Multi-View LS-SVM Regression for Black-Box Temperature Prediction in Weather Forecasting

Lynn Houthuys*, Zahra Karevan* and Johan A. K. Suykens

Department of Electrical Engineering ESAT-STADIUS, KU Leuven
Kasteelpark Arenberg 10 B-3001 Leuven, Belgium
Email: {lynn.houthuys, zahra.karevan, johan.suykens}@esat.kuleuven.be
* Authors contributed equally

***Abstract*—In multi-view regression, we have a regression problem where the input data can be represented in multiple ways. These different representations are called views. The aim of multi-view regression is to increase the performance of using only one view by taking into account the information available from all views. In this paper, we introduce a novel multi-view regression model called Multi-View Least Squares Support Vector Machines (MV LS-SVM) regression. This model is formulated in the primal-dual setting typical to Least Squares Support Vector Machines (LS-SVM) where a coupling term is introduced in the primal objective. This form of coupling allows for some degree of freedom to model the different representations while being able to incorporate the information from all views in the training phase. This work was motivated by the challenge of predicting temperature in weather forecasting. Black-box weather forecasting deals with a large number of observations and features and is one of the most challenging learning task around. In order to predict the temperature in a city, the historical data from that city as well as from the neighboring cities are taking into account. In the past, the data for different cities were usually simply concatenated. In this work, we use MV LS-SVM to do temperature prediction by regarding each city as a different view. Experimental results on the minimum and maximum temperature prediction in Brussels, show the improvement of the multi-view method with regard to previous work and that this technique is competitive to the existing state-of-the-art methods in weather prediction.**

I. Introduction

In multi-view learning, data are described trough multiple forms of representation or *views*. You could, for example, have a dataset consisting of images presented as pixel arrays and a corresponding dataset consisting of the associated captions [1]. Both datasets describe the same set of images but use a different representation. Instead of just using one, multi-view learning aims at combining the information from all views to increase the performance of the learning task. Other examples of multi-view datasets include a set of webpages with the text on the page itself and the anchor text on the hyperlinks pointing to the webpage [2], news stories coming from multiple sources [3], using the user profile as well as the friend links for tasks on social networks [4], etc.. The information from multiple views can be combined even before any training process by means of simply concatenating the features or some more complex forms of data fusion like for example the work done by Shi et al. [5]. This is sometimes called early fusion. Another option is to train the models completely independent from each other and afterwards using a weighted combination to perform the learning task (as was done by Bekker et al. [6] for example), which is called late fusion. A third option is to have different submodels for each view and couple them (as, for example, Kumar et. al [7] for clustering and Koco & Capponi [8] for classification). This third option exploits the advantage of early fusion, namely that information from other views is exploited during training, while giving some degree of freedom to model the views differently like in late fusion.

Multi-view regression has been developed by using early fusion for applications like customer wallet estimation by Merugu et al. [9] and human pose estimation by Zhao et al. [10]. Zheng et al. [11] and Peng et al. [12] use late fusion to do multi-view regression by using a (weighted) sum of the models and performing regression in the subspace of each view separately. While regression is usually considered a supervised learning task, Kakade & Foster [13] developed a multi-view semi-supervised regression model based on canonical correlation analysis. Liu et. al. [14] use a multi-task multi-view method to predict Urban Water Quality where two views are coupled during training.

The multi-view method proposed in this paper is called *Multi-View Least Squares Support Vector Machines (MV LS-SVM) Regression*. It is cast in the primal-dual setting typical to Least Squares Support Vector Machines (LS-SVM) [15] where the separate models for each view are combined in the primal objective function so that information from other views is taking into account during training.

In this study, we focus on a MV LS-SVM regression for multi-variate time-series forecasting and evaluate the performance of the proposed method on an application of weather forecasting. Accurate weather prediction is an important challenging problem that can influence our daily lives in different ways. Lazo et al. reported that the U.S. public obtains more than 300 billion forecasts with a total estimated value of $31.5 billion each year [16]. State-of-the-art methods utilized Numerical Weather Prediction (NWP) to have a reliable weather forecasting. However, NWP approach is computationally intense method and uses thousands of CPUs to model the data [17]. Recently, there has been an increasing interest in

data-driven weather forecasting. LS-SVM is one of the popular machine learning methods that have shown good performance in literature for reliable weather predictions. Signoretto et al. [18] has used LS-SVM as a black-box approach in multi-task learning framework to predict temperature at 350 stations located in United States. Authors in [19] have used LS-SVM to learn the data considering grouping information of the data.

In order to forecast a weather condition in a particular city, the historical data of some nearby cities have been taken into account to have a reliable weather prediction. One may use a large feature vector which is the concatenation of the weather variables from different cities to have a reliable prediction. However, instead of simply concatenating the feature vectors of all cities, this paper regards each city as a separate view. By using the novel MV LS-SVM regression method, the temperature influences of each city can be modeled separately, with different kernel and regularization parameters for each city, while the coupling term in the primal model enforces interaction between the views which allows information form all other cities to be taken into account during the training phase.

We will denote matrices as bold uppercase letters and vectors as bold lowercase letters. The superscript $^{[v]}$ will denote the vth view for the multi-view method.

The rest of this paper is organized as follows: Section II overviews the necessary background namely, Least Squares Support Vector Machine Regression and Nonlinear AutoRegressive eXogenous model for weather forecasting. Section III introduces the novel Multi-View LS-SVM Regression model. Section IV presents the experiments done for predicting temperature in weather forecasting. The section first describes how the dataset was gathered, then discusses model selection followed by the discussion of the obtained results. Finally, Section V concludes this work.

