Combination forecasts of tourism demand with machine learning models

Oscar Claveria[1]*, Enric Monte[2], Salvador Torra[3]

[1] AQR-IREA, University of Barcelona (UB)
[2] Department of Signal Theory and Communications, Polytechnic University of Catalunya (UPC)
[3] Riskcenter-IREA, Department of Econometrics and Statistics, University of Barcelona (UB)

Oscar Claveria
AQR-IREA (Institute of Applied Economics Research)
University of Barcelona
Diagonal, 690
08034 Barcelona
Spain
Tel.: +34-934021825
oclaveria@ub.edu

Enric Monte
Department of Signal Theory and Communications
Polytechnic University of Catalunya
Jordi Girona, 1-3
08034 Barcelona
Spain
Tel.: +34-934016435
enric.monte@upc.edu

Salvador Torra
Riskcenter-IREA
University of Barcelona
Diagonal, 690
08034 Barcelona
Spain
Tel.: +34-934024318
storra@ub.edu

* **Corresponding Author:**
Oscar Claveria, University of Barcelona, 08034 Barcelona, Spain. Email: oclaveria@ub.edu

Combination forecasts of tourism demand with machine learning models

The main objective of this study is to analyse whether the combination of regional predictions generated with machine learning (ML) models leads to improved forecast accuracy. With this aim we construct one set of forecasts by estimating models on the aggregate series, another set by using the same models to forecast the individual series prior to aggregation, and then we compare the accuracy of both approaches. We use three ML techniques: Support Vector Regression (SVR), Gaussian Process Regression (GPR) and Neural Network (NN) models. We use an ARMA model as a benchmark. We find that ML methods improve their forecasting performance with respect to the benchmark as forecast horizons increase, suggesting the suitability of these techniques for mid- and long-term forecasting. In spite of the fact that the disaggregated approach yields more accurate predictions, the improvement over the benchmark occurs for shorter forecast horizons with the direct approach.

I. Introduction

ML methods such as SVR and NN models are attracting increasing attention for economic time series prediction. SVR has been widely used for financial forecasting (Kim, 2003; Huang *et al.*, 2005), but few attempts have been made for tourism demand forecasting (Chen and Wang, 2007; Hong *et al.*, 2011; Wu *et al.*, 2012; Akin, 2015). A complete summary of NN forecasting can be found in Zhang *et al.* (1998). While the multi-layer perceptron (MLP) network has been commonly applied for tourism demand forecasting (Padhi and Aggarwal, 2011; Lin *et al.*, 2011; Claveria and Torra, 2014), radial basis function (RBF) networks have been less implemented (Cang, 2014).

GPR models are recently being applied with forecasting purposes (Banerjee *et al.*, 2008). The GPR can be regarded as a supervised learning method based on a generalized linear regression that locally estimates forecasts by the combination of values in a kernel (Rasmussen, 1996). GPR is a powerful, non-parametric tool for regression in high dimensional spaces, but to our knowledge there is only one previous study that uses GPR for tourism forecasting (Wu *et al.*, 2012). To fill this gap, we design a multiple-step-ahead forecasting experiment to compare GPR to SVR and NN models.

The main aim of this study is to analyse the relative improvement on forecast accuracy of ML methods over a linear stochastic process using two alternative approaches. First we apply the direct approach, which consists in forecasting the aggregate series. Then we use the same models to forecast the individual series for each region prior to aggregation at a national level. Finally, we compare the forecasting performance of both approaches.

Several authors have found evidence that combining forecasts tends to yield more accurate predictions than direct approaches (Bates and Granger, 1969; Stock and Watson, 2004; Ruth, 2008). We extend previous research by assessing this approach with ML techniques at a regional level.

The remainder of this study is organized as follows. Section 2 presents the different ML methods applied in the study. Section 3 presents the data, describes the experimental settings, and reports the results. Section 5 concludes.

