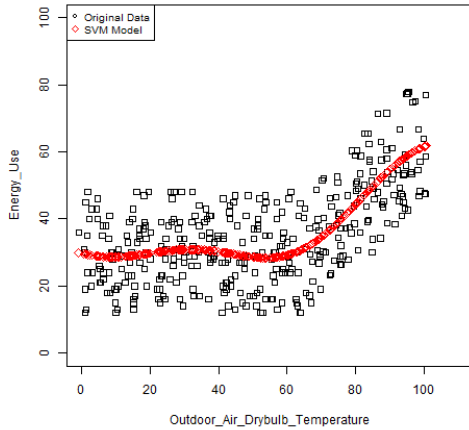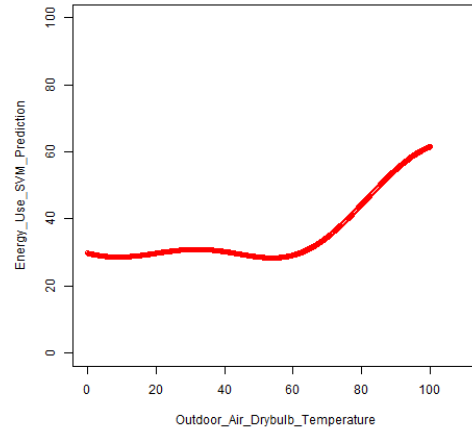
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 28.67, R^2 = 0.4735

Figure C 30 - SVM Regression Observation (50% Noise Piecewise (3P) Linear Daily Data 1)

2*)* heteroscedastic noise
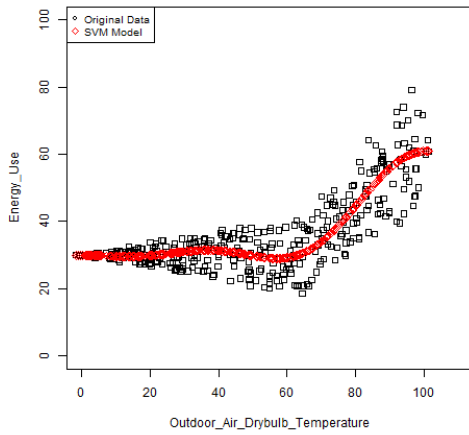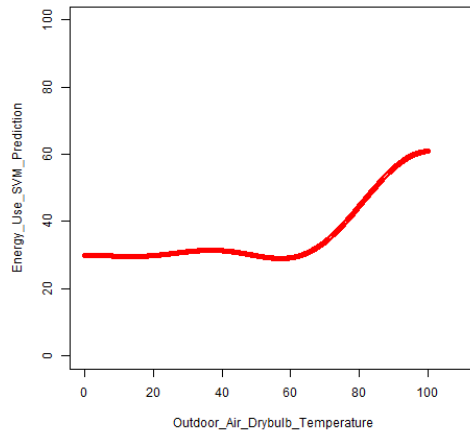
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 15.95, R^2 = 0.7437

Figure C 31 - SVM Regression Observation (50% Noise Piecewise (3P) Linear Daily Data 2)

C.2.1.5 100% Noisy Input

1) homoscedastic noise
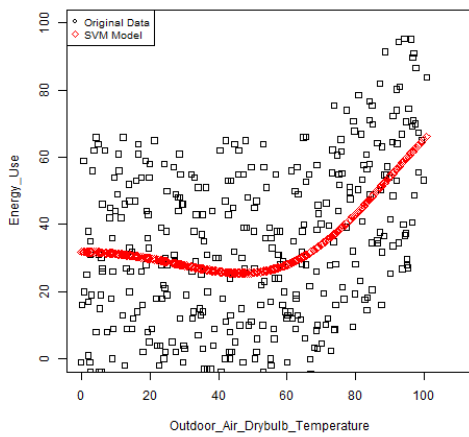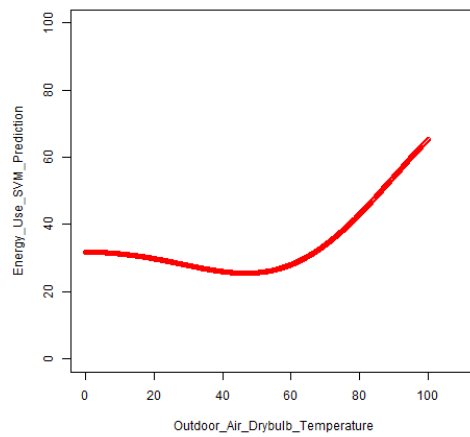
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 89.81, R^2 = 0.2314

Figure C 32 - SVM Regression Observation (100% Noise Piecewise (3P) Linear Daily Data 1)

2) heteroscedastic noise
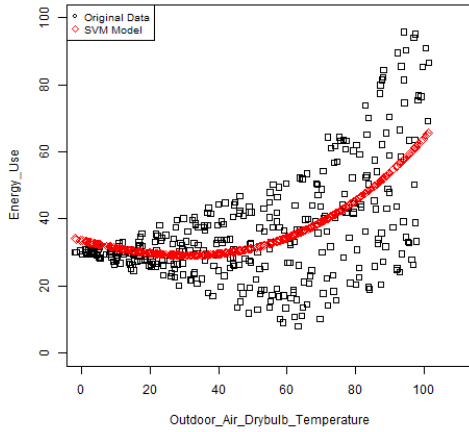
(a) original data and SVM regression points
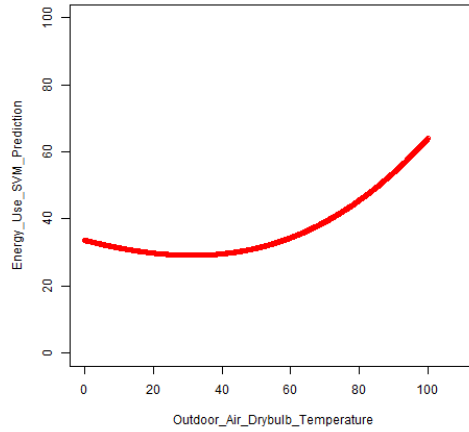
(b) SVM model continuous plot

CV (%) = 35.63, R^2 = 0.3648

Figure C 33 - SVM Regression Observation (100% Noise Piecewise (3P) Linear Daily Data 2)

C.2.2 Monthly Data (Type 1)
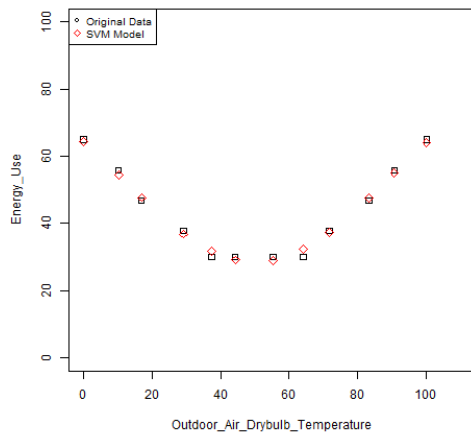
C.2.2.1 Ideal Input
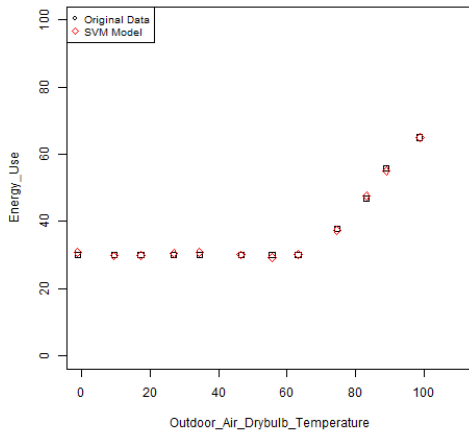
(a) original data and SVM regression points
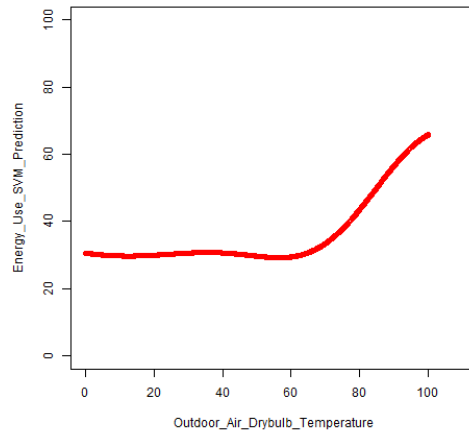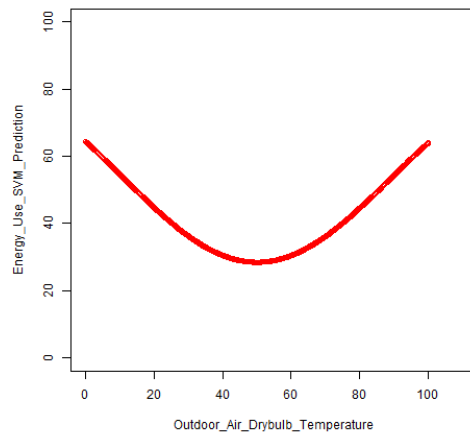
(b) SVM model continuous plot

CV (%) = 1.48, R^2 = 0.9980

Figure C 34 - SVM Regression Observation (Ideal Piecewise (3P) Linear Monthly Data)

C.2.2.2 10% Noisy Input
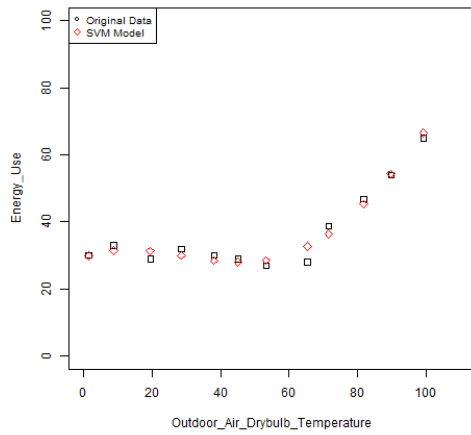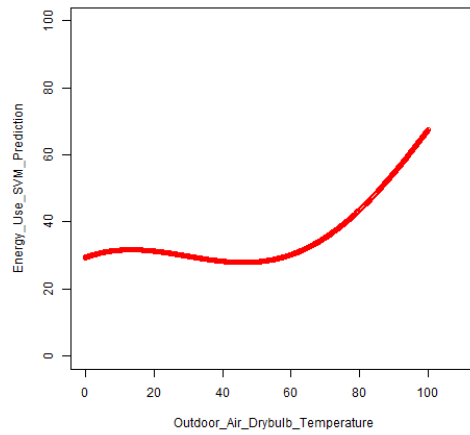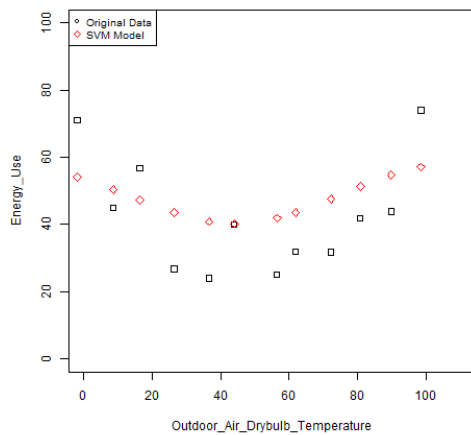

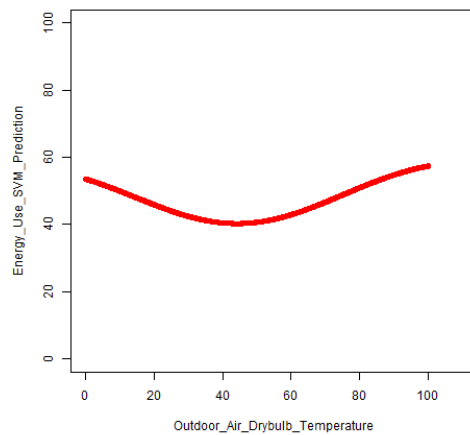(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 5.72, R^2 = 0.9703

Figure C 35 - SVM Regression Observation (10% Noise Piecewise (3P) Linear Monthly Data)

C.2.2.3 30% Noisy Input
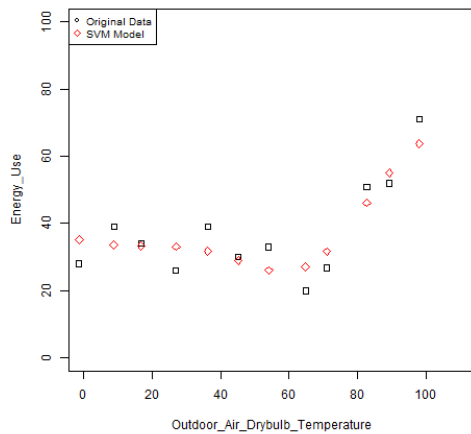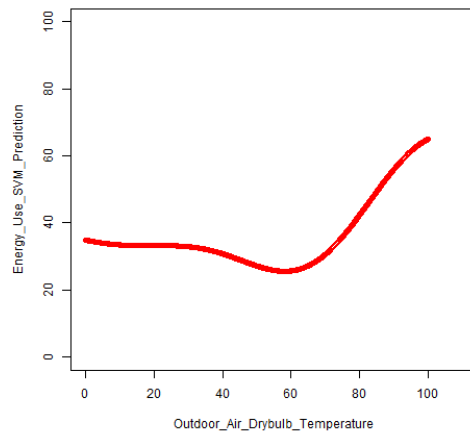
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 16.04, R^2 = 0.8352

Figure C 36 - SVM Regression Observation (30% Noise Piecewise (3P) Linear Monthly Data)

C.2.2.4 50% Noisy Input
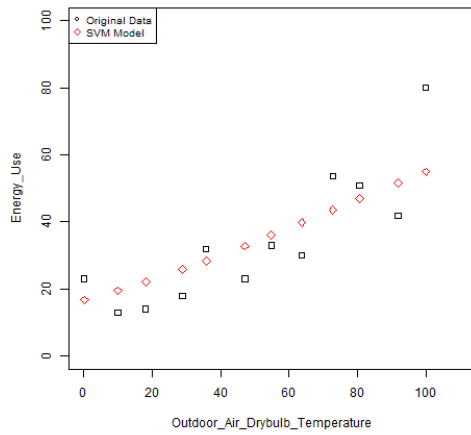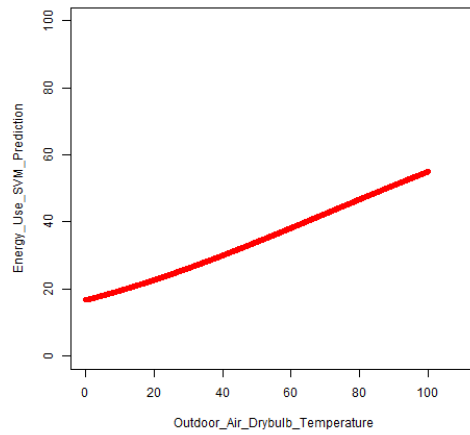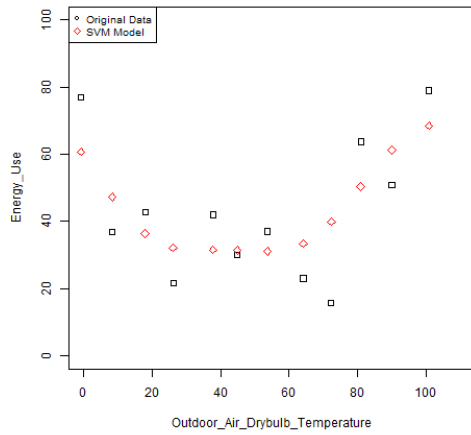

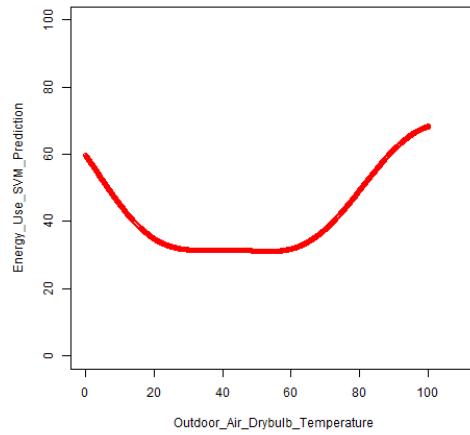(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 30.70, R^2 = 0.7455

Figure C 37 - SVM Regression Observation (50% Noise Piecewise (3P) Linear Monthly Data)

C.2.2.5 100% Noisy Input
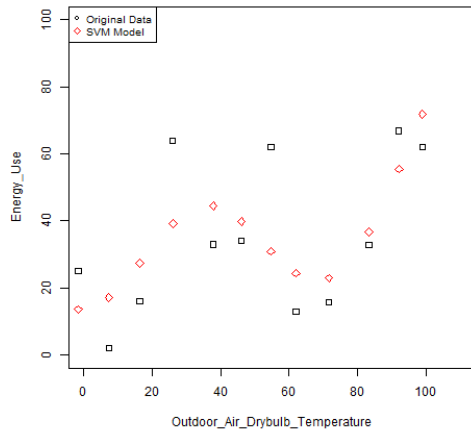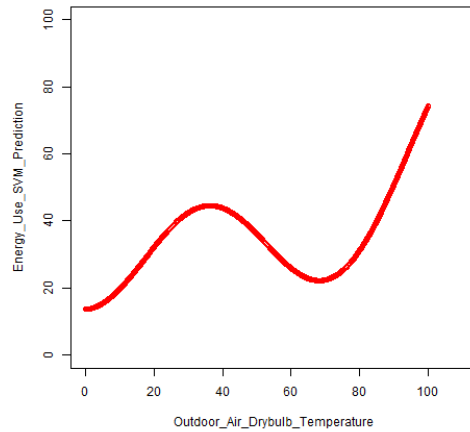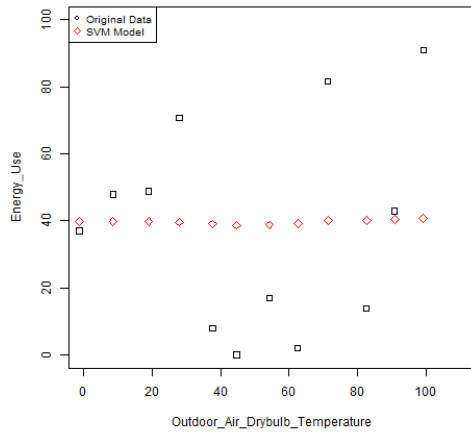

(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 44.00, R^2 = 0.5358

Figure C 38 - SVM Regression Observation (100% Noise Piecewise (3P) Linear Monthly Data)

C.2.3 Daily Data (Type 2)

C.2.3.1 Ideal Input
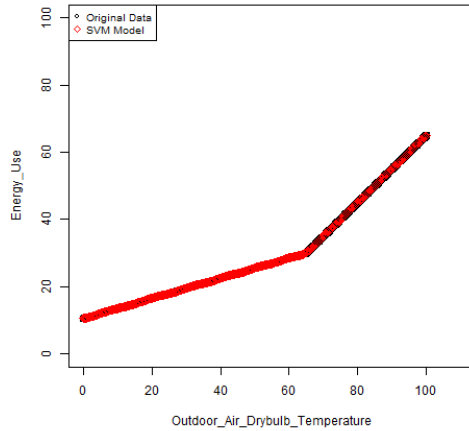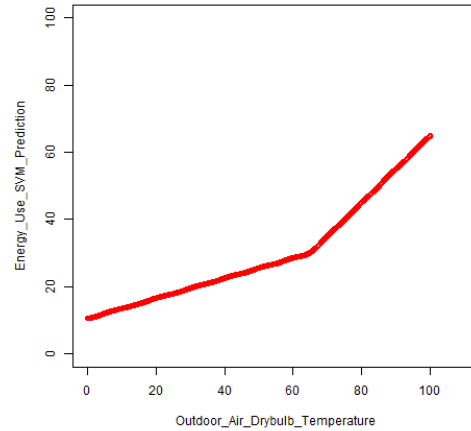
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 0.28, R^2 = 0.9999

Figure C 39 - SVM Regression Observation (Ideal Piecewise (4P) Linear Daily Data)

C.2.3.2 10% Noisy Input

1) homoscedastic noise
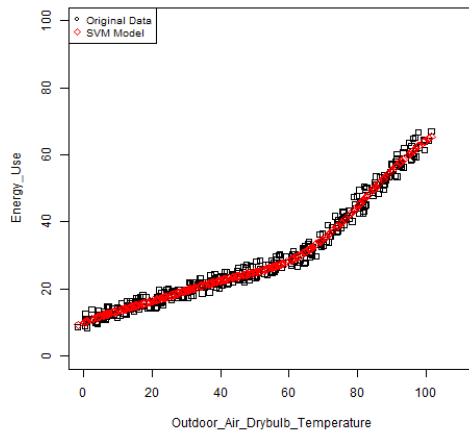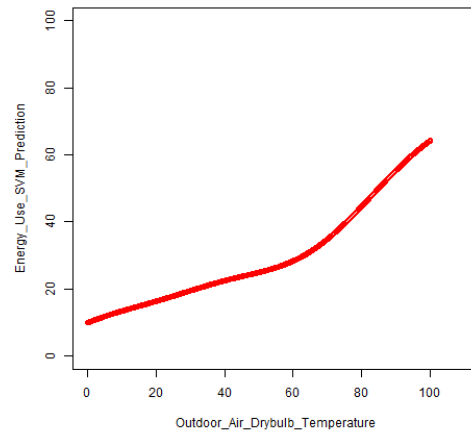
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 5.48, R^2 = 0.9886

Figure C 40 - SVM Regression Observation (10% Noise Piecewise (4P) Linear Daily Data 1)