II. Backgrounds

A. Least Squares Support Vector Machine Regression

This section summarizes the *Least Squares Support Vector Machine (LS-SVM)* model as described by Suykens et al. [15]. LS-SVM is a modification to the Vapnik Support Vector Machine (SVM) model [20] with a squared loss function and equality constraints which leads to solving a set of linear equations instead of a quadratic programming problem.

Given a training set of N data points $\{y_k, \mathbf{x}_k\}_{k=1}^N$ where $\mathbf{x}_k \in \mathbb{R}^d$ denotes the k-th input sample and $y_k \in \mathbb{R}$ the k-th target value, the primal formulation of the LS-SVM regression model is:

$$\min_{\mathbf{w},\mathbf{e},b} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{2}\gamma\mathbf{e}^T\mathbf{e} \tag{1}$$

$$\text{s.t. } \mathbf{y} = \boldsymbol{\Phi}\mathbf{w} + b\mathbf{1}_N + \mathbf{e}$$

where $\mathbf{e} \in \mathbb{R}^N$ are error variables, $\mathbf{y} = [y_1; \ldots; y_N]$ denotes the target vector, b is a bias terms and γ a positive real constant called the regularization parameter. The feature matrix $\boldsymbol{\Phi} \in \mathbb{R}^{N \times d_h}$ is defined as $\boldsymbol{\Phi} = [\boldsymbol{\varphi}(\mathbf{x}_1)^T; \ldots; \boldsymbol{\varphi}(\mathbf{x}_N)^T]$ where $\varphi : \mathbb{R}^d \to \mathbb{R}^{d_h}$ is the feature map which maps the d-dimensional input to a high dimensional feature space. Since this feature space is high dimensional, and can even be infinite dimensional, the function $\varphi(\cdot)$ is usually not explicitly defined. Rather, based on Mercer's condition [21], the function is implicitly defined trough the use of a positive kernel function $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ where $K(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i)^T\varphi(\mathbf{x}_j)$.

Note that this primal regression model is in fact equal to a ridge regression [22] cost function formulated in the feature space. When the data is high dimensional, it is not practical to work with this primal formulation; moreover, when $\mathbf{w}$ becomes infinite dimensional it is even not possible. Therefore, the dual problem is derived by taking the Lagrangian of the primal problem and deriving the KKT optimality conditions. After eliminating the primal variables $\mathbf{w}$ and $\mathbf{e}$ this results in the following dual formulation:

$$\left[\begin{array}{c|c} 0 & \mathbf{1}_N^T \\ \hline \mathbf{1}_N & \boldsymbol{\Omega} + \mathbb{I}_N/\gamma \end{array}\right] \left[\begin{array}{c} b \\ \hline \boldsymbol{\alpha} \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline \mathbf{y} \end{array}\right] \tag{2}$$

where $\mathbf{1}_N$ is an one column vector of dimension N and $\mathbb{I}_N$ is the identity matrix of dimension $N \times N$. The vector $\boldsymbol{\alpha}$ contains the dual variables which are linearly correlated with the error variables as $\alpha_k = \gamma e_k$ for $k = 1, \ldots, N$. $\boldsymbol{\Omega}$ is the kernel matrix whereon the kernel trick can be applied as follows:

$$\begin{aligned} \Omega_{ij} &= \varphi(\mathbf{x}_i)^T\varphi(\mathbf{x}_j) \\ &= K(\mathbf{x}_i, \mathbf{x}_j), \quad k, j = 1, \ldots, N. \end{aligned} \tag{3}$$

The resulting regressor in the dual space takes the form

$$\hat{y}(\mathbf{x}) = \sum_{k=1}^{N} \alpha_k K(\mathbf{x}, \mathbf{x}_k) + b. \tag{4}$$

In the training phase the model is built based on the training data, which means that Eq.(2) is solved to obtain b and $\boldsymbol{\alpha}$. The time complexity depends on the software used to compute this linear system but in general it will be $\mathcal{O}(N^3)$ since the left-hand side matrix is of dimension $(N+1) \times (N+1)$. Of course for unseen test data only Eq.(4) needs to be computed.

B. Nonlinear AutoRegressive eXogenous model for weather forecasting

As previously mentioned, accurate weather forecasting has become one of the major interests within the field of data-driven modeling. In our previous works, weather forecasting has been tackled as a time-series problem [19]. To predict the future weather conditions based on historical data, one may deploy the *Nonlinear AutoRegressive eXogenous* (NARX) [23] model, which is a discrete-time nonlinear system. In the NARX model, all historical weather elements of different weather stations can be used for model fitting. Obviously, the weather elements of the target city are also included in the feature vector. Assuming $y(t)$ and $u(t)$ are the output and input of the system at time t and $f(\cdot)$ is a non-linear function, the NARX model be formulated as follows:

$$\begin{aligned} y(t+s) \quad &= f(y(t), y(t-1), \ldots, y(t-p), \\ &\quad u(t), u(t-1), \ldots, u(t-q)) + \epsilon(t+s) \end{aligned} \tag{5}$$

where s is the number of steps ahead and $\epsilon(t+s)$ is the error of the model at time $t+s$. The values p and q are the *lags*, denoting the number of previous days in the time-series that are included in the model.

Let $\mathbf{x}^{(\xi)}(t)$ be a vector including all of the weather variables at time t in city ξ and $y^{(\xi)}(t+s)$ be the weather variables for s steps-ahead in city ξ. Considering ($p = q$) and c being the total number of weather stations, the weather forecasting model can be written as follows:

$$y^{(\xi)}(t+s) = f(\mathbf{x}^{\mathcal{C}}(t), \mathbf{x}^{\mathcal{C}}(t-1), \ldots, \mathbf{x}^{\mathcal{C}}(t-p)) + \epsilon(t+s) \tag{6}$$

where $\mathcal{C} = \{1, \ldots, c\}$ and $\xi \in \{1, ..., c\}$.

Note that considering equation (5), $\mathbf{x}^{(\xi)}(t)$ includes both $u(t)$ and $y(t)$ inputs in city ξ. In this study, $y^{\xi}(t+s)$ is considered to be the minimum and maximum temperature in Brussels for one to six days ahead. Furthermore, real measurements of weather variables in ten cities are involved in model fitting.

III. Multi-View LS-SVM Regression

In this section the model *Multi-View Least Squares Support Vector Machines (MV LS-SVM) Regression* is introduced. The primal formulation consist of multiple LS-SVM regression objectives and a coupling term. This newly introduced term enforces the alignment of the error variables over multiple views. So that when training on one view, the other views are taken into account.

Given a number of V views, a training set of N data points $\{y_k, \mathbf{x}_k^{[v]}\}_{k=1}^N$ for each view $v = 1, \ldots, V$ where $\mathbf{x}_k^{[v]} \in \mathbb{R}^{d^{[v]}}$ denotes the k-th input sample and $y_k \in \mathbb{R}$ the k-th target value, the primal formulation of the proposed model is:

$$\begin{aligned} \min_{\substack{\mathbf{w}^{[v]}, \mathbf{e}^{[v]}, \\ b^{[v]}}} \; & \frac{1}{2} \sum_{v=1}^{V} \mathbf{w}^{[v]^T} \mathbf{w}^{[v]} + \frac{1}{2} \sum_{v=1}^{V} \gamma^{[v]} \mathbf{e}^{[v]^T} \mathbf{e}^{[v]} \\ & + \rho \sum_{v,u=1; v \neq u}^{V} \mathbf{e}^{[v]^T} \mathbf{S}^{[v,u]} \mathbf{e}^{[u]} \end{aligned} \tag{7}$$

$$\text{s.t. } \mathbf{y} = \mathbf{\Phi}^{[v]} \mathbf{w}^{[v]} + b^{[v]} \mathbf{1}_N + \mathbf{e}^{[v]} \quad \text{for } v = 1, \ldots, V$$

where similarly to the LS-SVM notation, $\mathbf{y} = [y_1; \ldots; y_N]$ denotes the target vector and for each view v, $b^{[v]}$ are bias terms, $\gamma^{[v]}$ are positive real constants and $\mathbf{e}^{[v]} \in \mathbb{R}^N$ are error variables. $\mathbf{\Phi}^{[v]} \in \mathbb{R}^{N \times d_h^{[v]}}$ is defined as $\mathbf{\Phi}^{[v]} = [\varphi^{[v]}(\mathbf{x}_1^{[v]})^T; \ldots; \varphi^{[v]}(\mathbf{x}_N^{[v]})^T]$ where $\varphi^{[v]} : \mathbb{R}^{d^{[v]}} \to \mathbb{R}^{d_h^{[v]}}$ are the mappings to a high dimensional feature space related to the vth view. When comparing this formulation to the primal model of LS-SVM (Eq.(1)) one could see that it consist of V LS-SVM primal objective functions (one for each view) coupled by means of a *coupling term* defined as $\rho \sum_{v,u=1; v \neq u}^{V} \mathbf{e}^{[v]^T} \mathbf{S}^{[v,u]} \mathbf{e}^{[u]}$. This coupling term introduces a new regularization parameter $\rho > 0$ which is called the *coupling parameter*. The symmetric *coupling matrix* $\mathbf{S}^{[v,u]} \in \mathbb{R}^{N \times N}$, for $v, u = 1, \ldots, V$ and $v \neq u$, is positive definite and is used to model the correlation between the two sets of error variables. In practice one could define the coupling matrix equal to the identity matrix $\mathbb{I}_N$ or as $\mathbf{S}^{[v,u]} = \mathbf{D}^{[v]^{-\frac{1}{2}}} \mathbf{D}^{[u]^{-\frac{1}{2}}}$ for $v, u = 1, \ldots, V$, where $D_{ii}^{[v]} = \sum_j \varphi^{[v]}(\mathbf{x}_i^{[v]})^T \varphi^{[v]}(\mathbf{x}_j^{[v]})$ is a diagonal matrix called the degree matrix as previously used to weight score variables in Kernel Spectral Clustering by Alzate & Suykens [24]. If for a specific application there exist prior knowledge about the view correlation this could also be incorporated through the coupling matrix.