2) heteroscedastic noise
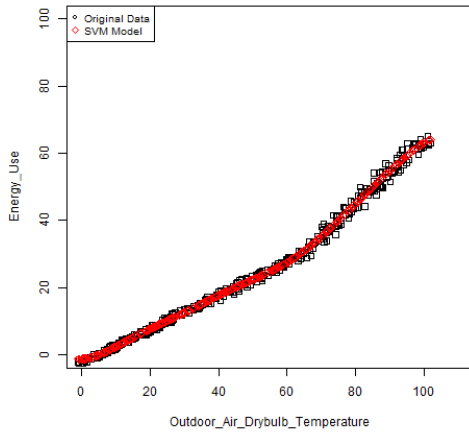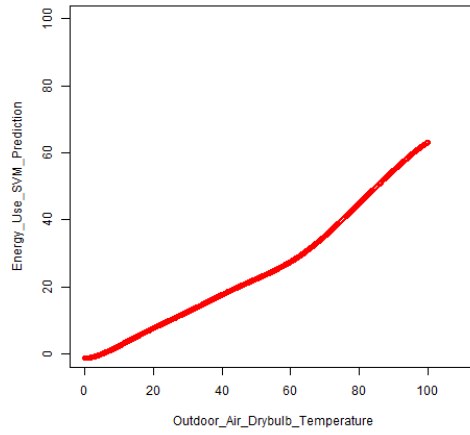
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 4.63, R^2 = 0.9960

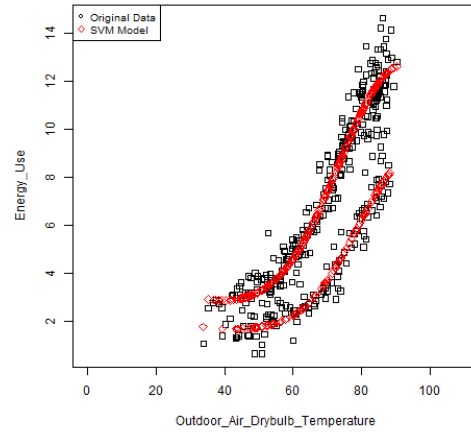Figure C 41 - SVM Regression Observation (10% Noise Piecewise (4P) Linear Daily Data 2)

C.2.3.3 30% Noisy Input

1) homoscedastic noise
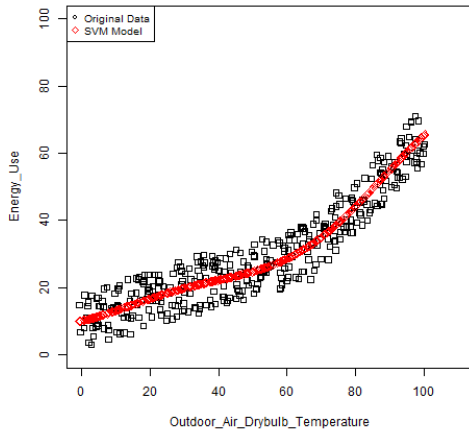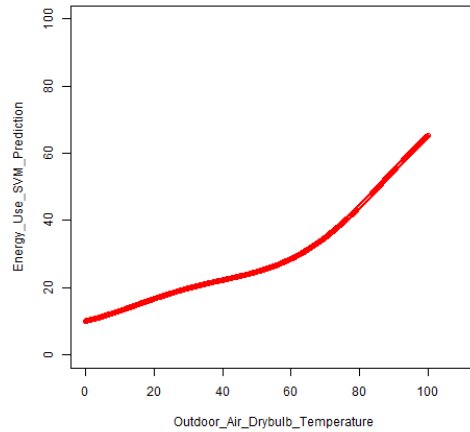
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 16.80, R^2 = 0.8971

Figure C 42 - SVM Regression Observation (30% Noise Piecewise (4P) Linear Daily Data 1)

2) heteroscedastic noise
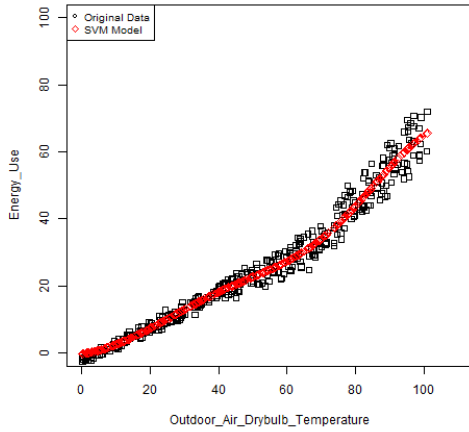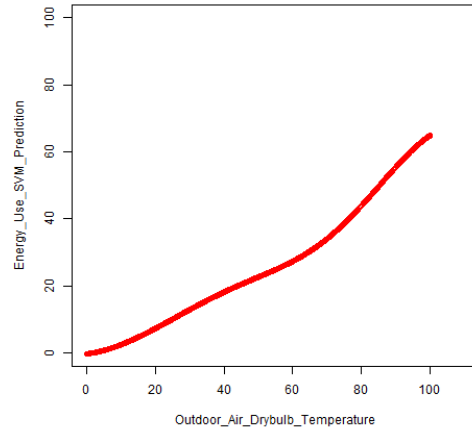
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 11.44, R^2 = 0.9763

Figure C 43 - SVM Regression Observation (30% Noise Piecewise (4P) Linear Daily Data 2)

C.2.3.4 50% Noisy Input

1) homoscedastic noise
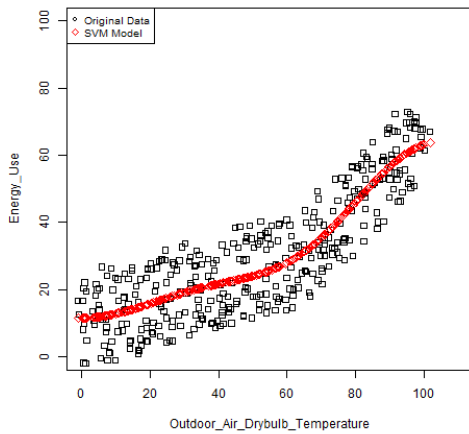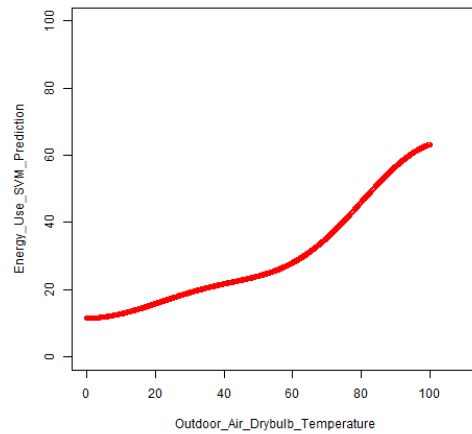
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 28.84, R^2 = 0.7743

Figure C 44 - SVM Regression Observation (50% Noise Piecewise (4P) Linear Daily Data 1)

2) heteroscedastic noise

(a) original data and SVM regression points　　　　(b) SVM model continuous plot

CV (%) = 18.64, R^2 = 0.9388

Figure C 45 - SVM Regression Observation (50% Noise Piecewise (4P) Linear Daily Data 2)

C.2.3.5 100% Input

1) homoscedastic noise

(a) original data and SVM regression points　　　　(b) SVM model continuous plot

CV (%) = 58.53, R^2 = 0.4911

Figure C 46 - SVM Regression Observation (100% Noise Piecewise (4P) Linear Daily Data 1)

2) heteroscedastic noise

(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 39.40, R^2 = 0.7768

Figure C 47 - SVM Regression Observation (10% Noise Piecewise (4P) Linear Daily Data 2)

C.2.4 Monthly Data (Type 2)

C.2.4.1 Ideal Input

(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 3.12, R^2 = 0.9972

Figure C 48 - SVM Regression Observation (Ideal Piecewise (4P) Linear Monthly Data)

C.2.4.2 10% Noisy Input

| (a) original data and SVM regression points | (b) SVM model continuous plot |

$$CV (\%) = 5.45, R^2 = 0.9908$$

Figure C 49 - SVM Regression Observation (10% Noise Piecewise (4P) Linear Monthly Data)

C.2.4.3 30% Input

| (a) original data and SVM regression points | (b) SVM model continuous plot |

$$CV (\%) = 13.69, R^2 = 0.9588$$

Figure C 50 - SVM Regression Observation (30% Noise Piecewise (4P) Linear Monthly Data)

C.2.4.4 50% Input

(a) original data and SVM regression points   (b) SVM model continuous plot

$CV (\%) = 37.30, R^2 = 0.6813$

Figure C 51 - SVM Regression Observation (50% Noise Piecewise (4P) Linear Monthly Data)

C.2.4.5 100% Input
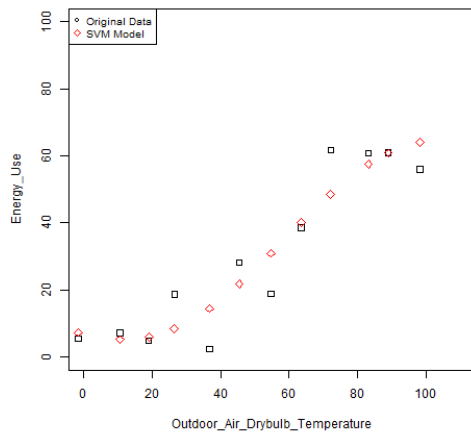
(a) original data and SVM regression points   (b) SVM model continuous plot

$CV (\%) = 26.16, R^2 = 0.8915$

Figure C 52 - SVM Regression Observation (100% Noise Piecewise (4P) Linear Monthly Data)

C.3 Special Case Input Data

C.3.1 Piecewise Linear Input Data (Three pieces)
C.3.1.1 Daily Data
C.3.1.1.1 Ideal Input
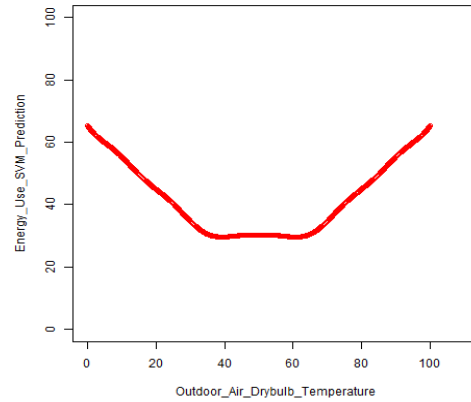
(a) original data and SVM regression points   (b) SVM model continuous plot

$$CV\ (\%) = 0.61,\ R^2 = 0.9995$$

Figure C 53 - SVM Regression Observation (Ideal Piecewise (5P) Linear Daily Data)

C.3.1.1.2 10% Noisy Input
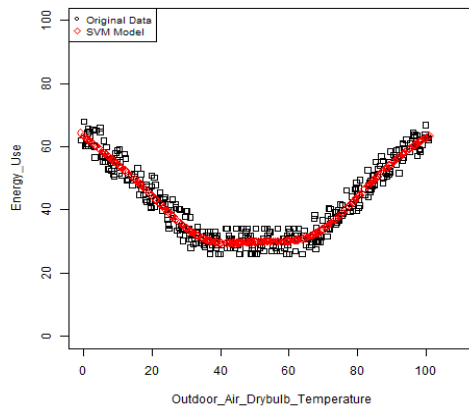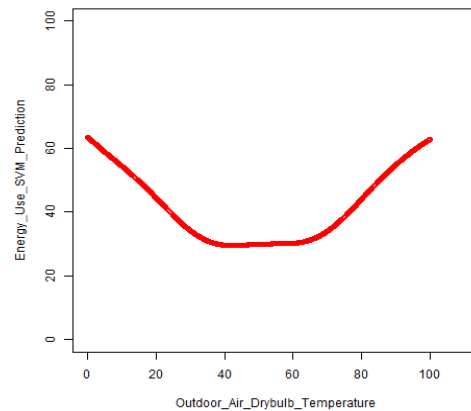
(a) original data and SVM regression points   (b) SVM model continuous plot

$$CV\ (\%) = 6.60,\ R^2 = 0.9454$$

Figure C 54 - SVM Regression Observation (10% Noise Piecewise (5P) Linear Daily Data)

C.3.1.1.3 30% Noisy Input
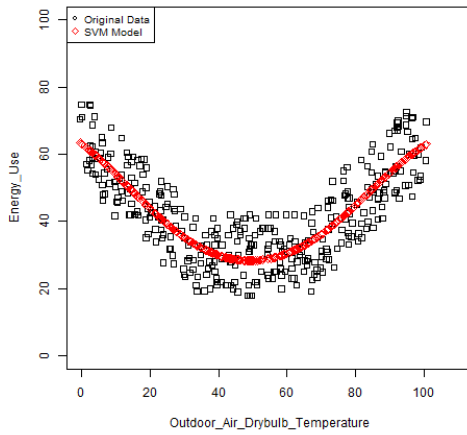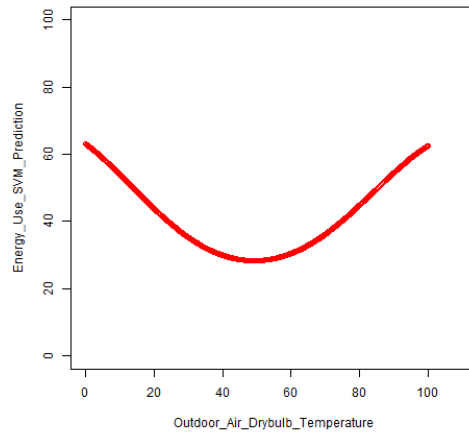
(a) original data and SVM regression points　　　(b) SVM model continuous plot

$$CV (\%) = 17.60, R^2 = 0.7088$$

Figure C 55 - SVM Regression Observation (30% Noise Piecewise (5P) Linear Daily Data)

C.3.1.1.4 50% Noisy Input
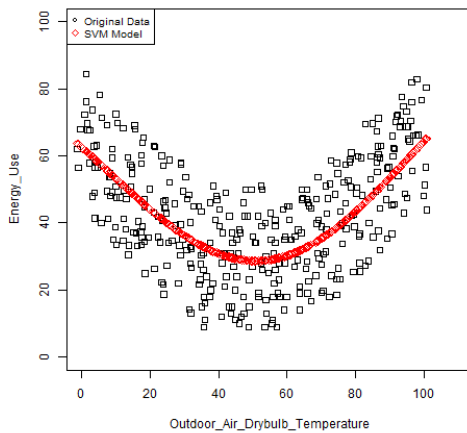
(a) original data and SVM regression points　　　(b) SVM model continuous plot

$$CV (\%) = 29.43, R^2 = 0.4839$$

Figure C 56 - SVM Regression Observation (50% Noise Piecewise (5P) Linear Daily Data)

C.3.1.1.5 100% Noisy Input

(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 59.04, R^2 = 0.2359

Figure C 57 - SVM Regression Observation (100% Noise Piecewise (5P) Linear Daily Data)

C.3.1.2 Monthly Data
C.3.1.2.1 Ideal Input
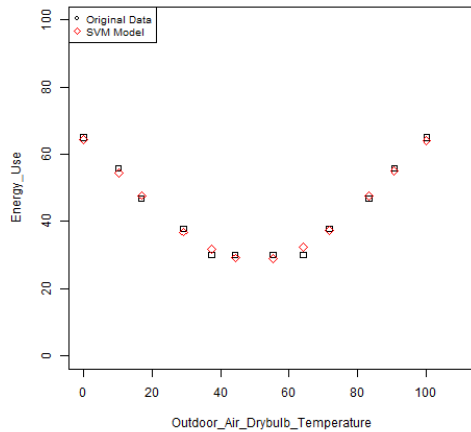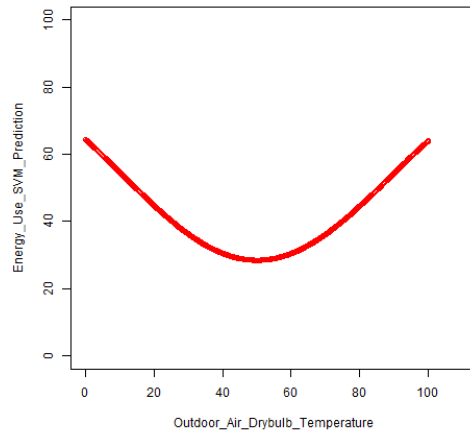
(a) original data and SVM regression points   (b) SVM model continuous plot

CV (%) = 2.79, R^2 = 0.9928

Figure C 58 - SVM Regression Observation (Ideal Piecewise (5P) Linear Monthly Data)

C.3.1.2.2 10% Noisy Input

(a) original data and SVM regression points

(b) SVM model continuous plot

CV (%) = 8.07, R^2 = 0.9542

Figure C 59 - SVM Regression Observation (10% Noise Piecewise (5P) Linear Monthly Data)

C.3.1.2.3 30% Noisy Input

(a) original data and SVM regression points

(b) SVM model continuous plot

CV (%) = 29.21, R^2 = 0.6131

Figure C 60 - SVM Regression Observation (30% Noise Piecewise (5P) Linear Monthly Data)

C.3.1.2.4 50% Noisy Input

(a) original data and SVM regression points

(b) SVM model continuous plot

CV (%) = 28.94, R^2 = 0.6573

Figure C 61 - SVM Regression Observation (50% Noise Piecewise (5P) Linear Monthly Data)

C.3.1.2.5 100% Noisy Input

(a) original data and SVM regression points
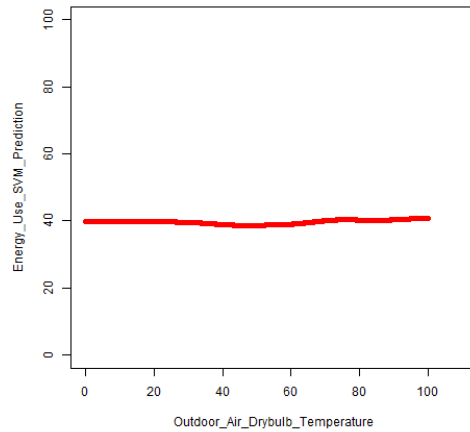
(b) SVM model continuous plot

CV (%) = 77.48, R^2 = 0.5259

Figure C 62 - SVM Regression Observation (100% Noise Piecewise (5P) Linear Monthly Data)

C.3.2 Scheduled Input Data

C.3.2.1 Weekday/Weekend Schedule
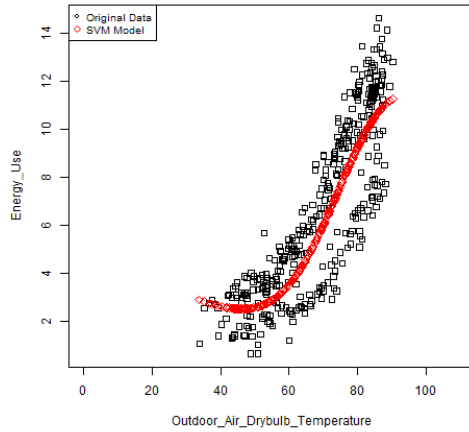
(a) data Input: OAT	(b) data input: OAT + schedule
CV (%) = 27.79, R^2 = 0.7602	CV (%) = 12.36, R^2 = 0.9483

Figure C 63 - SVM Regression Observation (Weekday/Weekend Schedule)

C.3.2.2 Semester/Summer Breaks/Winter Break/Holiday Schedule
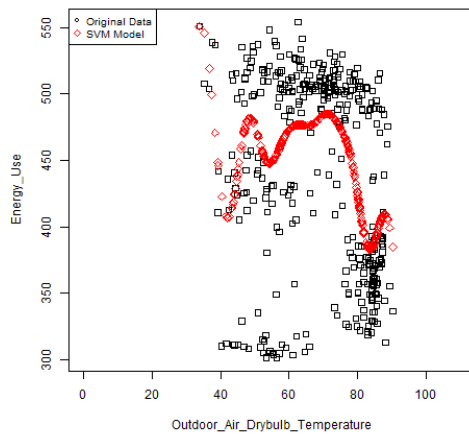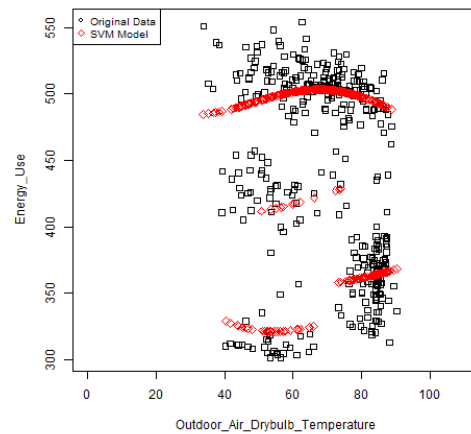
(a) data Input: OAT	(b) data input: OAT + schedule
CV (%) = 14.82, R^2 = 0.2444	CV (%) = 6.44, R^2 = 0.8592

Figure C 64 - SVM Regression Observation (Semester/Break/Holiday Schedule)

C.3.3 Scattering Input Data
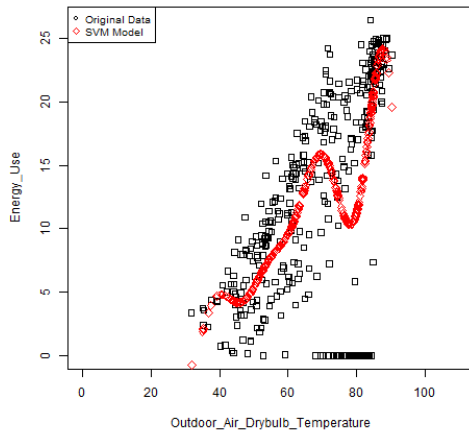
(a)

CV (%) = 47.48, R^2 = 0.4597
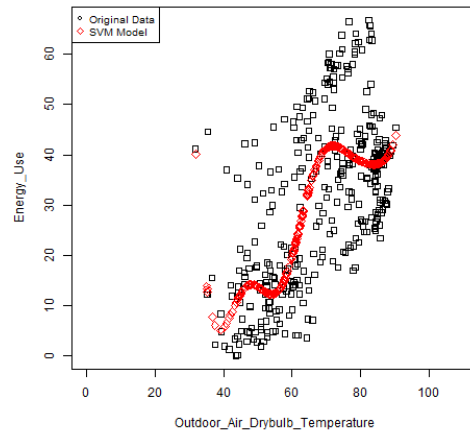
(b)