The Lagrangian of the primal problem is

$$\begin{aligned} &\mathcal{L}(\mathbf{w}^{[v]}, \mathbf{e}^{[v]}, b^{[v]}; \boldsymbol{\alpha}^{[v]}) = \\ &\frac{1}{2} \sum_{v=1}^{V} \mathbf{w}^{[v]^T} \mathbf{w}^{[v]} + \frac{1}{2} \sum_{v=1}^{V} \gamma^{[v]} \mathbf{e}^{[v]^T} \mathbf{e}^{[v]} + \rho \sum_{v,u=1; v \neq w}^{V} \mathbf{e}^{[v]^T} \mathbf{S}^{[v,u]} \mathbf{e}^{[u]} \\ &- \sum_{v=1}^{V} \boldsymbol{\alpha}^{[v]^T} (\mathbf{\Phi}^{[v]} \mathbf{w}^{[v]} + b^{[v]} \mathbf{1}_N + \mathbf{e}^{[v]} - \mathbf{y}) \end{aligned} \tag{8}$$

with conditions of optimality

$$\begin{cases} \dfrac{\partial \mathcal{L}}{\partial \mathbf{w}^{[v]}} = 0 \to \mathbf{w}^{[v]} = \mathbf{\Phi}^{[v]^T} \boldsymbol{\alpha}^{[v]}, \\ \dfrac{\partial \mathcal{L}}{\partial \mathbf{e}^{[v]}} = 0 \to \boldsymbol{\alpha}^{[v]} = \gamma^{[v]} \mathbf{e}^{[v]} + \rho \displaystyle\sum_{u=1; u \neq v}^{V} \mathbf{S}^{[v,u]} \mathbf{e}^{[u]}, \\ \dfrac{\partial \mathcal{L}}{\partial b^{[v]}} = 0 \to \mathbf{1}_N^T \boldsymbol{\alpha}^{[v]} = 0, \\ \dfrac{\partial \mathcal{L}}{\partial \boldsymbol{\alpha}^{[v]}} = 0 \to \mathbf{e}^{[v]} = \mathbf{\Phi}^{[v]} \mathbf{w}^{[v]} + b^{[v]} \mathbf{1}_N - \mathbf{y}, \\ \qquad \text{where } v = 1, \ldots, V. \end{cases} \tag{9}$$

Eliminating the primal variables $\mathbf{w}^{[v]}$ and $\mathbf{e}^{[v]}$ leads to the following linear system:

$$\left[\begin{array}{c|c} \mathbf{0}_{V \times V} & \mathcal{I}_M^T \\ \hline \mathbf{\Gamma}_M \mathcal{I}_M + \rho \mathbf{S}_\mathrm{M} \mathcal{I}_M & \mathbf{\Gamma}_M \mathbf{\Omega}_M + \mathbb{I}_{NV} + \rho \mathbf{S}_\mathrm{M} \mathbf{\Omega}_M \end{array}\right] \left[\begin{array}{c} \mathbf{b}_\mathrm{M} \\ \hline \boldsymbol{\alpha}_M \end{array}\right] = \left[\begin{array}{c} \mathbf{0}_V \\ \hline \mathbf{\Gamma}_M \mathbf{y}_\mathrm{M} + \rho \mathbf{S}_\mathrm{M} \mathbf{y}_\mathrm{M} \end{array}\right] \tag{10}$$

where $\mathbf{0}_V$ is a zero column vectors of dimension V, $\mathbf{0}_{V \times V}$ is a zero matrix of dimension $V \times V$ and $\mathbb{I}_{NV}$ is the identity matrix of dimension $NV \times NV$. The other matrices are defined as follows:

$$\mathcal{I}_{\boldsymbol{M}} = \text{blockdiag}\{\underbrace{\mathbf{1}_N, \ldots, \mathbf{1}_N}_{V \text{ times}}\} \in \mathbb{R}^{N \cdot V \times V}$$

$$\boldsymbol{\Gamma}_{\boldsymbol{M}} = \text{blockdiag}\{{\boldsymbol{\gamma}_{\boldsymbol{M}}}^{[1]}, \ldots, {\boldsymbol{\gamma}_{\boldsymbol{M}}}^{[V]}\} \in \mathbb{R}^{N \cdot V \times N \cdot V}$$

$${\boldsymbol{\gamma}_{\boldsymbol{M}}}^{[v]} = \text{diag}\{\underbrace{\gamma^{[v]}, \ldots, \gamma^{[v]}}_{N \text{ times}}\} \in \mathbb{R}^{N \times N}$$

$$\mathbf{S_M} = \begin{bmatrix} 0 & \mathbf{S}^{[1,2]} & \ldots & \mathbf{S}^{[1,V]} \\ \mathbf{S}^{[2,1]} & 0 & \ldots & \mathbf{S}^{[2,V]} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}^{[V,1]} & \mathbf{S}^{[V,2]} & \ldots & 0 \end{bmatrix} \in \mathbb{R}^{N \cdot V \times N \cdot V} \quad (11)$$

$$\boldsymbol{\Omega}_{\boldsymbol{M}} = \text{blockdiag}\{\boldsymbol{\Omega}^{[1]}, \ldots, \boldsymbol{\Omega}^{[V]}\} \in \mathbb{R}^{N \cdot V \times N \cdot V}$$

$$\mathbf{b_M} = \begin{bmatrix} b^{[1]} \\ \vdots \\ b^{[V]} \end{bmatrix} \in \mathbb{R}^V, \; \boldsymbol{\alpha}_{\boldsymbol{M}} = \begin{bmatrix} \boldsymbol{\alpha}^{[1]} \\ \vdots \\ \boldsymbol{\alpha}^{[V]} \end{bmatrix} \in \mathbb{R}^{N \cdot V}$$

$$\mathbf{y_M} = [\underbrace{\mathbf{y}^T, \ldots, \mathbf{y}^T}_{V \text{ times}}]^T \in \mathbb{R}^{N \cdot V}$$

with $\boldsymbol{\alpha}^{[v]}$ being dual variables. $\boldsymbol{\Omega}^{[v]} = \boldsymbol{\Phi}^{[v]} {\boldsymbol{\Phi}^{[v]}}^T$ are the kernel matrices with $\Omega_{kl}^{[v]} = \varphi^{[v]}(\mathbf{x}_k^{[v]})^T \varphi^{[v]}(\mathbf{x}_l^{[v]}) = K^{[v]}(\mathbf{x}_k, \mathbf{x}_l)$ where the kernel functions $K^{[v]} : \mathbb{R}^{d^{[v]}} \times \mathbb{R}^{d^{[v]}} \to \mathbb{R}$ are positive definite.

The resulting regressor can be defined in two possible ways:

1) Separately for each view v as:

$$\hat{y}^{[v]}(\mathbf{x}^{[\mathbf{v}]}) = \sum_{k=1}^{N} \alpha_k^{[v]} K^{[v]}(\mathbf{x}^{[v]}, \mathbf{x}_k^{[v]}) + b^{[v]}. \quad (12)$$

The estimated function can hence differ slightly among the different views.

2) Together on all views. A new function is defined as:

$$\hat{y}_{\text{total}}(\mathbf{x}^{[u]}) = \sum_{v=1}^{V} \beta_v \sum_{k=1}^{N} \alpha_k^{[v]} K^{[v]}(\mathbf{x}^{[v]}, \mathbf{x}_k^{[v]}) + b^{[v]} \quad (13)$$

which enforces that $\hat{y}_{\text{total}}(\mathbf{x}^{[1]}) = \ldots = \hat{y}_{\text{total}}(\mathbf{x}^{[V]})$, so the estimated function is equal over all views. The value of β_v for each $v = 1, \ldots, V$ can be $1/V$ to take the average, or can be calculated based on the error covariance matrix. In this last case the value of β_v for each $v = 1, \ldots, V$ can be chosen so that it minimizes the prediction error, similarly to how it is done for committee networks [25]. Alternatively, also the median could be considered.

The time complexity of the training phase is dominated by the computation of the linear system in Eq.(10). The left-hand side matrix is of dimension $V(N+1) \times V(N+1)$, hence the time complexity is given by $\mathcal{O}((V(N+1))^3)$. We can see that the multi-view model complexity is approximately V times the complexity of the classic LS-SVM model. Since in most real-life applications the number of views is very small (e.g. for the application in this paper $V = 10$), the influence of V is negligible and the time complexity of the training phase can thus be given by $\mathcal{O}(N^3)$.

IV. Experiments

A. Data

As mentioned before, in addition to the comparison between LS-SVM and MV LS-SVM regression for multivariate time-series, this study attempts to compare the performance of the proposed method with state-of-the-art approach in weather forecasting. In this study, the performance of data-driven methods are compared with the one of the Weather Underground website[1] which is one of the most popular ones in weather forecasting. Considering that this paper aims at forecasting the minimum and maximum temperature in Brussels for one to six days ahead, the predictions of these two weather elements for one to six days ahead in the test period were collected from the website.

Moreover, historical data of ten cities, including Brussels, Liege, Antwerp, Amsterdam, Eindhoven, Dortmund, London, Frankfurt, Groningen, and Dublin, were collected from the Weather Underground website. The data include real measurements of weather variables, such as minimum and maximum temperature, humidity and pressure, from the beginning of 2007 until mid 2014. To evaluate the performance of the proposed methods in different time periods, the experiments were conducted on two test sets in different seasons: one from mid-November 2013 until mid-December 2013 (test set Nov/Dec) and the other one from mid-April 2014 to mid-May 2014 (test set Apr/May).

Note that, the number of samples for model fitting is equal to the number of days from the beginning of 2007 until the day before the test point (varies from 2489 to 2667 points). Furthermore, there are 18 measured weather variables for each day in each city. The total number of features in LS-SVM is equal to $lag \times 198$, while there are $lag \times 18$ features for each view in multi-view case.

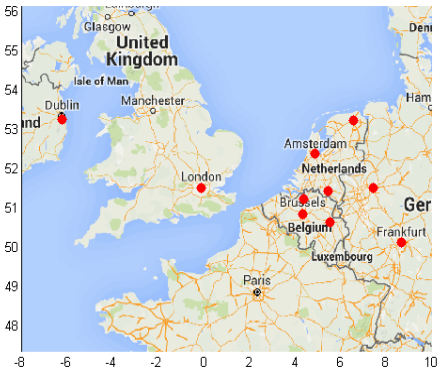


Fig. 1. Cities included in the model

B. Model selection

The feature vectors of each city correspond to a view. Since there are 10 cities taken into account for temperature prediction in Brussels, the number of views V equals 10.

[1] www.wunderground.com

TABLE I
MAE ON TEST DATA FOR MINIMUM AND MAXIMUM TEMPERATURE PREDICTION BY WEATHER UNDERGROUND (WU), LS-SVM ON THE CONCATENATED FEATURES, AND MV LS-SVM FOR THE TEST SET APR/MAY.