CV (%) = 40.77, R^2 = 0.4570
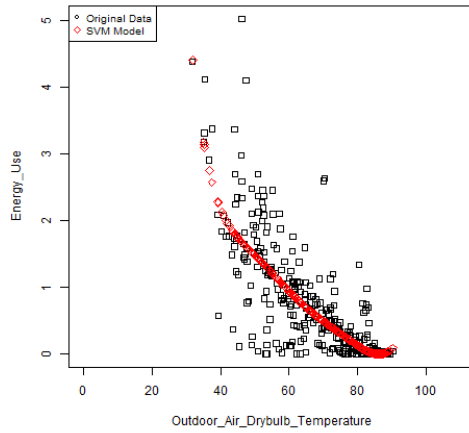
(c)

CV (%) = 76.58, R^2 = 0.6464
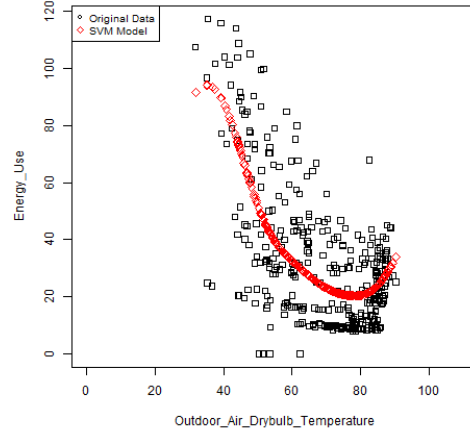
(d)

CV (%) = 55.12, R^2 = 0.4526

Figure C 65 - SVM Regression Observation (Scattering Data)

C.3.4 Hourly Input Data
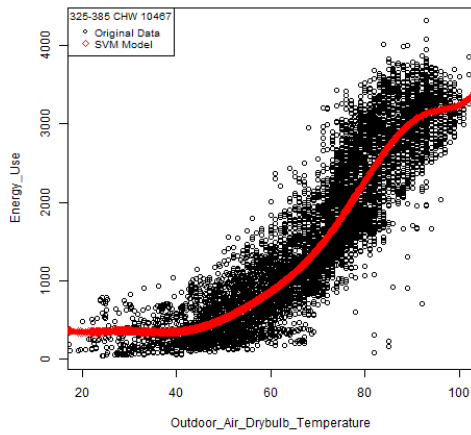
(a) CHW hourly data with OAT

CV (%) = 25.42, R^2 = 0.8392
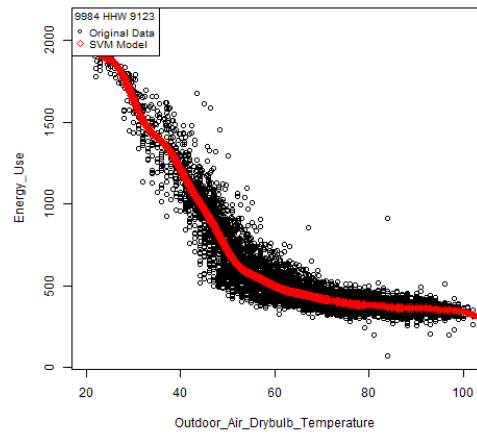
(b) HHW hourly data with OAT

CV (%) = 14.15, R^2 = 0.9292
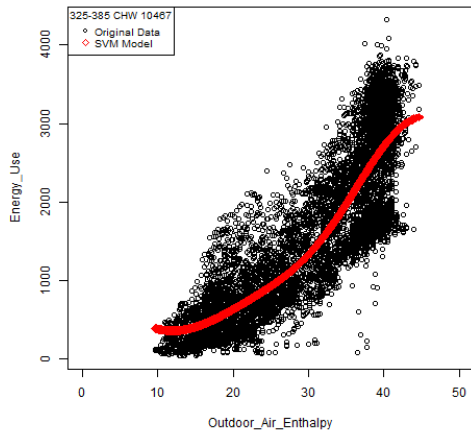
(c) CHW hourly data with OAE

CV (%) = 31.85, R^2 = 0.7461
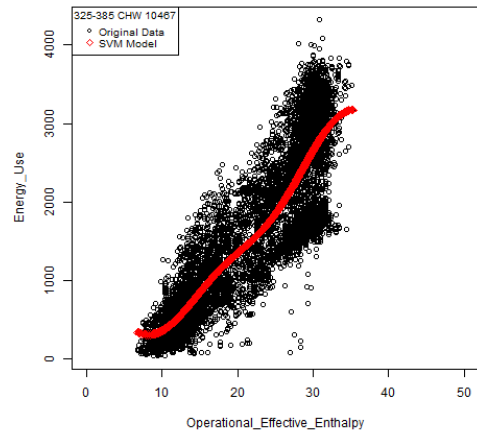
(d) CHW hourly data with OEE

CV (%) = 30.97, R^2 = 0.7585

Figure C 66 - SVM Regression Observation (Hourly Data)

C.3.5 Extraordinary Input Data
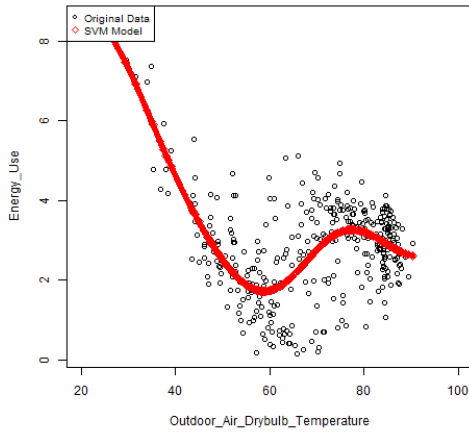
(a) chilled water hourly data

CV (%) = 33.61, R^2 = 0.4816
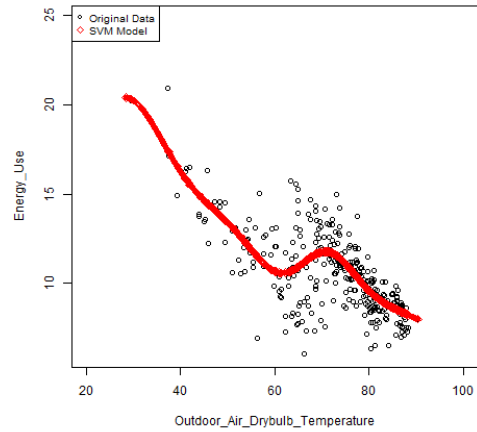
(b) heating hot water hourly data

CV (%) = 14.09, R^2 = 0.6238
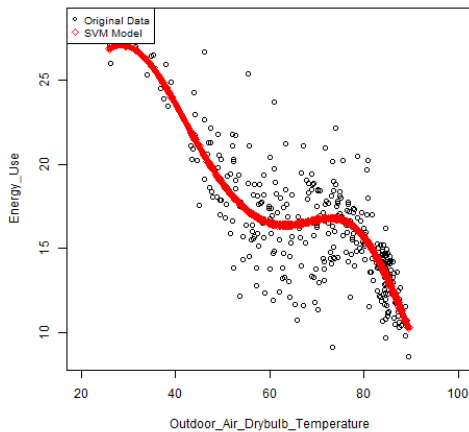
(a) chilled water hourly data

CV (%) = 12.78, R^2 = 0.6605
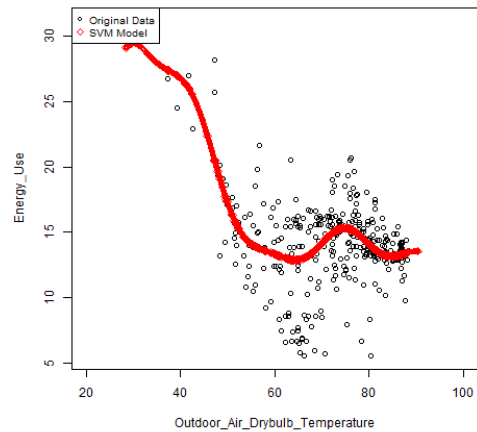
(b) heating hot water hourly data

CV (%) = 18.38, R^2 = 0.4573

Figure C 67 - SVM Regression Observation (Data with Break Points)

APPENDIX D. SVM, CPLR REGRESSION PLOTS

D.1 Daily

D.1.1 CHW - OAT
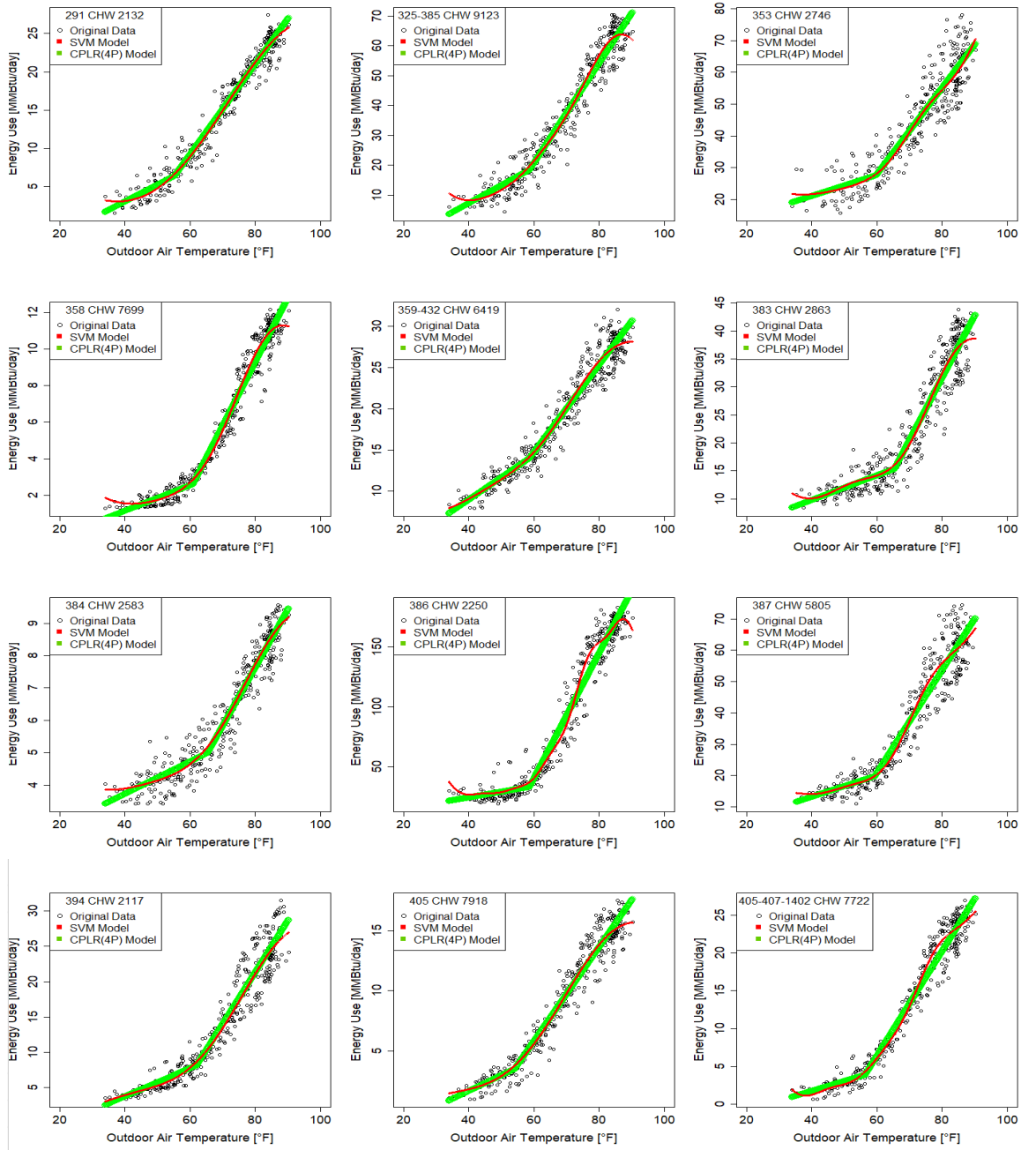
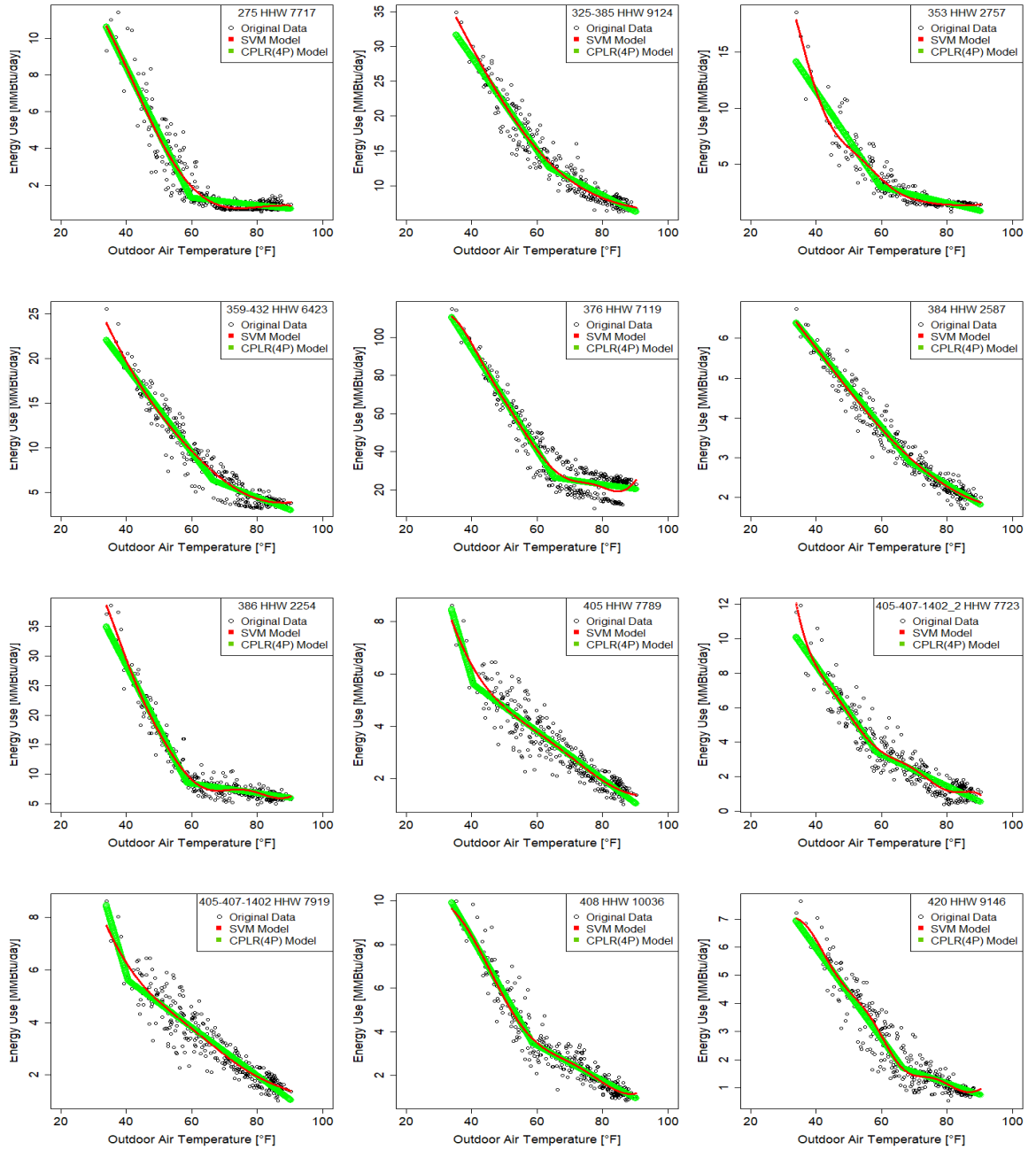

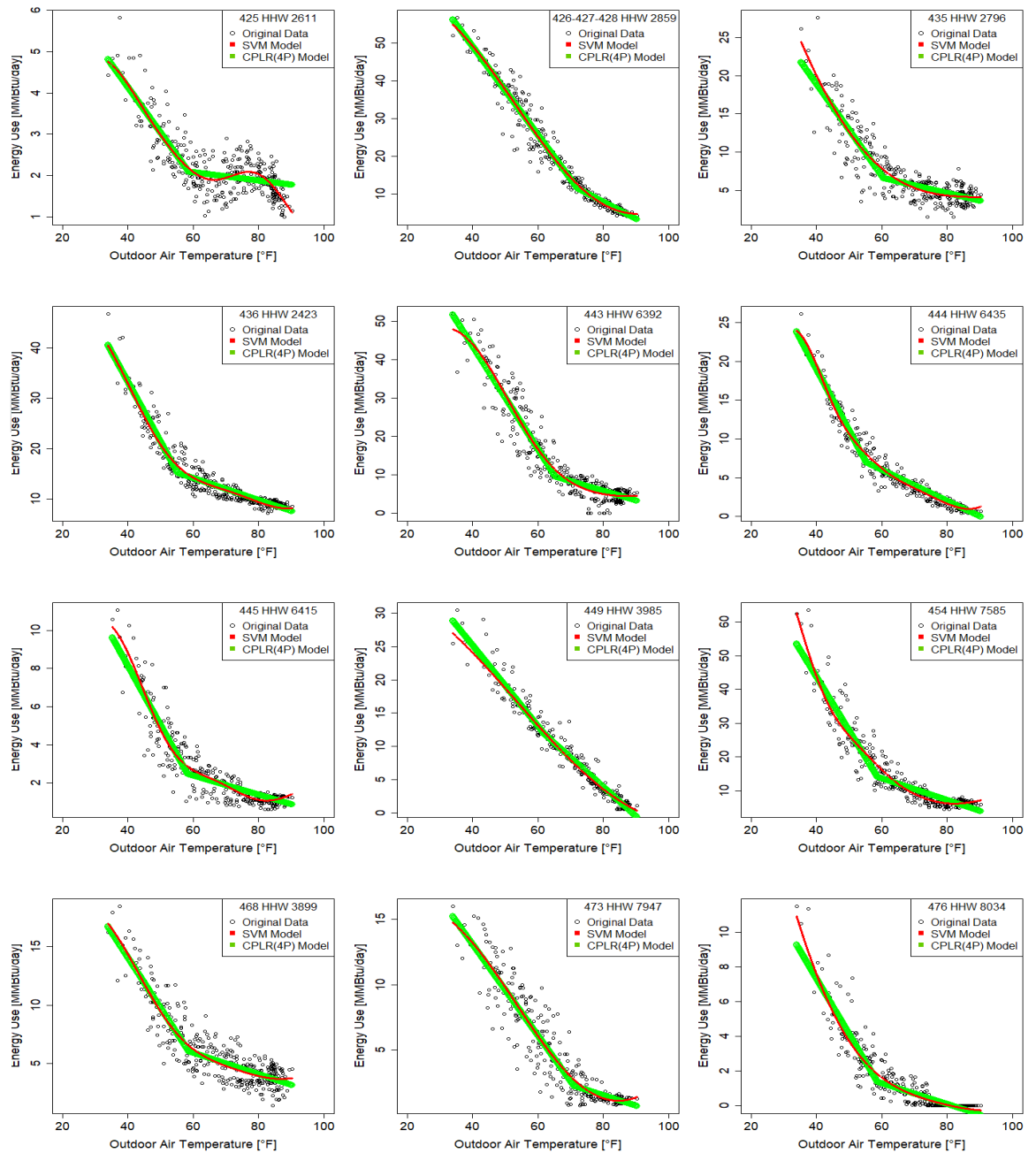Figure D 1 - SVM and CPLR (4P) Regression Plots (CHW-OAT) Sample No. 1 – No. 12 (Daily)
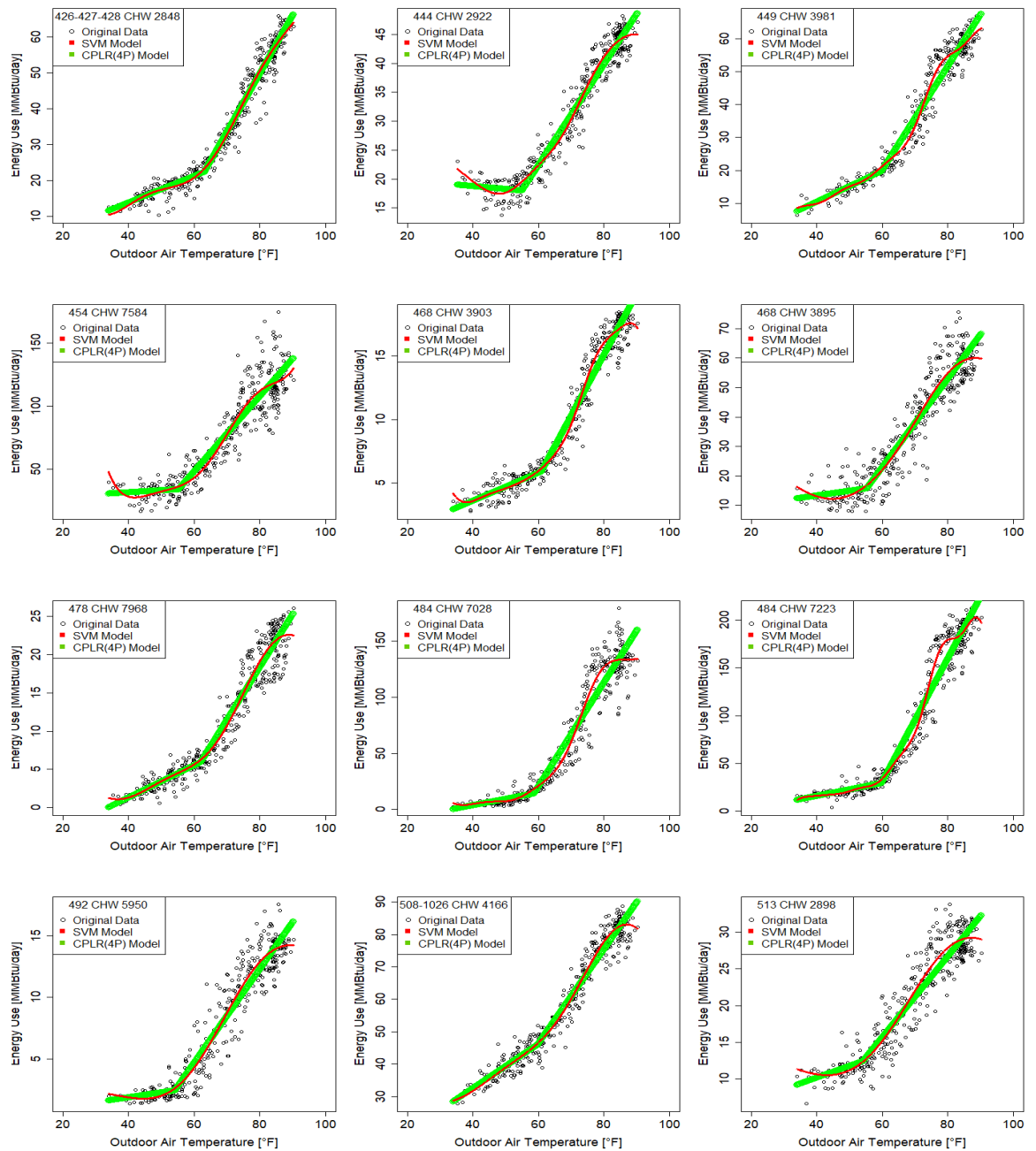
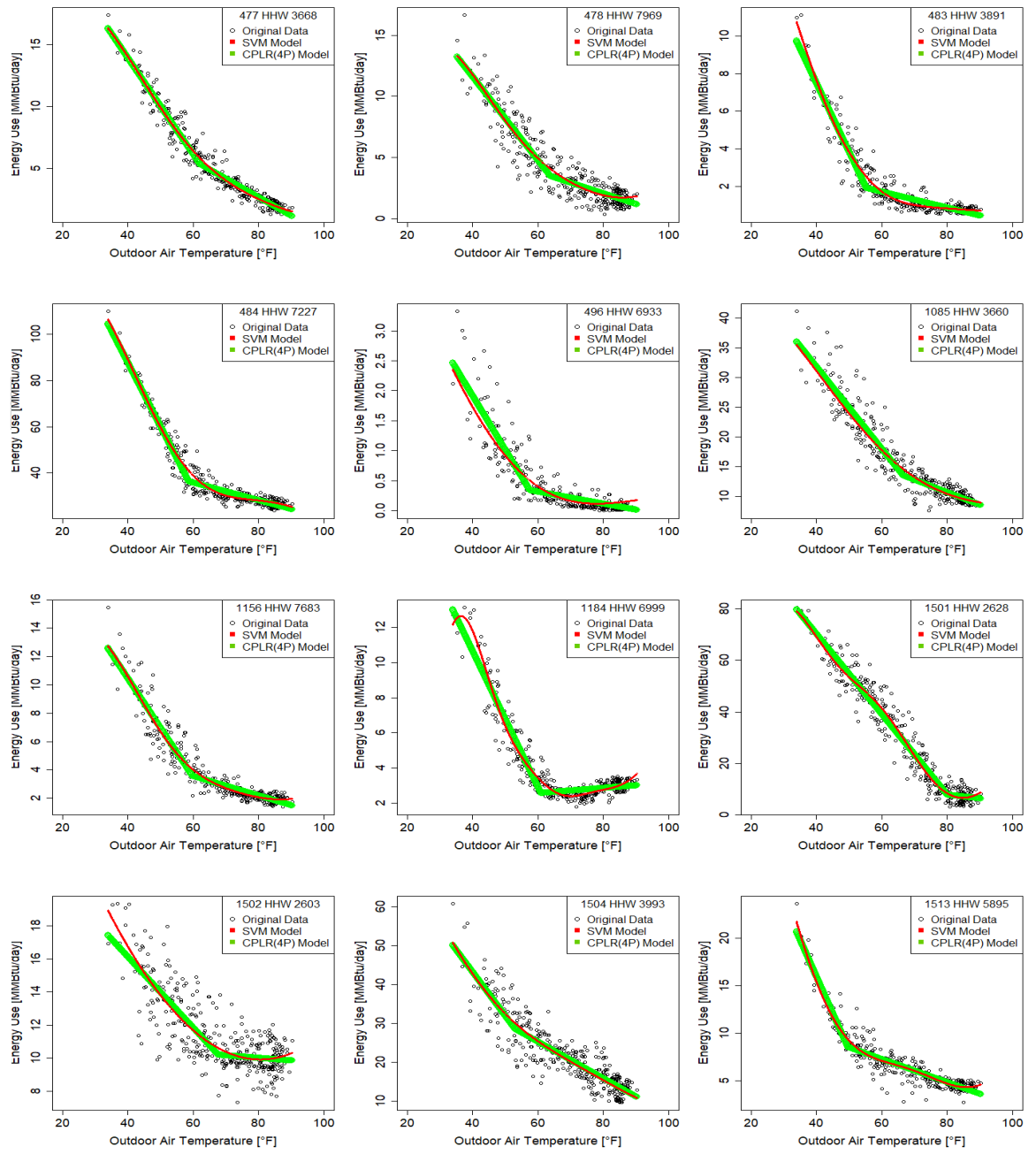Figure D 2 - SVM and CPLR (4P) Regression Plots (CHW-OAT) Sample No. 13 – No. 24 (Daily)
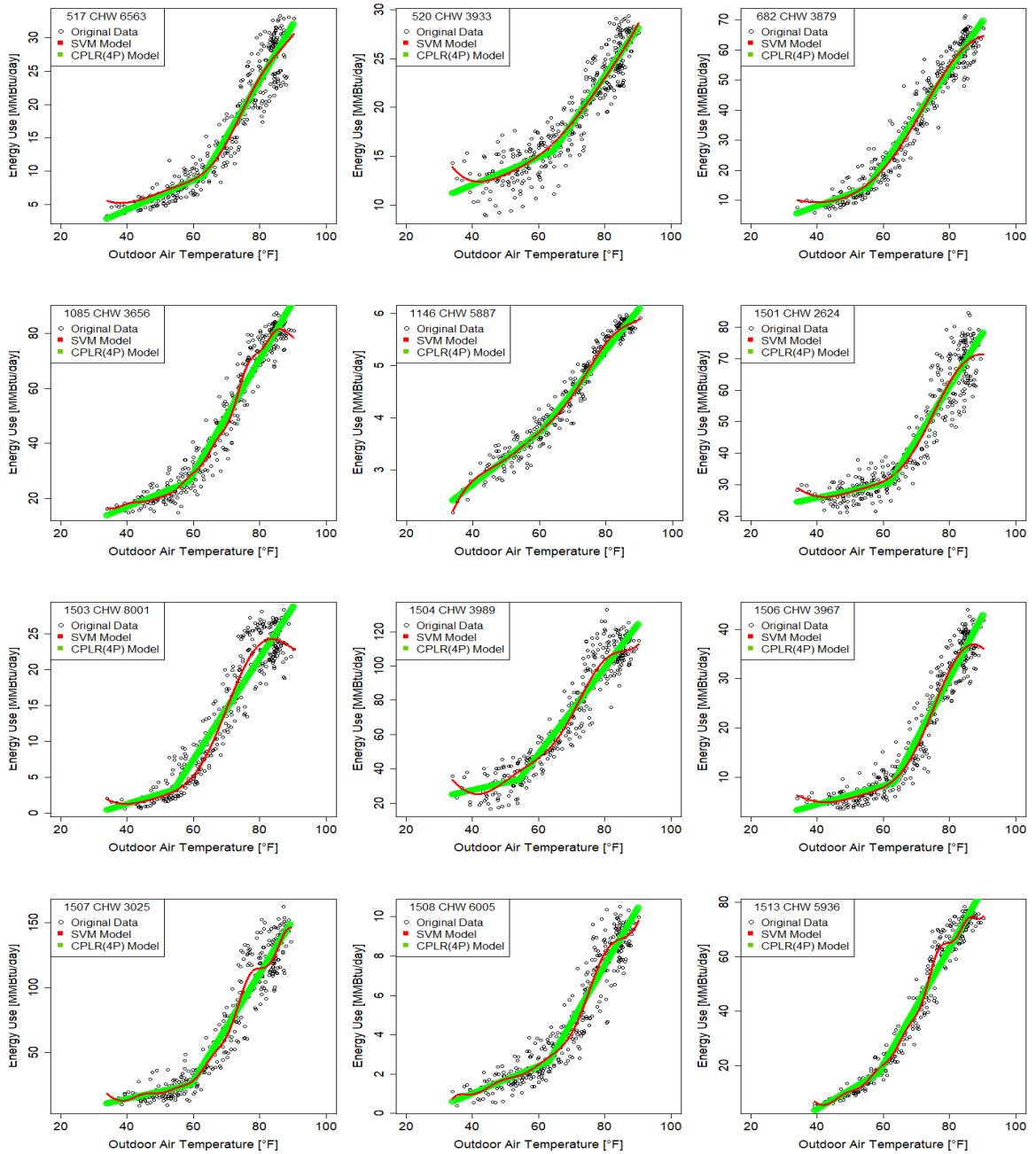
Figure D 3 - SVM and CPLR (4P) Regression Plots (CHW-OAT) Sample No. 25 – No. 36 (Daily)
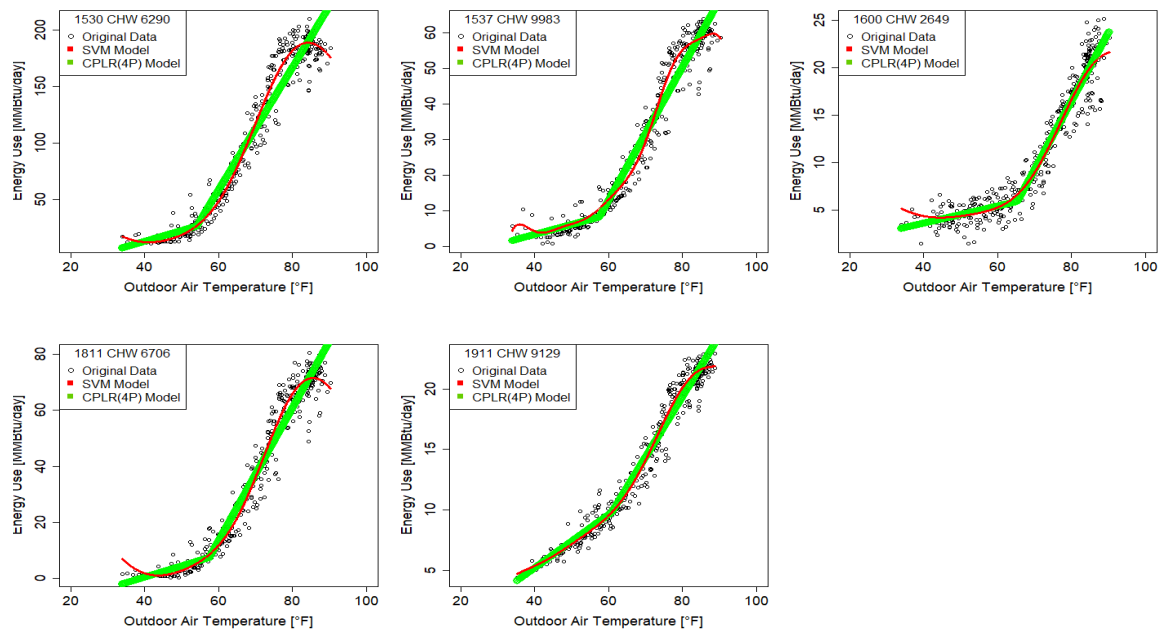
Figure D 4 - SVM and CPLR (4P) Regression Plots (CHW-OAT) Sample No. 37 – No. 41 (Daily)

D.1.2 CHW - OAE
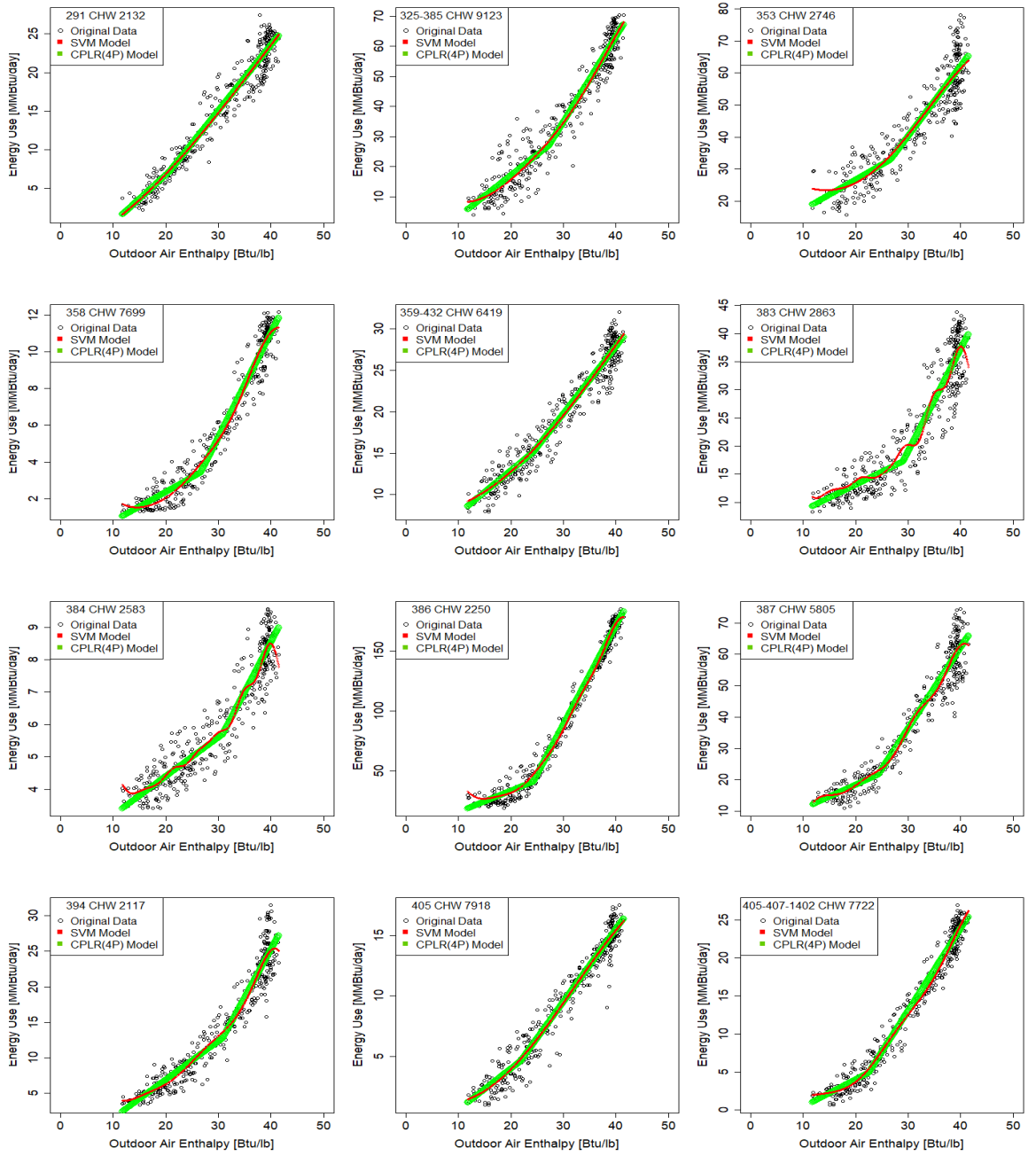
Figure D 5 - SVM and CPLR (4P) Regression Plots (CHW-OAE) Sample No. 1 – No. 12 (Daily)
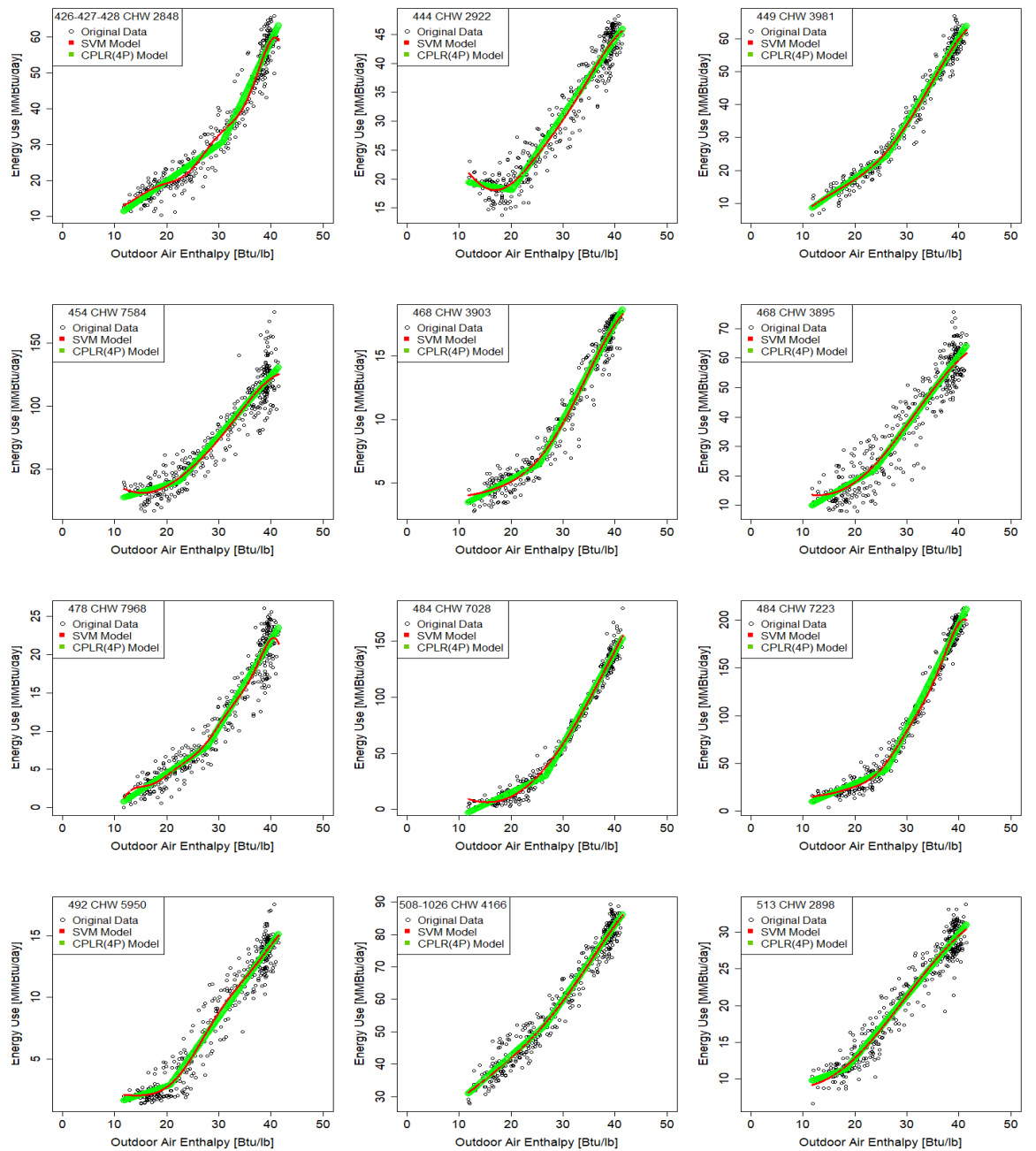
Figure D 6 - SVM and CPLR (4P) Regression Plots (CHW-OAE) Sample No. 13 – No. 24 (Daily)
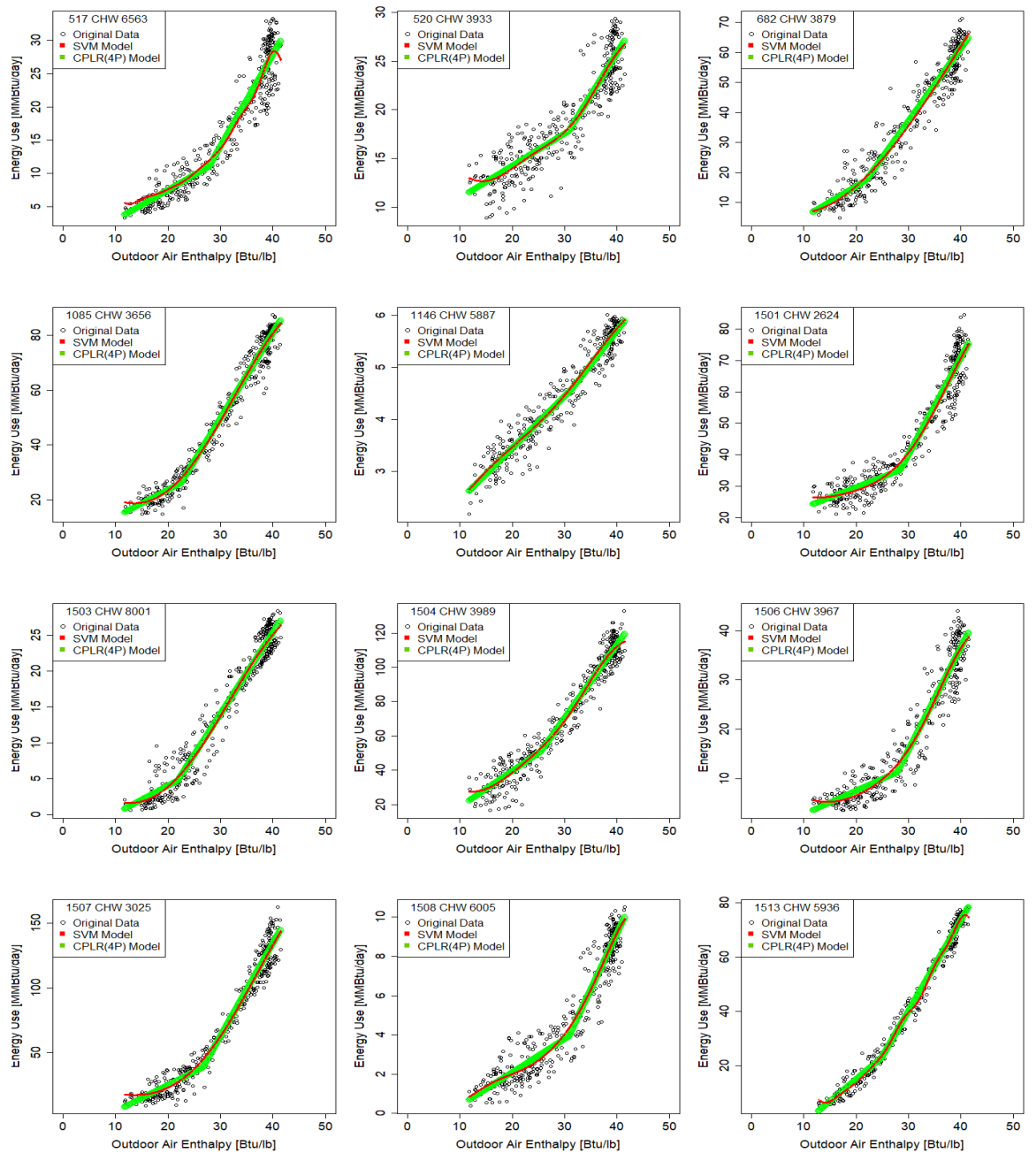
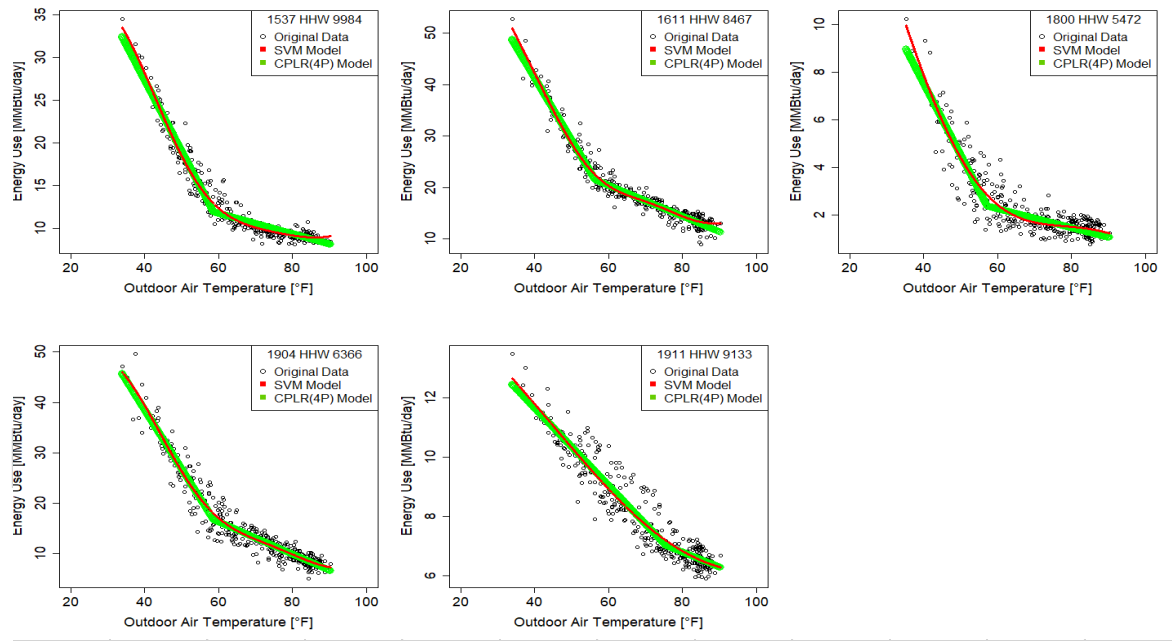Figure D 7 - SVM and CPLR (4P) Regression Plots (CHW-OAE) Sample No. 25 – No. 36 (Daily)
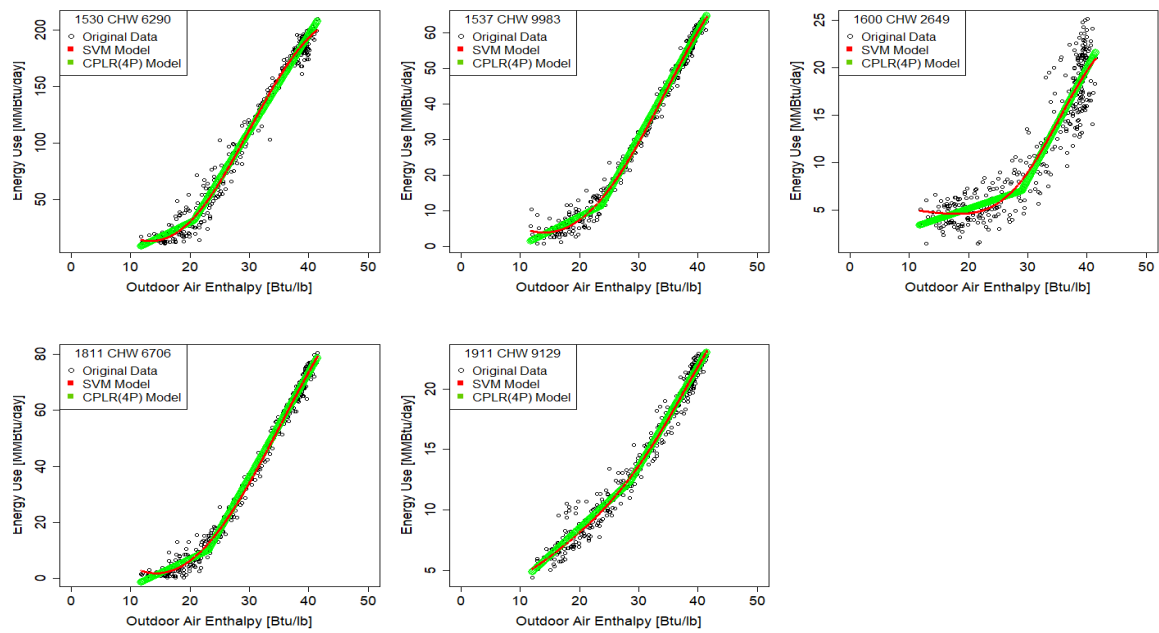
Figure D 8 - SVM and CPLR (4P) Regression Plots (CHW-OAE) Sample No. 37 – No. 41 (Daily)