Step ahead	Temp.	WU	LS-SVM	MV LS-SVM *sep*	MV LS-SVM *mean*	MV LS-SVM *median*
1	Min	2.59	**1.35**	9.88	1.53	1.50
	Max	**1.07**	2.25	2.29	1.86	2.00
2	Min	2.37	2.07	10.11	**1.75**	**1.75**
	Max	**0.88**	2.28	2.48	2.18	2.14
3	Min	2.40	**1.89**	10.11	2	1.96
	Max	**1.51**	2.25	2.47	2.21	1.89
4	Min	1.92	2.03	10.4	2.07	**1.91**
	Max	2.22	**2.17**	2.47	2.21	2.21
5	Min	**1.48**	2.07	9.95	2.17	2.18
	Max	**2.07**	2.1	2.11	2.14	2.11
6	Min	2.08	2.07	10.23	**2.06**	2.11
	Max	2.22	2.35	2.46	2.14	**2.11**

TABLE II
MAE ON TEST DATA FOR MINIMUM AND MAXIMUM TEMPERATURE PREDICTION BY WEATHER UNDERGROUND (WU), LS-SVM ON THE CONCATENATED FEATURES, AND MV LS-SVM FOR THE TEST SET NOV/DEC.

Step Ahead	Temp.	WU	LS-SVM	MV LS-SVM *sep*	MV LS-SVM *mean*	MV LS-SVM *median*
1	Min	1.57	**1.43**	5.69	1.53	1.50
	Max	**0.96**	1.14	1.60	1.21	1.14
2	Min	1.57	1.78	5.99	1.60	**1.54**
	Max	**1.15**	1.25	2.23	1.46	2.21
3	Min	1.76	2.14	5.21	**1.61**	1.79
	Max	**1.26**	1.35	2.69	2.04	2.04
4	Min	**1.23**	2.07	5.10	1.47	1.50
	Max	**1.38**	1.39	2.74	2.39	2.64
5	Min	1.76	2.07	5.01	1.54	**1.46**
	Max	1.65	**1.28**	2.71	2.64	2.61
6	Min	2.42	2.17	5.10	**1.57**	1.75
	Max	2.26	**1.46**	2.67	2.61	**2.39**

The results obtained by the MV LS-SVM regression model depend on the lag variable, the choice of the coupling matrix S, the manner in which the resulting regressor is obtained (separately or together on all views), the choice of kernels and kernel parameters, the parameters $\gamma^{[v]}$ for each view v and on the coupling parameter ρ.

For this application, it was chosen to not couple specific correlations between views and hence to choose $S^{[v,u]} = \mathbb{I}_N$ for all $v, u = 1, \ldots, 10$. For defining the final regressor three options were considered:

- Defined separately on each view (see Eq.(12)).
- Defined together on all views by taking the mean (hence Eq.(13) with $\beta_v = 1/10$ for all $v = 1, \ldots, 10$).
- Defined together on all views by taking the median

These three variations of MV LS-SVM will be denoted in the results section as *sep*, *mean* and *median* respectively.

The radial basis function (RBF) kernel is chosen for all views, so the corresponding kernel functions are $K^{[v]}(\mathbf{x}_i^{[v]}, \mathbf{x}_j^{[v]}) = \exp\left(-\frac{||\mathbf{x}_i^{[v]} - \mathbf{x}_j^{[v]}||^2}{2\sigma^{[v]2}}\right)$ for $v = 1, \ldots, 10$ and where $\sigma^{[v]}$ is the kernel parameter for each view.

To decrease tuning complexity the parameters $\sigma^{[v]}$ and $\gamma^{[v]}$ were not tuned in the multi-view setting. Instead view-specific optimal values were chosen, which were obtained by previous tuning when using LS-SVM with features from only the vth city.

The lag variable ($\in \{7, 9, 11, 13\}$) and the coupling parameter ρ were obtained by means of 5-fold cross validation where simulated annealing is used to optimize the Mean Absolute Error (MAE).

C. Results

The MAE on test data for minimum and maximum temperature prediction by Weather Underground, LS-SVM based on the concatenated features and MV LS-SVM are depicted in Table I for the test set Apr/May and in Table II for the test set Nov/Dec.

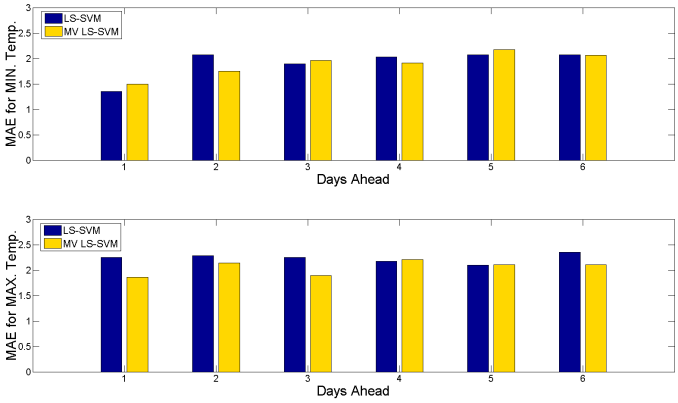


Fig. 2. Comparison of the MAE on minimum (top) and maximum (bottom) temperature prediction between LS-SVM on the concatenated features and MV LS-SVM for the test set Apr/May.