D.1.3 CHW - OEE
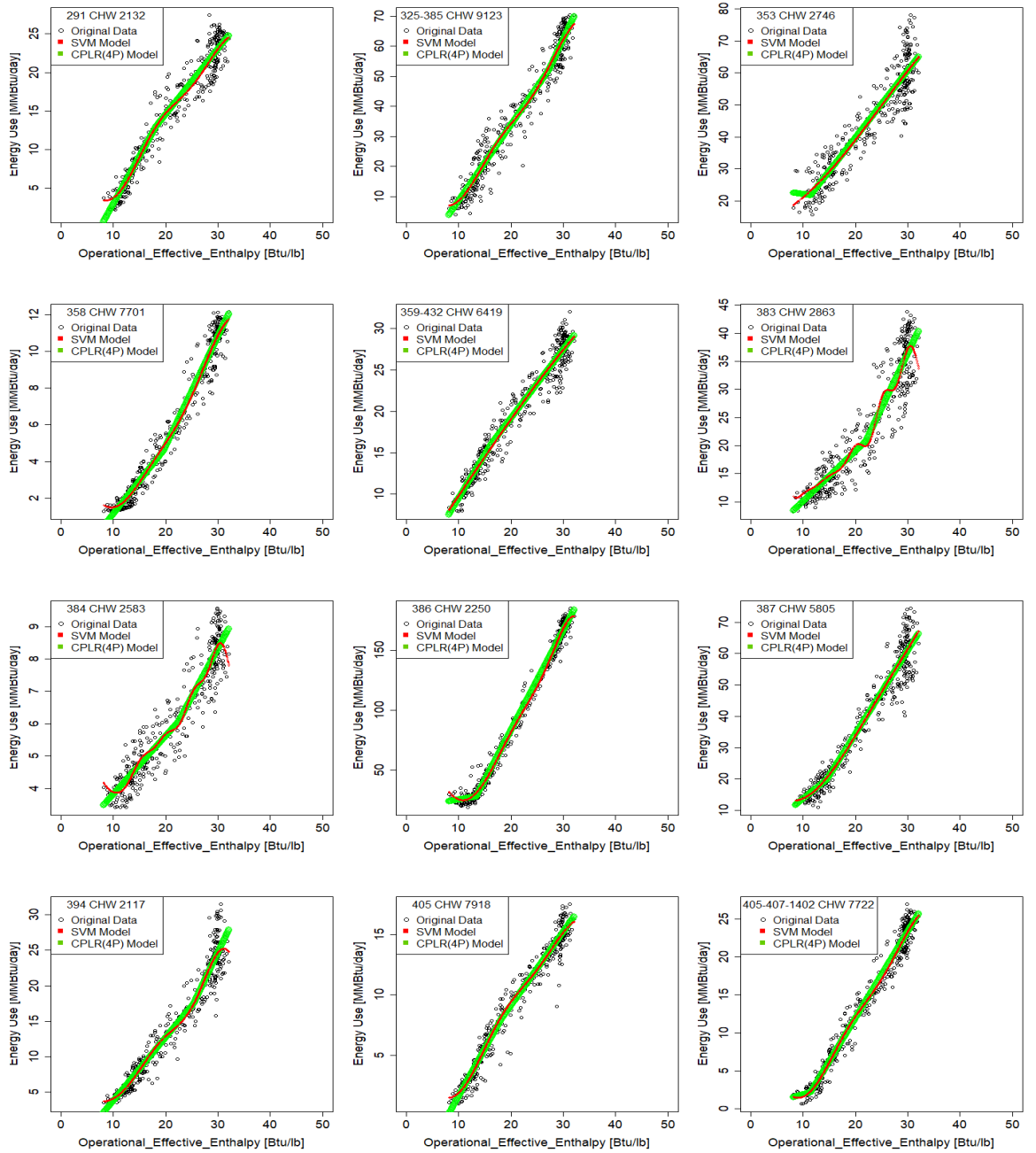
Figure D 9 - SVM and CPLR (4P) Regression Plots (CHW-OEE) Sample No. 1 – No. 12 (Daily)
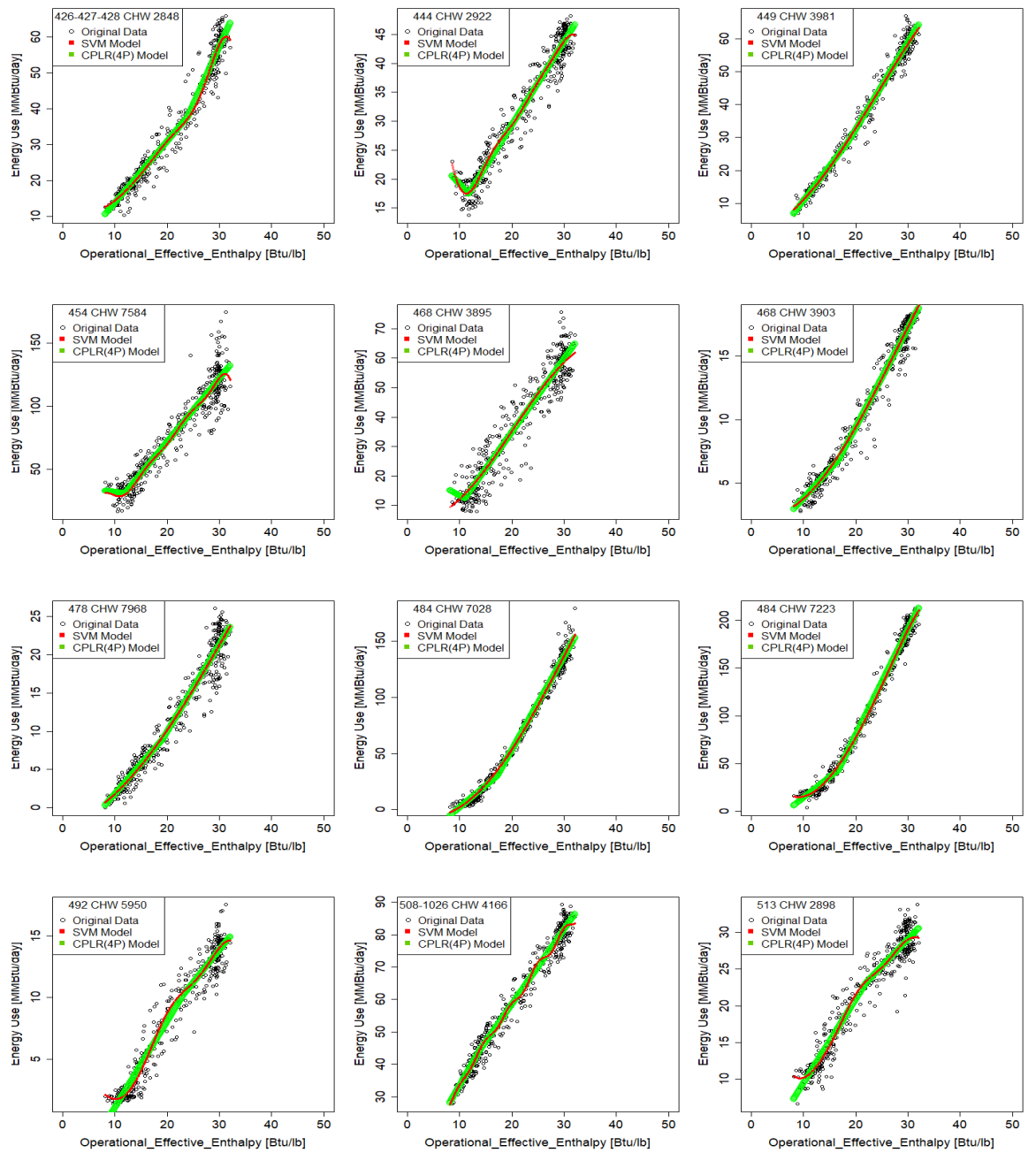
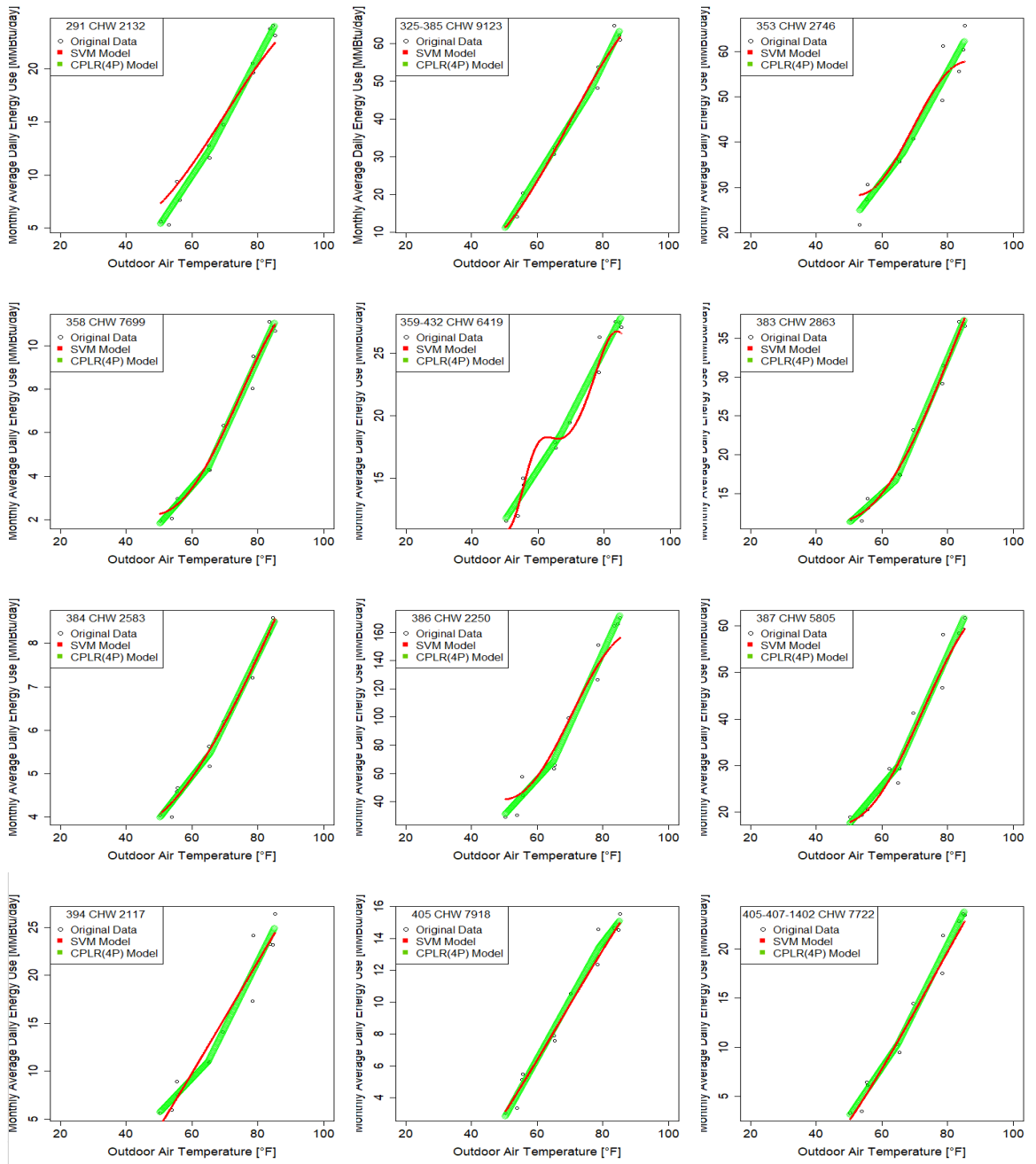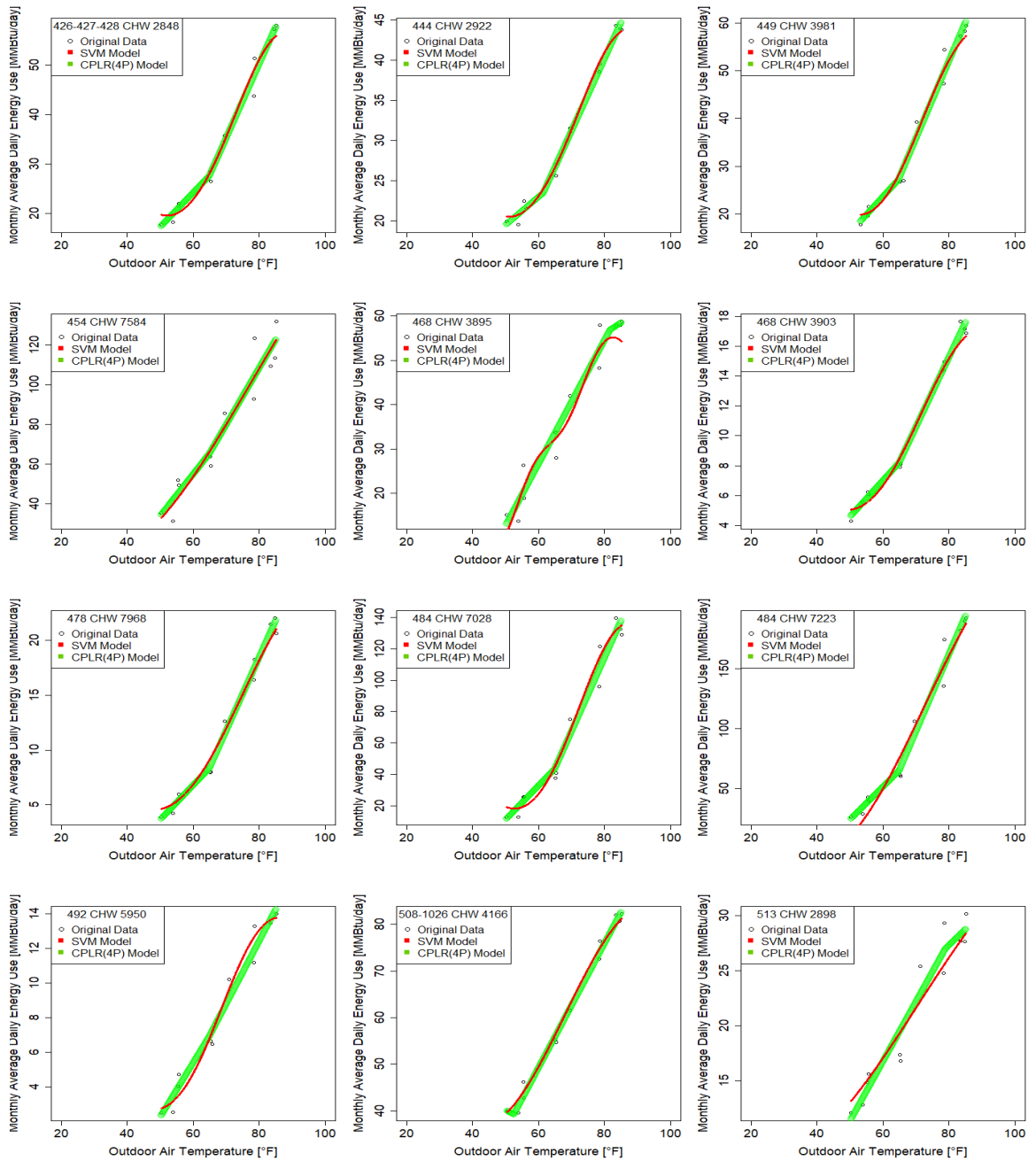Figure D 10 - SVM and CPLR (4P) Regression Plots (CHW-OEE) Sample No. 13 – No. 24 (Daily)
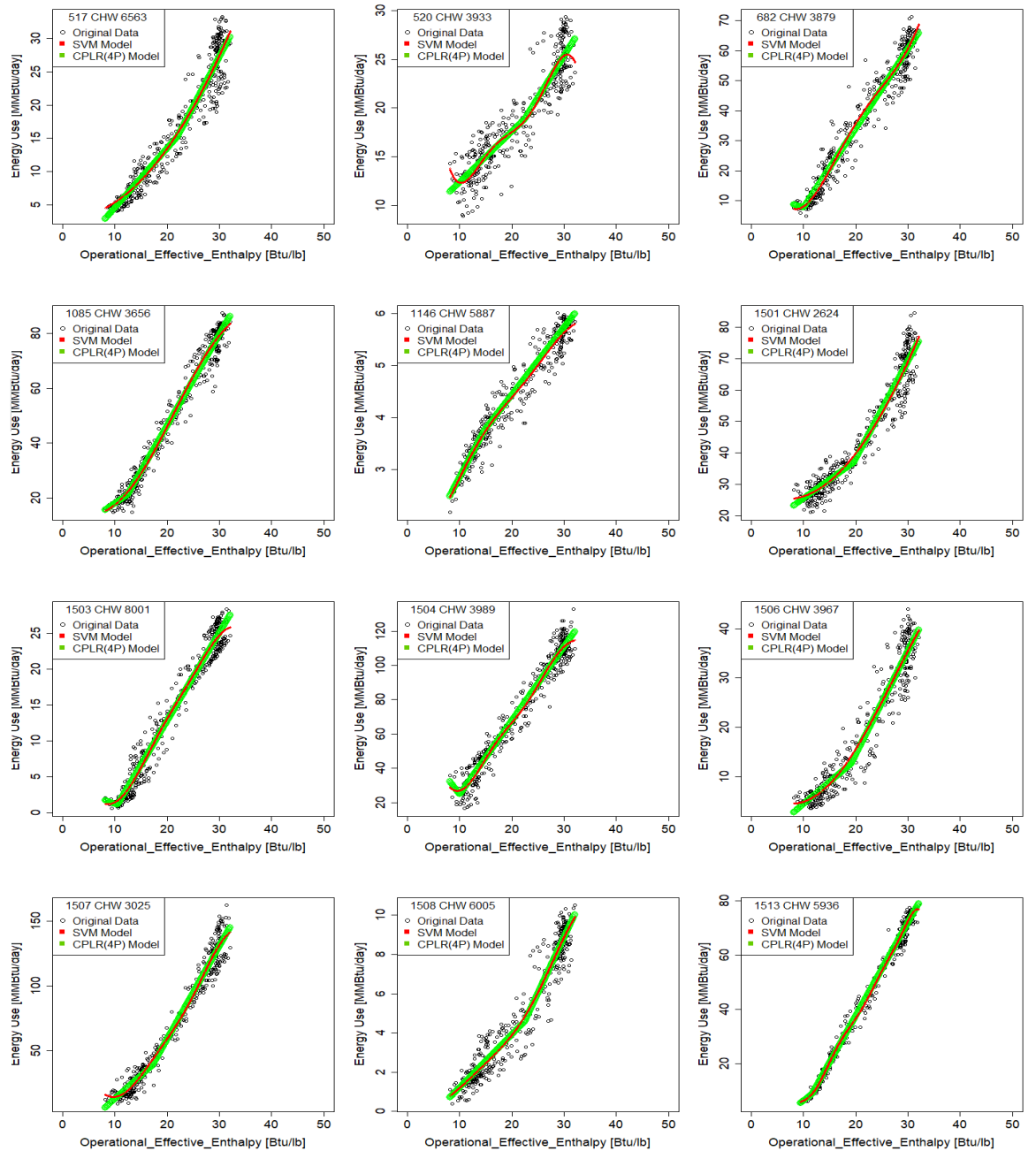
Figure D 11 - SVM and CPLR (4P) Regression Plots (CHW-OEE) Sample No. 25 – No. 36 (Daily)
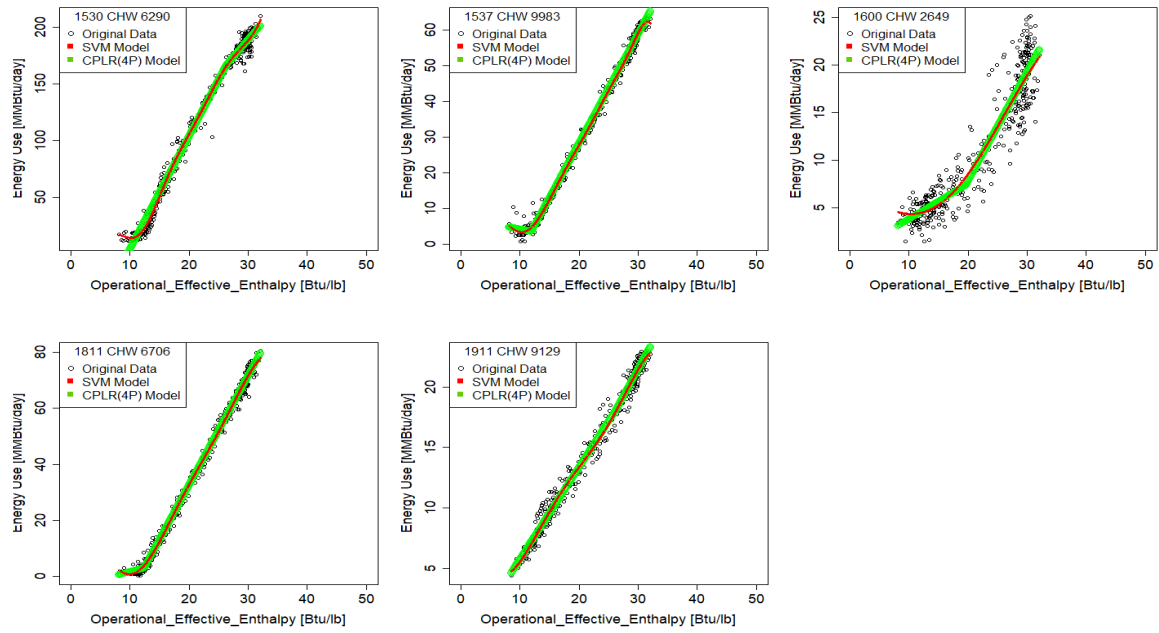
Figure D 12 - SVM and CPLR (4P) Regression Plots (CHW-OEE) Sample No. 37 – No. 41 (Daily)

D.1.4 HHW - OAT

Figure D 13 - SVM and CPLR (4P) Regression Plots (HHW-OAT) Sample No. 1 – No. 12 (Daily)

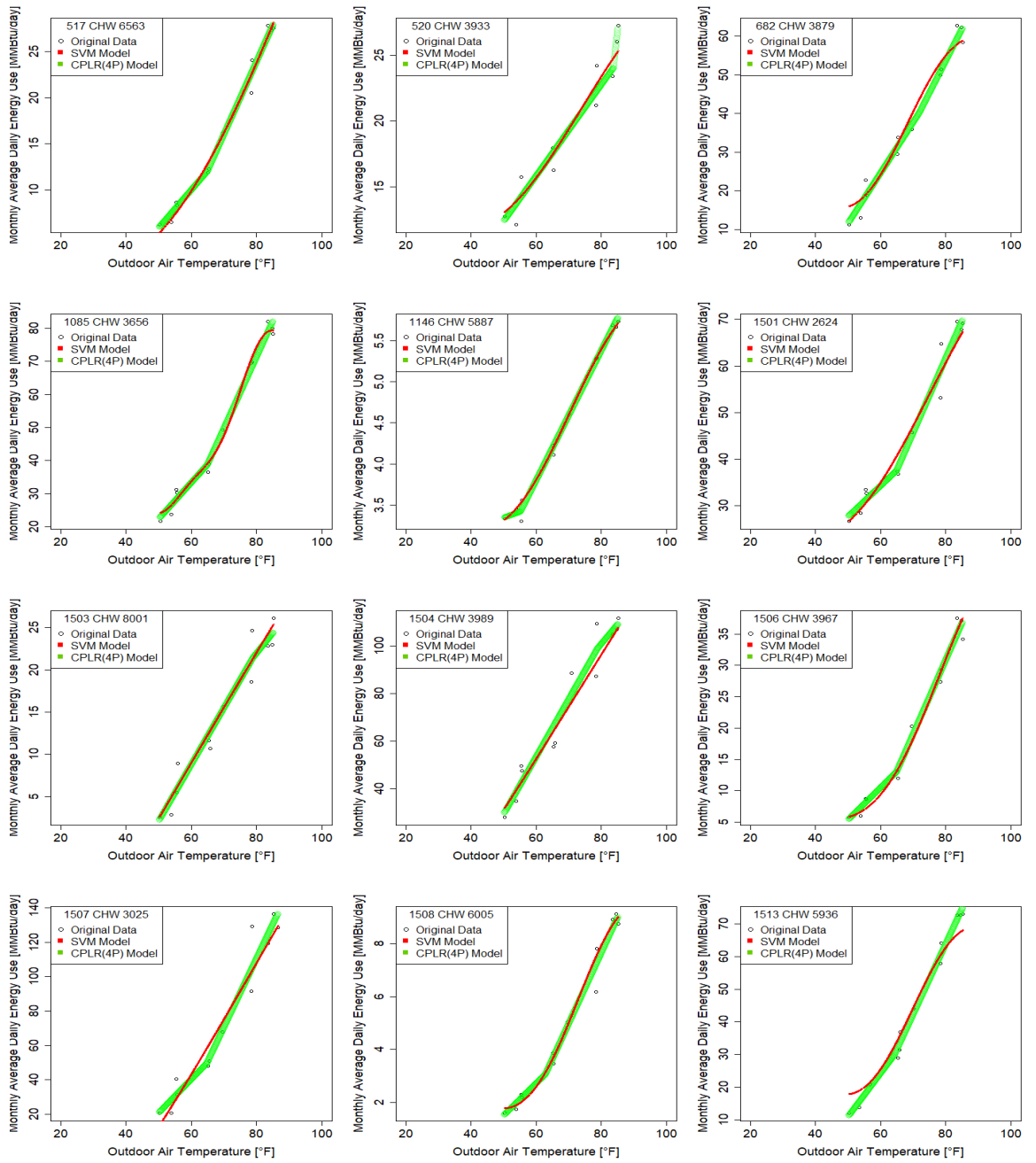Figure D 14 - SVM and CPLR (4P) Regression Plots (HHW-OAT) Sample No. 13 – No. 24 (Daily)

Figure D 15 - SVM and CPLR (4P) Regression Plots (HHW-OAT) Sample No. 25 – No. 36 (Daily)

Figure D 16 - SVM and CPLR (4P) Regression Plots (HHW-OAT) Sample No. 37 – No. 41 (Daily)

D.2 Monthly

D.2.1 CHW - OAT

Figure D 17 - SVM and CPLR (4P) Regression Plots (CHW-OAT) Sample No. 37 – No. 41 (Monthly)

Figure D 18 - SVM and CPLR (4P) Regression Plots (CHW-OAT) Sample No. 13 – No. 24 (Monthly)

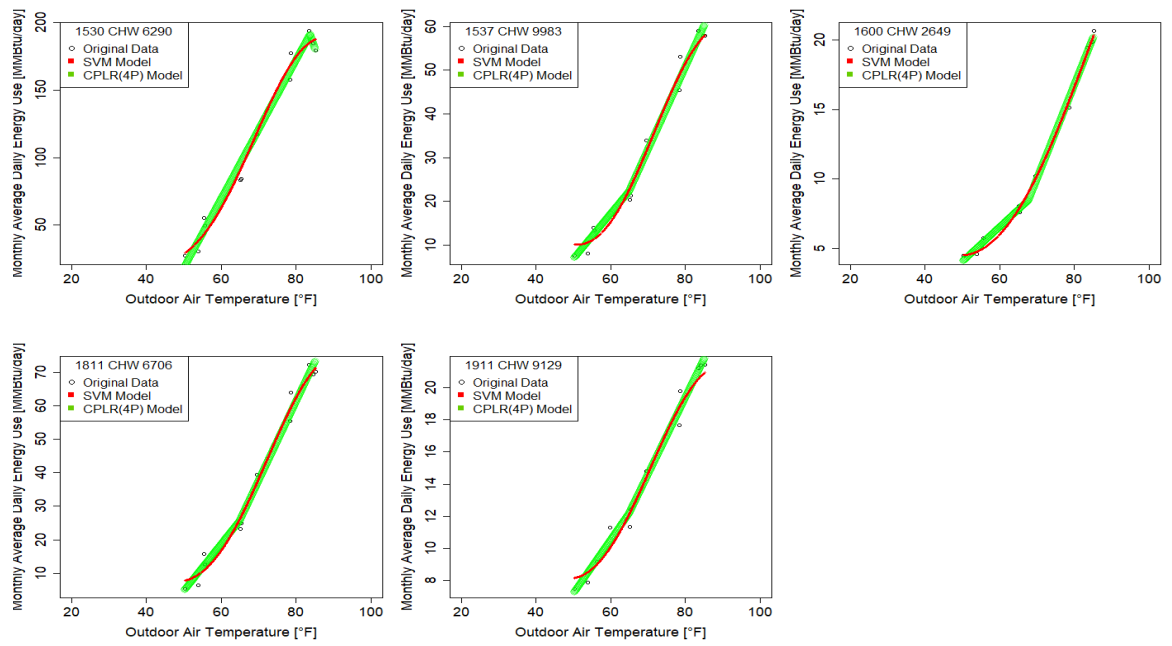Figure D 19 - SVM and CPLR (4P) Regression Plots (CHW-OAT) Sample No. 25– No. 36 (Monthly)

Figure D 20 - SVM and CPLR (4P) Regression Plots (CHW-OAT) Sample No. 37– No. 41 (Monthly)
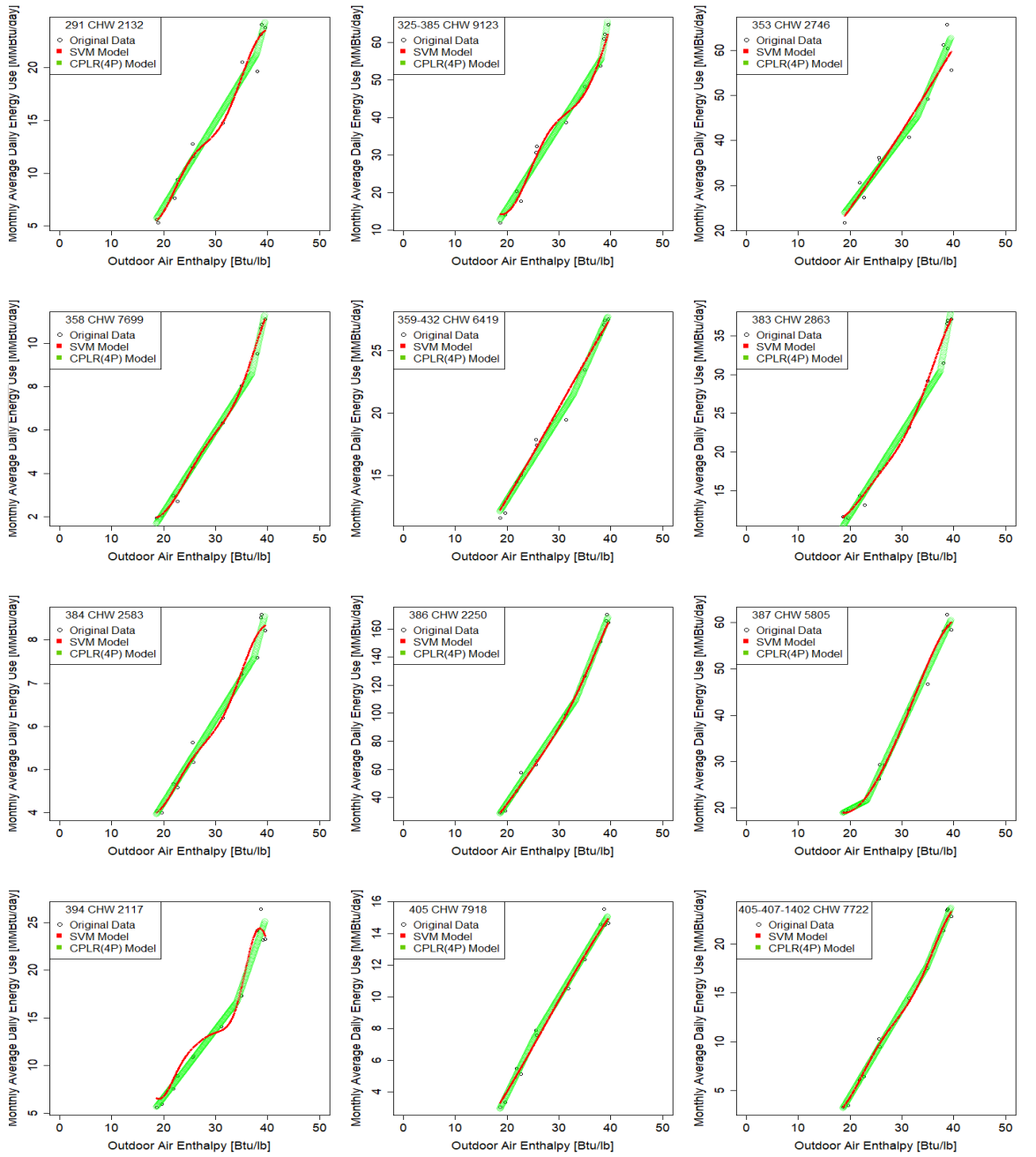
D.2.2 CHW - OAE