When we look at minimum temperature prediction for both test sets, it can be observed that the black-box modeling techniques considered here perform very well in comparison to Weather Underground. More specifically, Weather Underground is only able to achieve a more accurate minimum temperature prediction than the best black-box model in two out of the twelve test cases. Furthermore, for long term (4 to 6 days ahead) maximum temperature prediction, black-box models are able to outperform Weather Underground in most cases. For short term (1 to 3 days ahead) maximum temperature prediction, however, Weather Underground is always able to obtain the lowest MAE.

Considering the results for the test set Apr/May, it is shown that the performance is improved when the multi-view method is used. This improvement can be seen more clearly in Figure 2 where the best results obtained by MV LS-SVM and LS-SVM on the concatenated features are shown. We can see that MV LS-SVM usually outperforms LS-SVM on this test set.

For maximum temperature prediction, LS-SVM only has a slightly (a difference of 0.04 and 0.01 in MAE) better

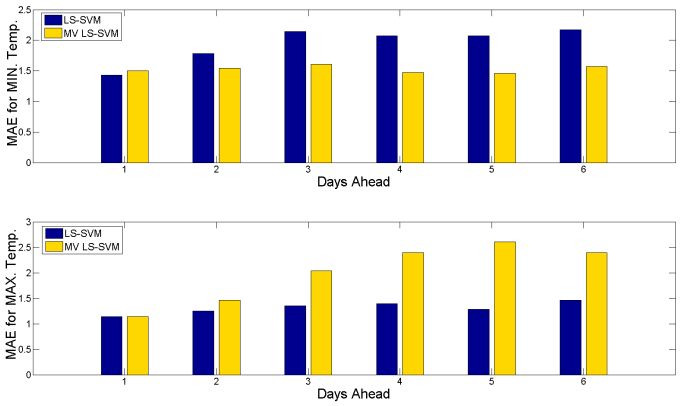


Fig. 3. Comparison of the MAE on minimum (top) and maximum (bottom) temperature prediction between LS-SVM on the concatenated features and MV LS-SVM for the test set Nov/Dec.

performance for 4 and 5 days ahead, whereas MV LS-SVM is able to outperform LS-SVM with a maximum difference of 0.39 in MAE. Furthermore, MV LS-SVM obtains a better regression accuracy for 4 and 6 steps ahead than Weather Underground, whereas LS-SVM was only able to outperform Weather Underground for 4 steps ahead.

For minimum temperature prediction, LS-SVM achieves a better performance in three test cases with at most a difference of 0.15 in MAE. However, MV LS-SVM is able to outperform LS-SVM in the three remaining test cases with a maximum improvement of 0.32 in MAE. Furthermore, while LS-SVM is able to outperform Weather Underground in 4 test cases (1,2,3 and 6 step ahead), MV LS-SVM is able to do this in 5 test cases (1,2,3,4 and 6 step ahead).

While for the test Apr/May, the improvement of MV LS-SVM with regard to LS-SVM was similar in minimum and maximum temperature prediction cases, this is not the case for the test set Nov/Dec. This improvement is depicted in Figure 3 where the best result obtained by MV LS-SVM and LS-SVM based on the concatenated features is shown. Although we can clearly see that MV LS-SVM is able to improve the results for minimum temperature, for maximum temperature, the multi-view method is not performing very well.

For minimum temperature prediction, LS-SVM only achieves a moderately (a difference of 0.04 in MAE) better performance for 1 step ahead prediction, however, MV LS-SVM outperforms LS-SVM on all remaining step ahead predictions with a maximum improvement of 0.61 in MAE. Moreover, while LS-SVM only achieves a better regression accuracy than Weather Underground for 1 and 6 step ahead prediction, MV LS-SVM achieves this for all step ahead predictions except for 4 steps ahead.

For maximum temperature prediction, however, it is clear that MV LS-SVM does not behave adequately. While it is able to achieve the same accuracy as LS-SVM for 1 step ahead prediction, it performs badly for longer term prediction. Unlike LS-SVM which is able to surpass Weather Underground for long term prediction, for the multi-view method this is not the case.

Table I and Table II report the MAE for three variations of MV LS-SVM where the final regressor is computed differently. It indicates that the variation *sep* (final regression is performed separately on each view) is not well suited for this dataset, especially for minimum temperature prediction. Whereas both *mean* and *median* (final regression is performed on all views together) usually perform very well.

V. Conclusion

This paper proposes a novel multi-view regression model called Multi-View Least Squares Support Vector Machines (MV LS-SVM) that performs regression when two or more views are available. The model is based on Least Squares Support Vector Machines where the primal model consist of the summation of the view-specific primal LS-SVM formulations plus a coupling term between each pair of views. This form of coupling allows for a degree of freedom in modeling the different views while also allows taking into account the information from all views in the training phase.

The model was tested on the challenge of temperature prediction in weather forecasting. While previously historical data from neighboring cities was concatenated for prediction temperature in a certain city, we proposed to regard the data from each city as a different view. We aimed at predicting the minimum and maximum temperature in Brussels by using its ten nearby cities. We compared our results to Weather Underground, which is one of the popular weather forecasting companies, and to LS-SVM based on the concatenated data.

From our results, we can conclude that the proposed method MV LS-SVM can often outperform LS-SVM for minimum and maximum temperature prediction. They also showed that MV LS-SVM can outperform Weather Underground for minimum temperature prediction and is competitive with it for maximum temperature in the test set Apr/May. These results suggest the merit of using a multi-view approach instead of simply concatenating the features.

In future work, it could be interesting to examine the poor performance of MV LS-SVM on long term maximum temperature prediction on the Nov/Dec test set. Furthermore, we will research other uses of multi-view learning in weather forecasting.