Figure D 21 - SVM and CPLR (4P) Regression Plots (CHW-OAE) Sample No. 1– No. 12 (Monthly)
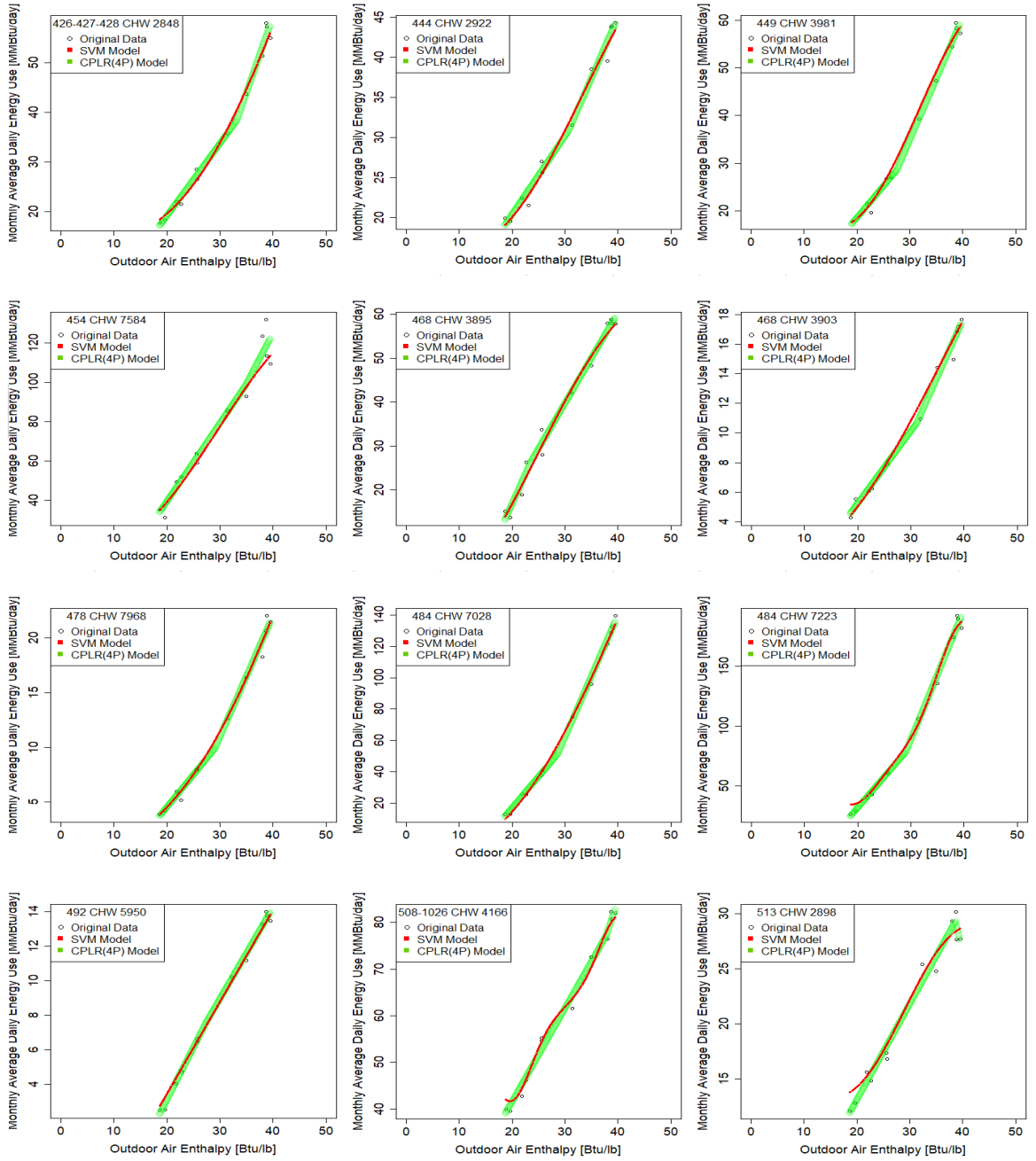
Figure D 22 - SVM and CPLR (4P) Regression Plots (CHW-OAE) Sample No. 13– No. 24 (Monthly)
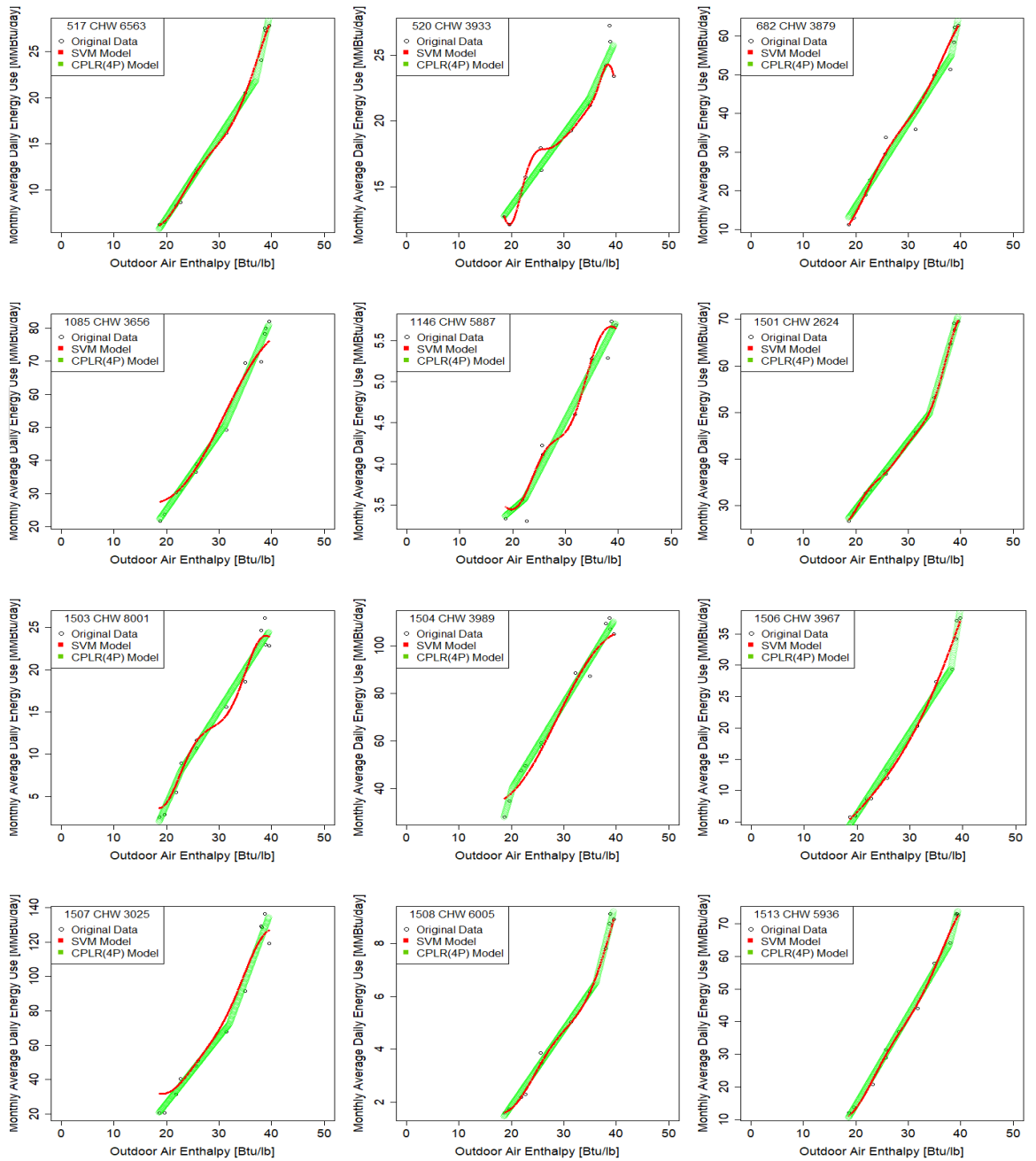
Figure D 23 - SVM and CPLR (4P) Regression Plots (CHW-OAE) Sample No. 25– No. 36 (Monthly)
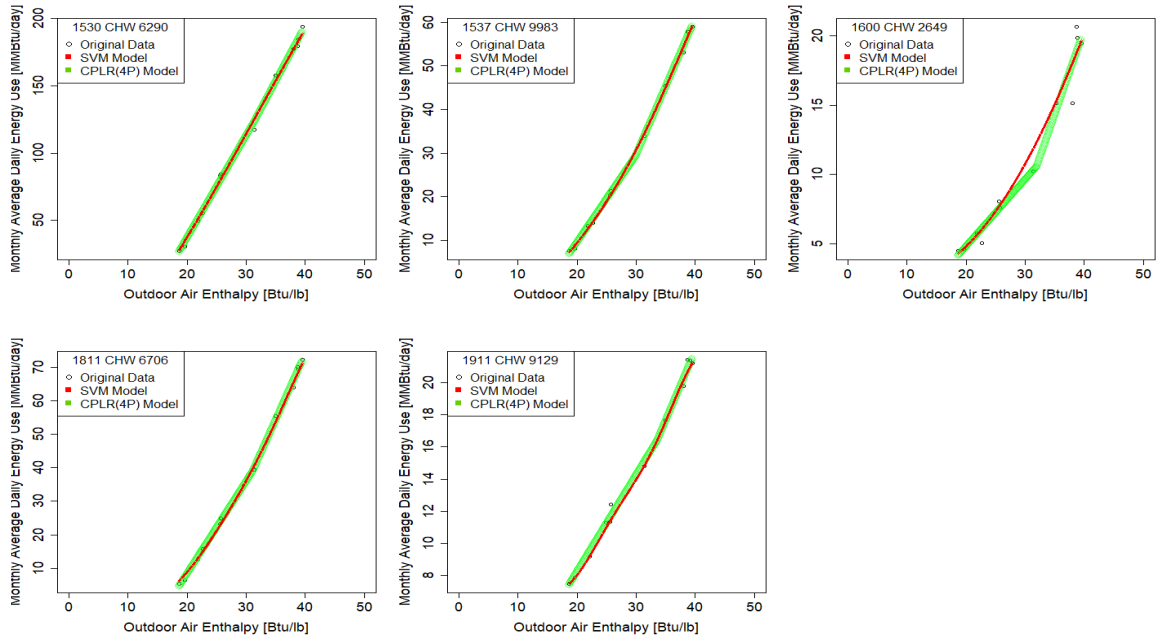
Figure D 24 - SVM and CPLR (4P) Regression Plots (CHW-OAE) Sample No. 37– No. 41 (Monthly)

D.2.3 CHW - OEE

Figure D 25 - SVM and CPLR (4P) Regression Plots (CHW-OEE) Sample No. 1– No. 12 (Monthly)

Figure D 26 - SVM and CPLR (4P) Regression Plots (CHW-OEE) Sample No. 13– No. 24 (Monthly)
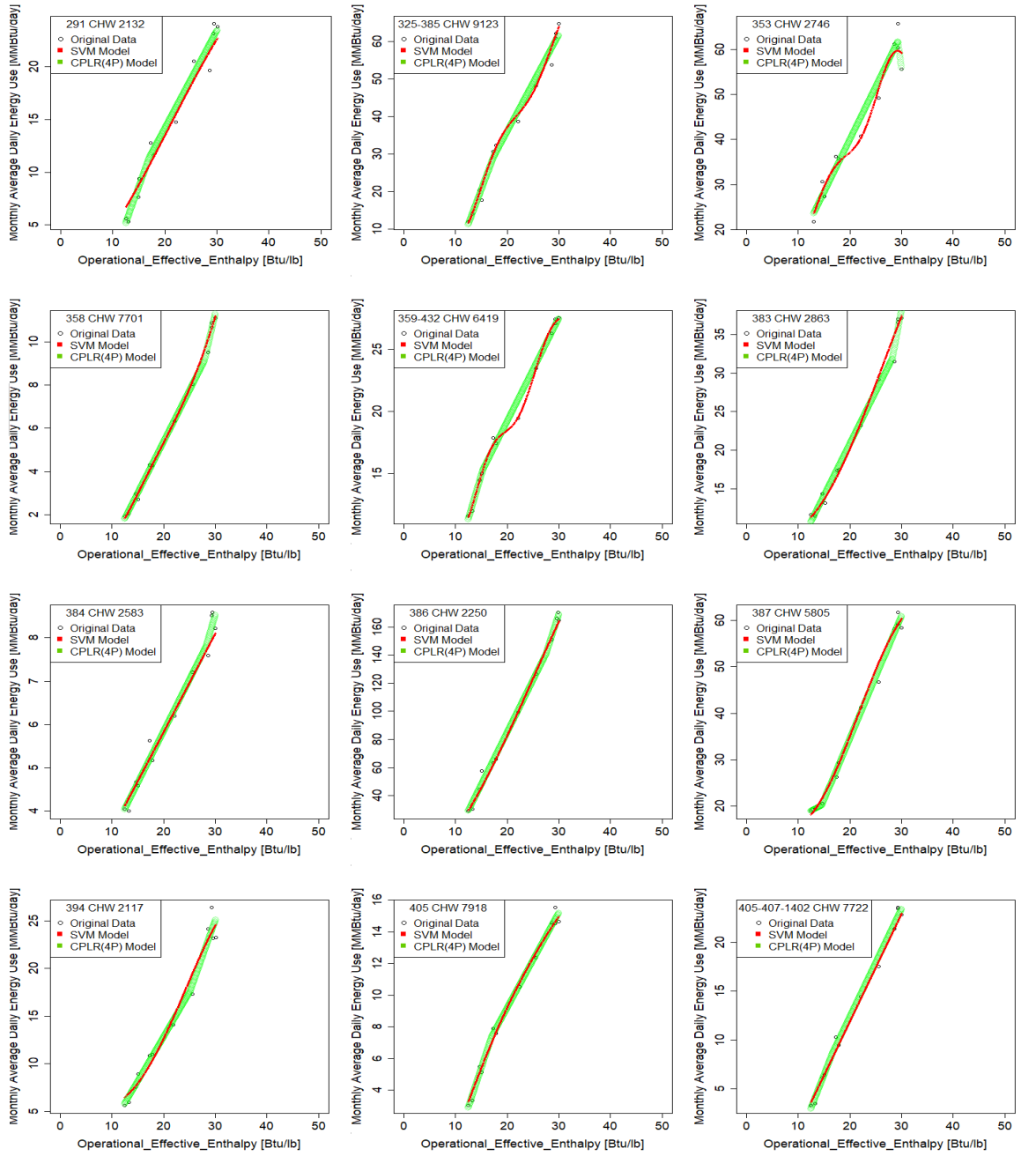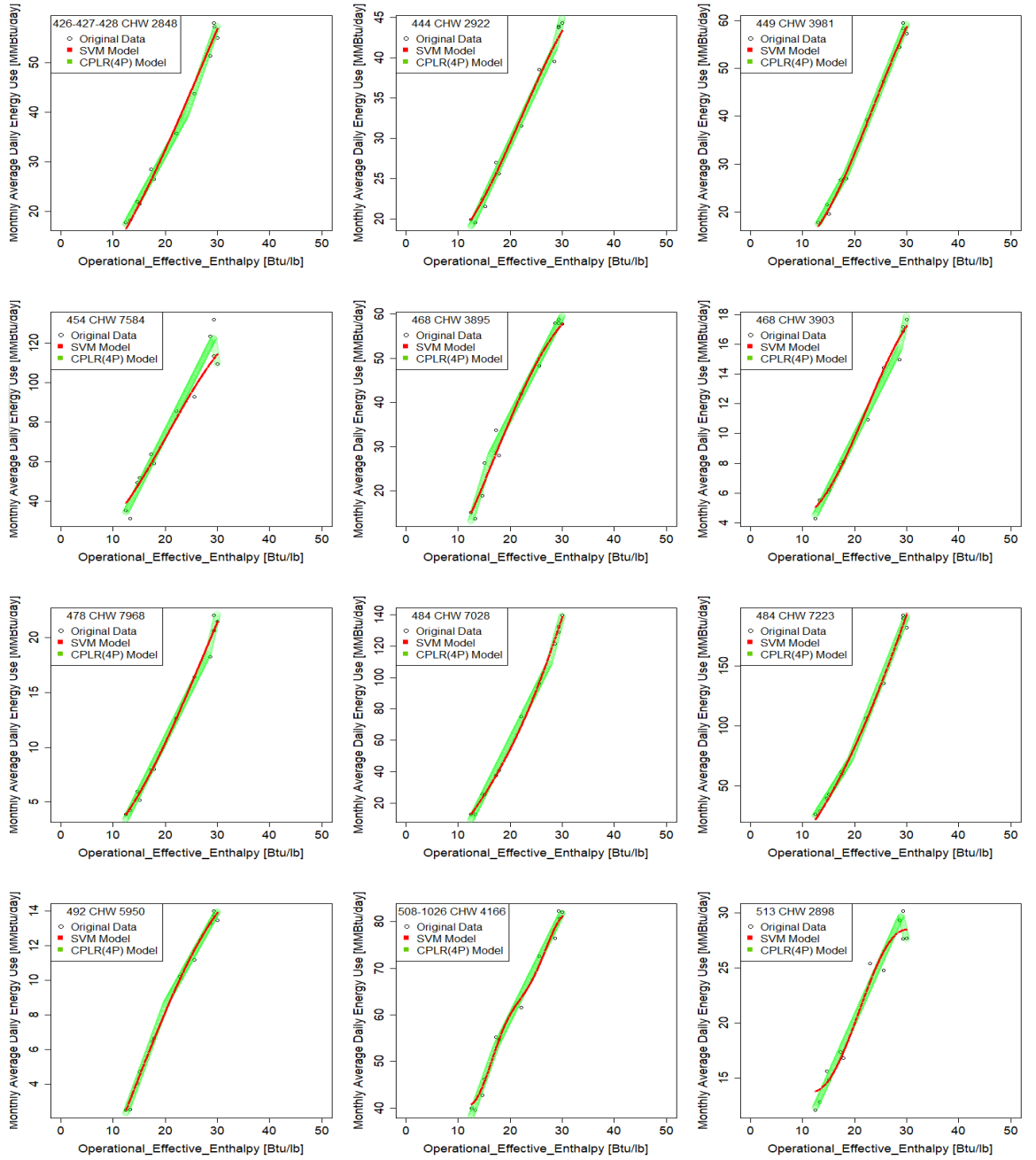
Figure D 27 - SVM and CPLR (4P) Regression Plots (CHW-OEE) Sample No. 25– No. 36 (Monthly)

Figure D 28 - SVM and CPLR (4P) Regression Plots (CHW-OEE) Sample No. 37– No. 41 (Monthly)

D.2.4 HHW - OAT

Figure D 29 - SVM and CPLR (4P) Regression Plots (HHW-OAT) Sample No. 1– No. 12 (Monthly)

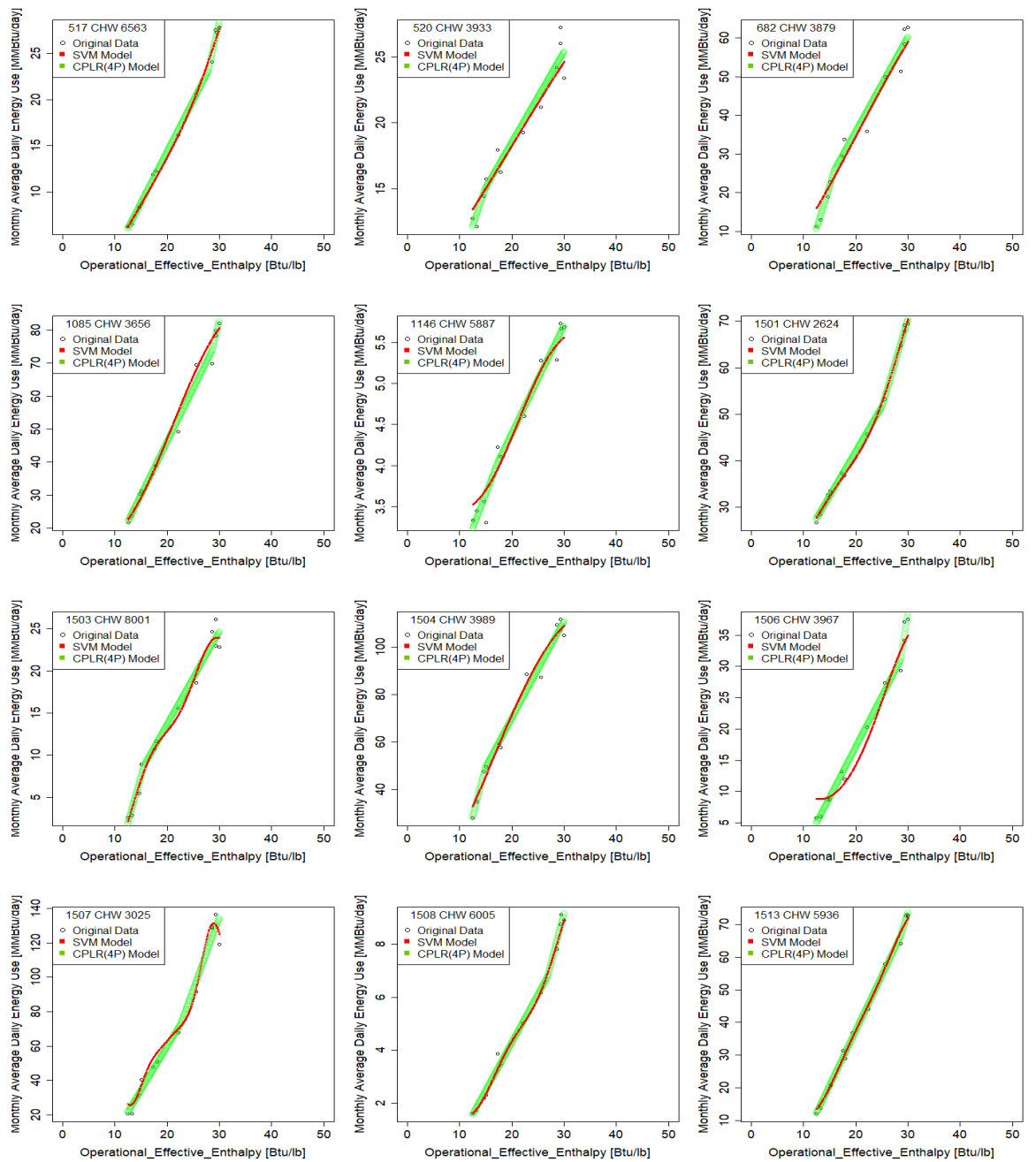Figure D 30 - SVM and CPLR (4P) Regression Plots (HHW-OAT) Sample No. 13 – No. 24 (Monthly)
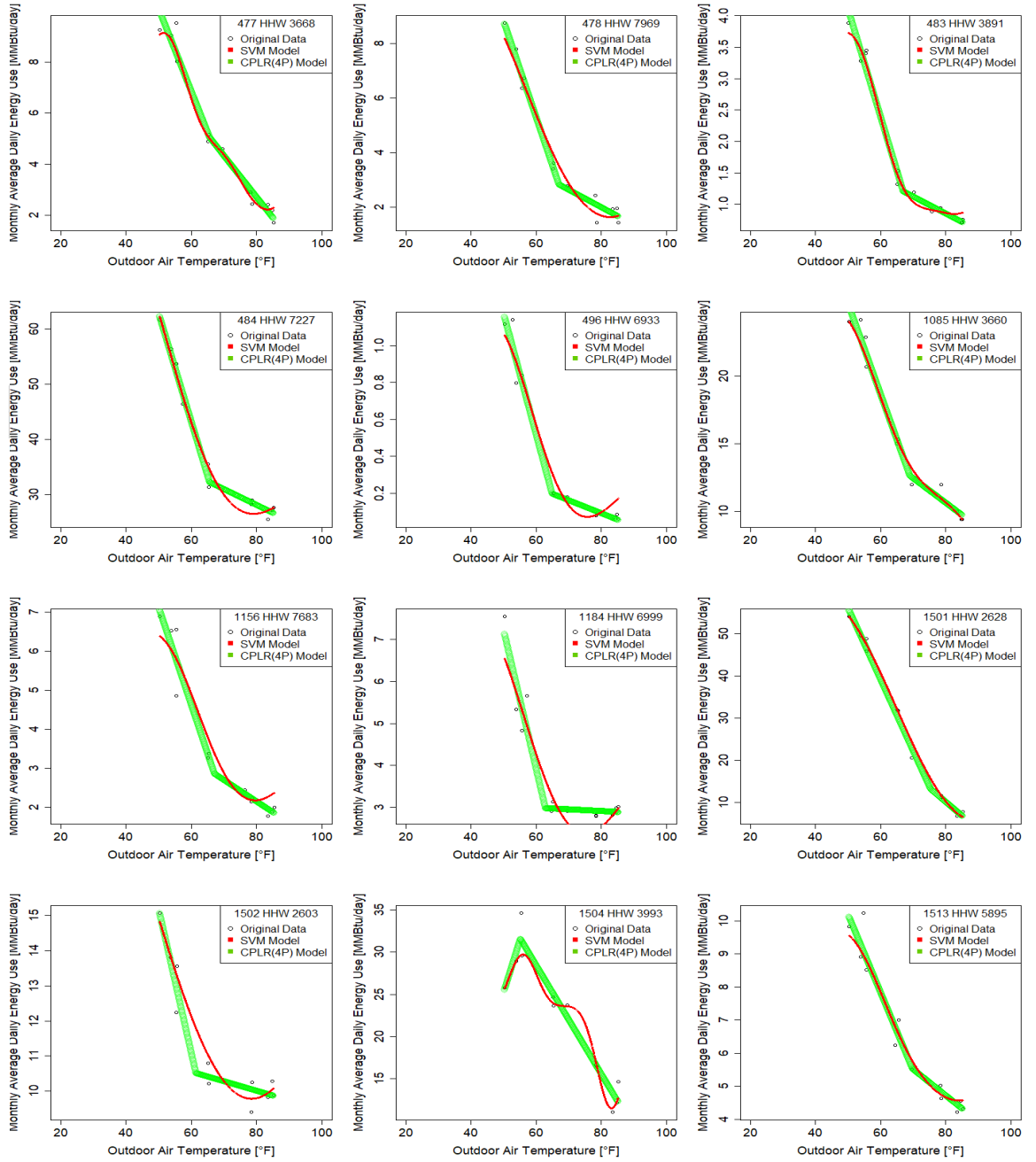
Figure D 31 - SVM and CPLR (4P) Regression Plots (HHW-OAT) Sample No. 25 – No. 36 (Monthly)
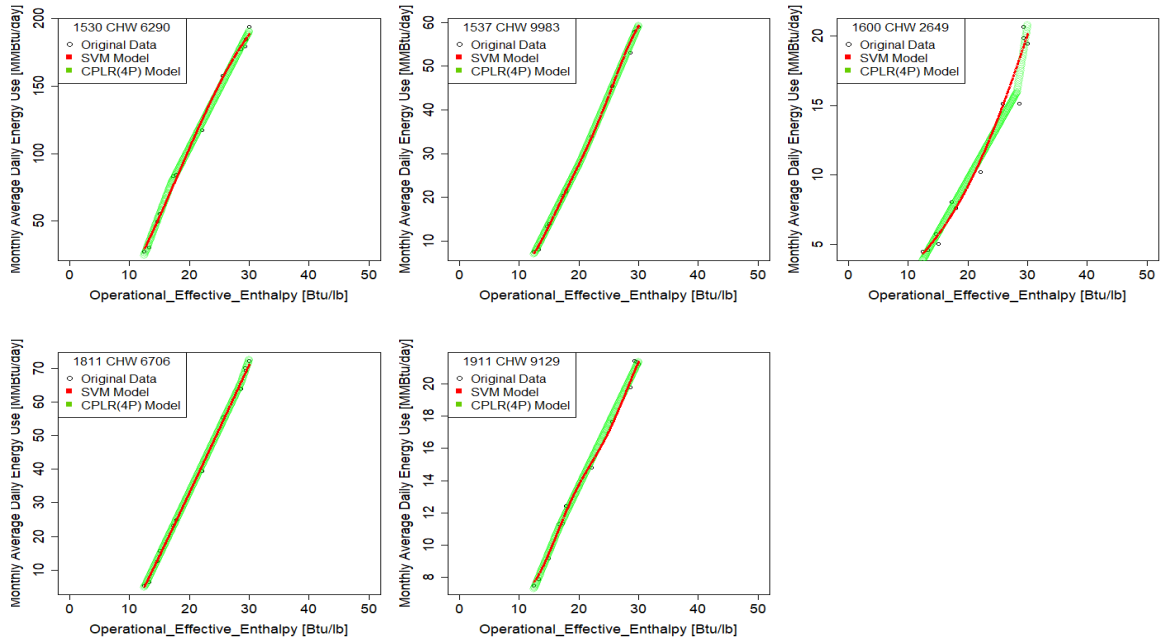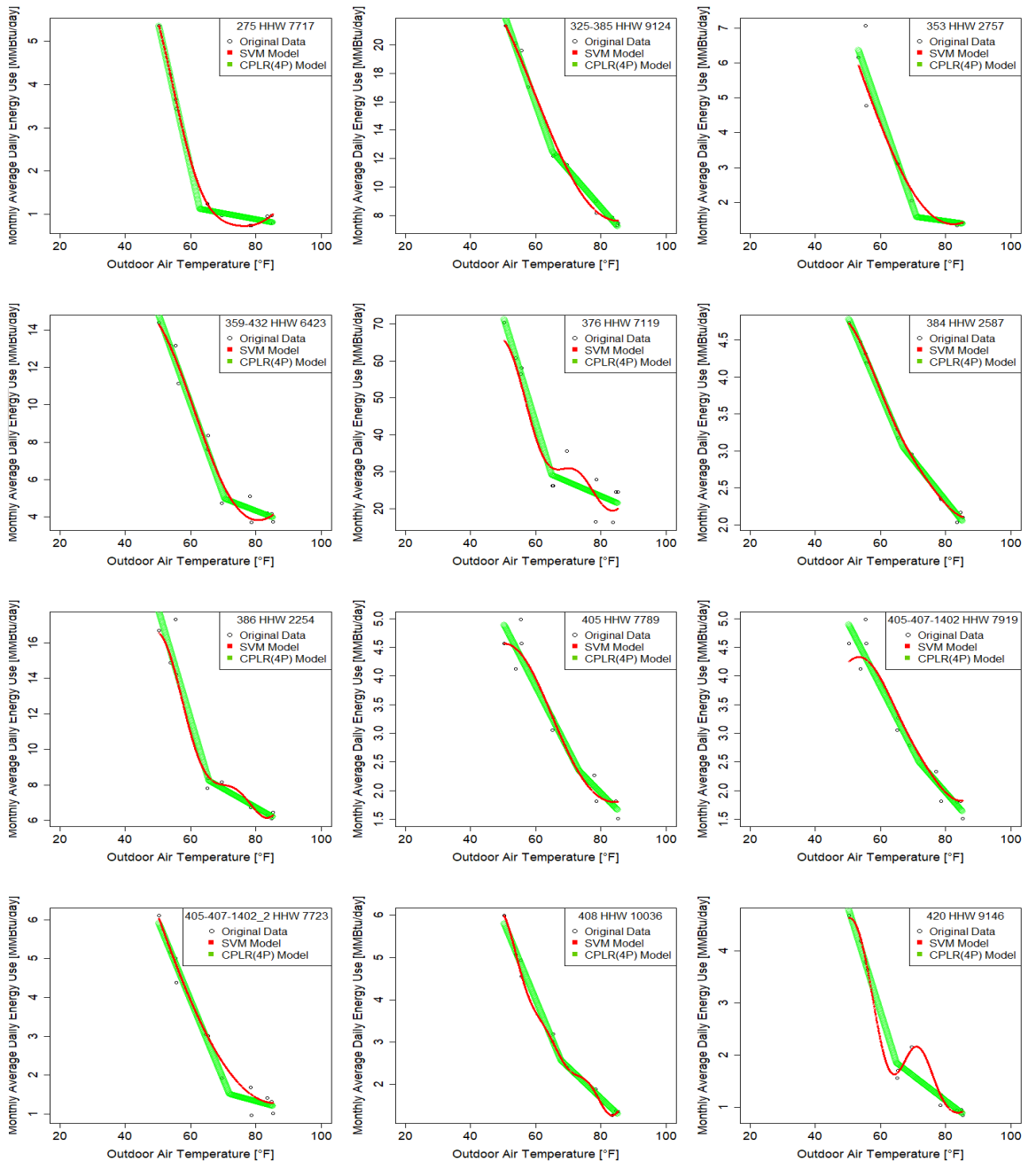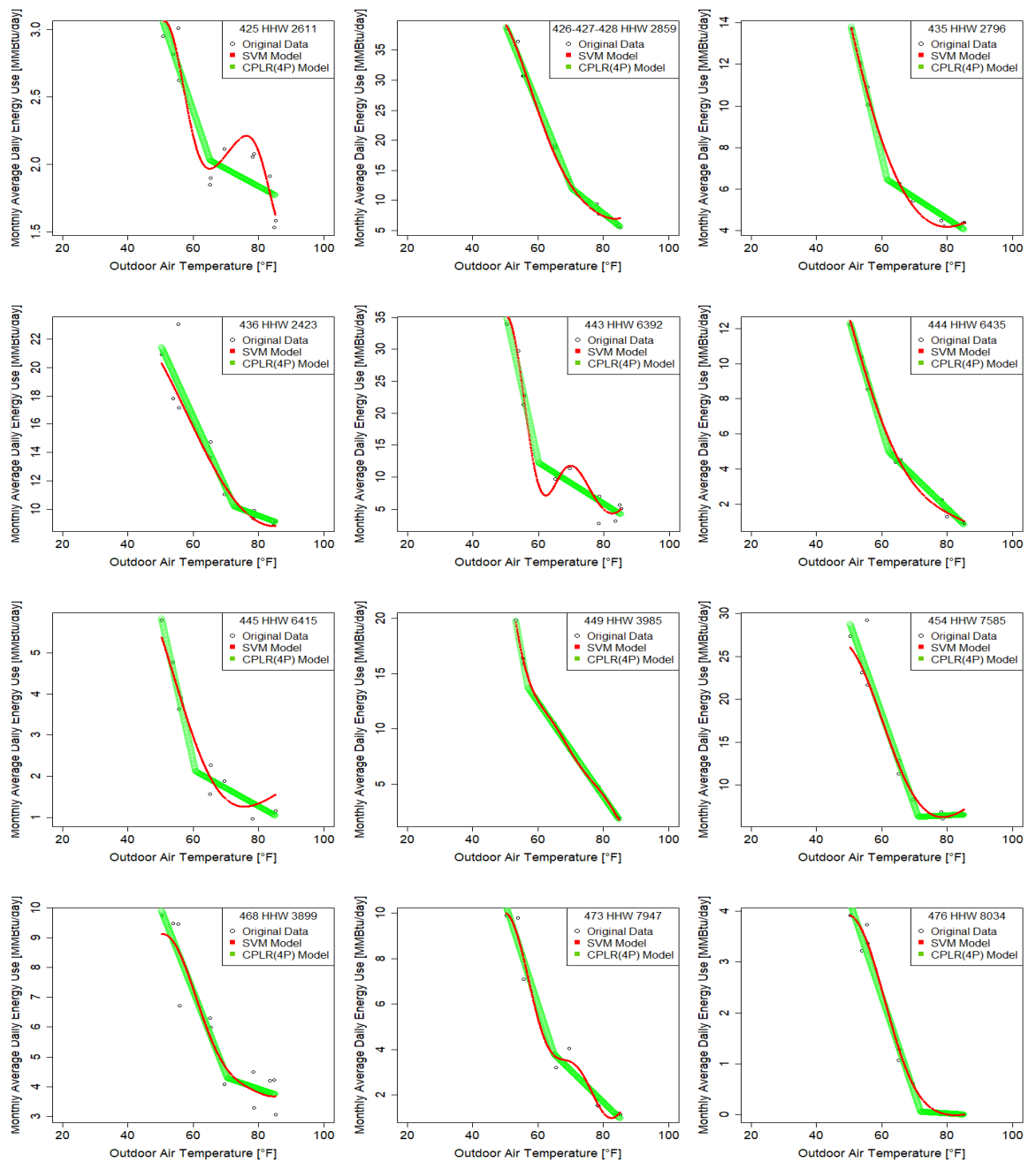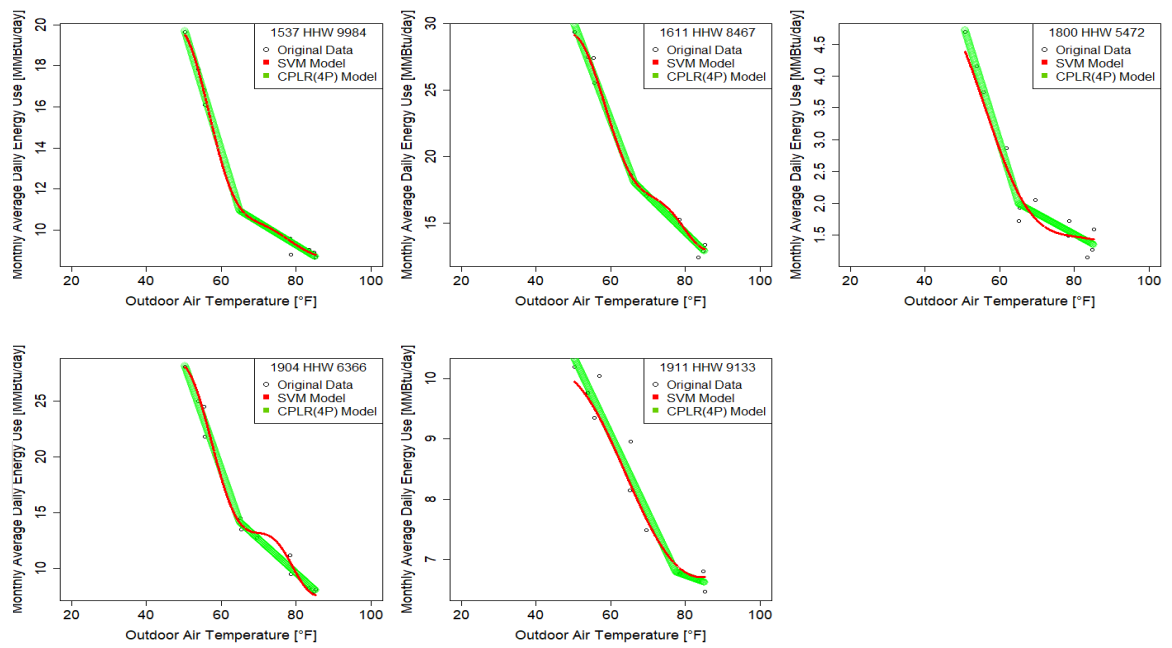
Figure D 32 - SVM and CPLR (4P) Regression Plots (HHW-OAT) Sample No. 37 – No. 41 (Monthly)

APPENDIX E. EXAMPLE OF ENERGY SAVINGS CALCULATION USING SVM REGRESSION

One of the main applications of building energy baseline model is to calculate savings between different periods. According to International Performance Measurement & Verification Protocol (IPMVP)[63], there are four options to determine savings as below:

1) Option A: Partially Measured Retrofit Isolation

2) Option B: Retrofit Isolation

3) Option C: Whole Building

4) Option D: Calibrated Simulation

What is dealt with about SVM regression model in this thesis is in line with the Option C. Thus, calculation process below is laid out according to the option C.

Sample Data

Wisenbaker Engineering Research Center, Texas A&M University campus, College Station, Texas.

Baseline Period: Jan 1, 2014 – Dec 31, 2014

Post Period: Jan 1, 2017 – Dec 31, 2017

Energy Type: Heating Hot Water

SVM Input

SVM input attributes

 - dependent variable: Daily HHW Use in 2014 [MMBtu/day]

 - independent variable: Daily Average Outdoor Air Temperature in 2014 [°F]

SVM Output (Baseline Model Equation)

RBF kernel SVM regression takes the following form.

$$f(x) = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)e^{-\gamma\|x-x_i\|^2} + b \qquad\qquad \text{(EQ.F 1)}$$

where b = - 0.9237, γ = 0.25, N = 101 (Number of Support Vector).

In this example, the EQ. F1 is established with normalized data as below.

$$x_{scaled} = \frac{x_{orginal} - x_{original,mean}}{\sigma_{x,original}} \qquad\qquad \text{(EQ.F 2)}$$

where $x_{original,mean}$ = 67.4, $\sigma_{original,mean}$ = 14.5.

Thus, scaled data should be inserted instead of the original one. Likewise, the f(x) of the Eq. F2 is scaled value. Thus, it should be inverse-scaled so that it has physical meaning.

$$f(x) = f(x)_{scaled} \times \sigma_{y,original} + y_{original,mean} \qquad (EQ.F\,3)$$

where $y_{original,mean} = 23.1$, $y_{original,mean} = 11.9$

The values of coefficient ($\alpha_i - \alpha_i^*$) and support vectors (x_i) are listed in the Table F1.

Table F 1 - Independent Variables (OAT) Corresponding to Support Vectors and Coefficients

i	x_i	$\alpha_i - \alpha_i^*$	i	x_i	$\alpha_i - \alpha_i^*$	i	x_i	$\alpha_i - \alpha_i^*$	i	x_i	$\alpha_i - \alpha_i^*$
1	-1.14	-4.00	27	-1.49	4.00	53	0.93	-4.00	79	-0.27	-4.00
2	-1.90	-4.00	28	-1.88	4.00	54	0.97	-4.00	80	-0.68	4.00
3	-1.89	-4.00	29	-0.91	4.00	55	0.94	-4.00	81	-2.07	4.00
4	-0.77	-4.00	30	-1.51	-4.00	56	1.16	-4.00	82	-1.57	4.00
5	-1.51	-4.00	31	0.12	-4.00	57	1.16	-4.00	83	-2.02	4.00
6	-2.78	-4.00	32	-0.88	-4.00	58	0.77	-4.00	84	-0.32	4.00
7	-2.31	-4.00	33	-0.57	-4.00	59	0.82	-0.92	85	-0.77	4.00
8	-1.35	-4.00	34	-0.07	-4.00	60	0.58	-4.00	86	-1.13	-4.00
9	-0.60	-4.00	35	-0.45	-4.00	61	0.54	-4.00	87	-1.43	-4.00
10	-0.31	-4.00	36	-0.41	-4.00	62	1.21	-4.00	88	0.16	4.00
11	-0.72	-4.00	37	-1.76	-4.00	63	0.60	-4.00	89	-0.44	4.00
12	-0.75	-4.00	38	-1.33	4.00	64	0.87	-4.00	90	-0.88	4.00
13	-0.59	-4.00	39	-2.76	2.66	65	0.98	-4.00	91	-1.18	4.00
14	-1.06	-4.00	40	-0.93	4.00	66	0.96	-4.00	92	-0.84	2.89
15	-1.42	-4.00	41	-0.60	4.00	67	1.31	-4.00	93	-1.18	4.00
16	-1.09	-4.00	42	-0.14	4.00	68	0.99	4.00	94	-1.02	4.00
17	-1.20	-4.00	43	-0.84	4.00	69	1.20	4.00	95	-0.43	4.00
18	-1.18	-4.00	44	-0.36	4.00	70	1.24	4.00	96	-1.04	4.00
19	-1.03	-4.00	45	-0.73	-4.00	71	1.11	4.00	97	-1.53	4.00
20	-0.54	-4.00	46	-0.79	4.00	72	1.16	4.00	98	-1.21	4.00
21	-1.18	-4.00	47	0.76	4.00	73	1.26	1.60	99	-0.32	4.00
22	-1.25	-4.00	48	0.81	4.00	74	0.85	4.00	100	-1.03	4.00
23	-2.65	4.00	49	-0.38	4.00	75	0.91	4.00	101	-2.04	4.00
24	-1.75	1.77	50	0.55	4.00	76	0.59	4.00			
25	-0.71	4.00	51	0.72	4.00	77	0.72	4.00			
26	-1.45	4.00	52	0.38	4.00	78	-0.08	-4.00		r	

The reported performance is as below.

CV = 18.16 %, R2 = 0.8758
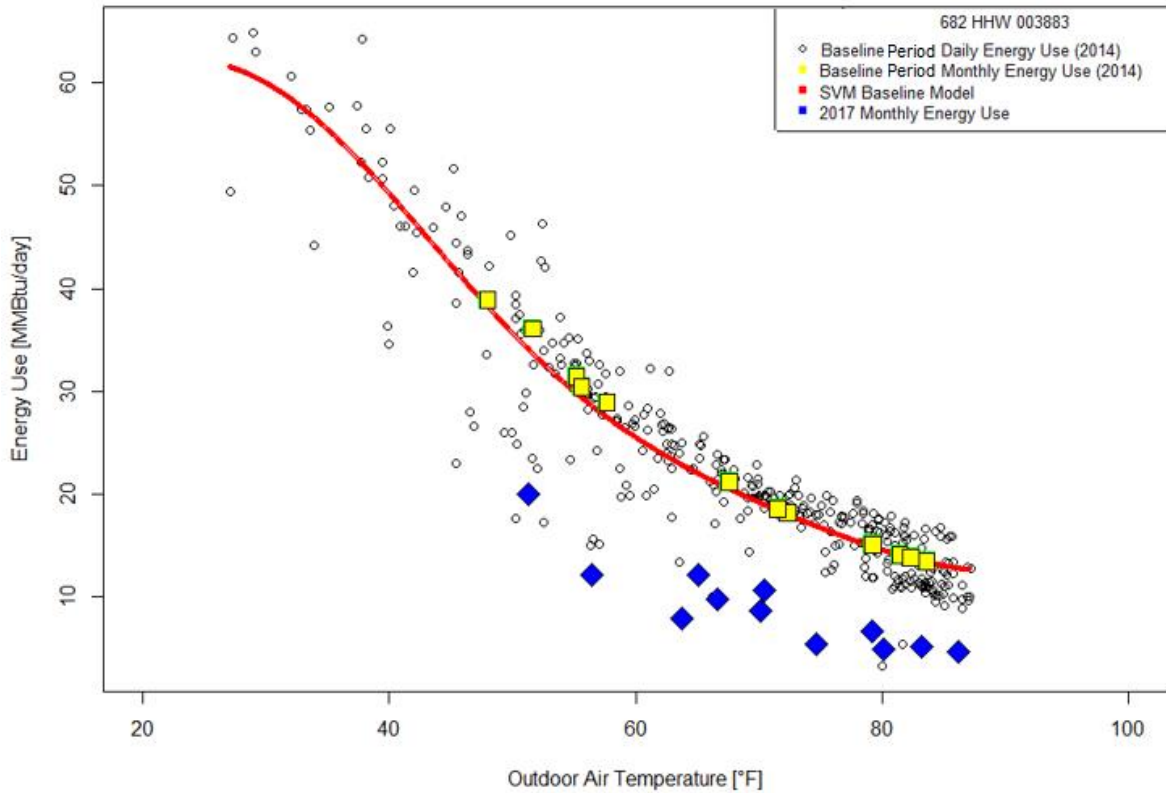
Savings Calculation

Figure F 33 - HHW versus Outdoor Air Temperature during the Baseline and Savings
Period

The baseline model is generated based on the daily data. Hence, if daily consumption

data is available for the savings period, savings can be calculated just by subtracting that

daily values from the corresponding SVM regression model values. If only monthly

values are available, such as monthly bill, those values need to be converted to monthly

average daily values by dividing by the number of days of the month. In the Figure F1

and Table F2, monthly average daily HHW is used for the savings period.

Table F 2 - Monthly Average Daily HHW Savings in 2017

Year	Month	OAT	Measured	Baseline	Savings	
		°F	MMBtu/day	MMBtu/day	MMBtu/day	%
2017	1	56.46	12.15	28.56	16.42	57.48
2017	2	63.71	7.95	22.86	14.91	65.23
2017	3	66.62	9.78	21.06	11.28	53.56
2017	4	70.13	8.62	19.12	10.50	54.91
2017	5	74.68	5.45	16.86	11.41	67.67
2017	6	80.15	4.89	14.51	9.62	66.32
2017	7	86.24	4.68	12.75	8.08	63.33
2017	8	83.20	5.17	13.48	8.31	61.66
2017	9	79.23	6.69	14.87	8.18	55.00
2017	10	70.43	10.61	18.97	8.36	44.05
2017	11	65.09	12.17	21.98	9.81	44.61
2017	12	51.31	20.04	34.06	14.03	41.18
Total			108.19	239.09	130.90	54.75