Acknowledgments

The research leading to these results has received funding from the European Research Council under the European Union? Seventh Framework Programme (FP7/2007-2013) / ERC AdG ADATADRIVE- B (290923). This paper reflects only the authors views and the Union is not liable for any use that may be made of the contained information. Research Council KUL: CoE PFV/10/002 (OPTEC), BIL12/11T; PhD/Postdoc grants Flemish Government: FWO: projects: G.0377.12 (Structured systems), G.088114N (Tensor based data similarity); PhD/Postdoc grant iMinds Medical Information Technologies SBO 2015 IWT: POM II SBO 100031 Belgian Federal Science Policy Office: IUAP P7/19 (DYSCO, Dynamical systems, control and optimization, 2012-2017).

REFERENCES

[1] T. Kolenda, L. K. Hansen, J. Larsen, and O. Winther, "Independent component analysis for understanding multimedia content," *Proceedings of IEEE Workshop on Neural Networks for Signal Processing*, vol. 12, pp. 757–766, 2002.

[2] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," *Conference on Learning Theory*, pp. 92–100, 1998.

[3] D. Greene and P. Cunningham, "A matrix factorization approach for integrating multiple data views," *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, pp. 423–438, 2009.

[4] Y. Yang, C. Lan, X. Li, J. Huan, and B. Luo, "Automatic social circle detection using multi-view clustering," *ACM Conference on Information and Knowlwedge Management (CIKM)*, pp. 1019–1028, 2014.

[5] S. Yu, L.-C. Tranchevent, X. Liu, W. Glanzel, J. A. K. Suykens, B. De Moor, and Y. Moreau, "Optimized data fusion for kernel k-means clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 5, pp. 1031–1039, 2012.

[6] A. Bekker, M. Shalhon, H. Greenspan, and J. Goldberger, "Multi-view probabilistic classification of breast microcalcifications," *IEEE Transactions on Medical Imaging*, vol. 35, no. 2, pp. 645–653, 2016.

[7] A. Kumar, P. Rai, and H. Daume, "Co-regularized multi-view spectral clustering," *Neural Information Processing Systems 2011*, pp. 1413–1421, 2011.

[8] S. Koço and C. Capponi, "A boosting approach to multiview classification with cooperation," *European Conference on Machine Learning and Knowledge Discovery in Databases*, vol. 2, pp. 209–228, 2011.

[9] S. Merugu, S. Rosset, and C. Perlich, "A new multi-view regression approach with an application to customer wallet estimation," *Proceedings of the 12th ACM SIGKDD international conference on Knowledge Discovery and Data mining*, pp. 656–661, 2006.

[10] X. Zhao, Y. Fu, H. Ning, Y. Liu, and T. S. Huang, "Human Pose Estimation with Regression by Fusing Multi-View Visual Information," *Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 7, pp. 957–966, 2010.

[11] S. Zheng, X. Cai, C. Ding, F. Nie, and H. Huang, "A Closed Form Solution to Multi-View Low-Rank Regression," *AAAI Conference on Artificial Intelligence*, vol. 29, pp. 1973–1979, 2016.

[12] H. Peng, K. Li, B. Li, H. Ling, W. Xiong, and W. Hu, "Predicting Image Memorability by Multi-view Adaptive Regression," *Proc. of ACM Multimedia Conference (MM)*, pp. 1147–1150, 2015.

[13] S. Kakade and D. Foster, "Multi-view regression via canonical correlation analysis," *Conference on Learning Theory*, vol. 20, pp. 82–96, 2007.

[14] Y. Liu, Y. Liang, S. Liu, D. S. Rosenblum, and Y. Zheng, "Predicting Urban Water Quality with Ubiquitous Data," pp. 1–14, 2016. [Online]. Available: http://arxiv.org/abs/1610.09462

[15] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines*. World Scientific, 2002.

[16] J. K. Lazo, R. E. Morss, and J. L. Demuth, "300 billion served: Sources, perceptions, uses, and values of weather forecasts," *Bulletin of the American Meteorological Society*, vol. 90, no. 6, pp. 785–798, 2009.

[17] P. Bauer, A. Thorpe, and G. Brunet, "The quiet revolution of numerical weather prediction," *Nature*, vol. 525, no. 7567, pp. 47–55, 2015.

[18] M. Signoretto, E. Frandi, Z. Karevan, and J. A. K. Suykens, "High level high performance computing for multitask learning of time-varying models," *IEEE Symposium on Computational Intelligence in Big Data*, pp. 1–6, 2014.

[19] Z. Karevan and J. A. K. Suykens, "Clustering-based feature selection for black-box weather temperature prediction," *International Joint Conference on Neural Networks*, pp. 1–8, 2016.

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

[21] J. Mercer, "Functions of positive and negative type, and their connection with the theory of integral equations," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 209, pp. 415–446, 1909.

[22] G. Golub and C. Van Loan, *Matrix Computations*. Baltimor MD: Johns Hopkins University Press, 1989.

[23] I. Leontaritis and S. A. Billings, "Input-output parametric models for non-linear systems part i: deterministic non-linear systems," *International journal of Control*, vol. 41, no. 2, pp. 303–328, 1985.

[24] C. Alzate and J. A. K. Suykens, "Multiway spectral clustering with out-of-sample extensions through weighted kernel PCA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 2, pp. 335–347, 2010.

[25] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.