II. Machine learning methods

2.1. Gaussian Process Regression

GPR can be conceived as a method of interpolation. The training set is assumed to be drawn from the process:

$$y_i = f(x_i) + \varepsilon \qquad \text{with } \varepsilon \sim N(0, \sigma^2) \qquad (1)$$

where x_i is an input vector and y_i is a scalar output. For notational convenience, we aggregate the inputs and the outputs into matrix $X = [x_1, x_2, ..., x_n]$ and $y = [y_1, y_2, ..., y_n]$ respectively. Thus, the GPR model is defined by the mean μ and the variance Σ:

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

$$\Sigma = K(X^*, X^*) - K(X^*, X)[K(X, X) + \sigma^2 I]^{-1} K(X, X^*) \qquad (3)$$

$K(X, X)$ is the covariance matrix, also called kernel matrix. In this study we make use of a Gaussian radial basis kernel with a linear trend:

$$K_{ij} = k(x_i, x_j) = v^2 \exp\left(-\frac{(x_i - x_j)^2}{2\lambda^2}\right) + \gamma x_i^T x_j + \kappa \qquad (4)$$

The parameter λ determines the distance between x_i and x_j for f_i to be unrelated to f_j, while v^2 controls the prior variance. Alternative sets of kernels are discussed in MacKay (2003). For a comprehensive introduction to GPR see Williams and Rasmussen (2006).

2.2. Support Vector Regression

The SVR was first proposed by Drucker *et al.* (1997). The objective is to infer a function $f(x_t)$ such that its output is as near as possible to the desired output d_t:

$$f(x_t) = \omega\phi(x_t) + b \qquad (5)$$

x_t is the input vector, ω a weight vector, b a constant, and $\varphi(x_t)$ is the non-linear function that maps the input into a high-dimensional feature space F within a tube of radius ε that maps the input data vector x_t into a high-dimensional feature space F. The $\varepsilon-$ insensitive loss function L_ε does not take into account the errors within the ε-tube:

$$L_\varepsilon(d_t, y_t) = \begin{cases} |d_t - y_t| - \varepsilon, & |d_t - y_t| \geq \varepsilon \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

The introduction of two positive slack variables ξ_t and ξ_t^* allows to reformulate the SVR as an optimization problem:

$$Minimize \quad \frac{1}{2}\|\omega\|^2 + C\sum_{t=1}^{n}(\xi_t + \xi_t^*) \quad \text{subjected to} \quad \begin{cases} d_t - \omega\phi(x_t) - b \leq \varepsilon + \xi_t \\ \omega\phi(x_t) + b - d_t \leq \varepsilon + \xi_t^* \end{cases} \quad \xi_t^{(*)}, C \geq 0 \tag{7}$$

where ω is a weight vector and b is a constant. The selection of the hyperparameters ε and C is done by means of cross-validation. To solve (7), we can introduce two Lagrange multipliers and a kernel function $K(x_i, x_j)$ in the decision function. In this study we use three different kernels:

Linear kernel (L-SVR) $\qquad K(x, y) = a_1 x * y + a_2$ $\qquad\qquad$ (8)

Polynomial kernel (P-SVR) $\qquad K(x, y) = (a_1 x * y + a_2)^h$ $\qquad\qquad$ (9)

Gaussian RBF kernel (G-SVR) $\quad K(x, y) = \exp\left[(1/\delta^2)(x - y)^2\right]$ $\qquad$ (10)

Where a_1 and a_2 are constants, h is the degree of the polynomial kernel, and δ^2 is the bandwidth of the Gaussian RBF kernel. For a comprehensive introduction to SVR see Cristianini and Shawhe-Taylor (2000).

2.3. Neural Networks

The RBF architecture can be specified as:

$$y_t = \beta_0 + \sum_{j=1}^{q} \beta_j g_j(x_{t-i})$$
$$g_j(x_{t-i}) = \exp\left(-\sum_{j=1}^{p}(x_{t-i} - \mu_j)^2 \Big/ 2\sigma_j^2\right) \tag{11}$$

Where $\{x_{t-i}; i = 1, \ldots, p\}$ and $\{\beta_j; \sigma_j; j = 1, \ldots, q\}$. The output vector is denoted by y_t, x_{t-i} is the input value, g_j the activation function, μ_j the centroid vectors, β_j the weights, and σ_j the spread for neuron j. We denote q as the number of neurons in the hidden layer, which ranges from 5 to 30, increasing for longer forecast horizons.

The MLP architecture is given by:

$$y_t = \beta_0 + \sum_{j=1}^{q} \beta_j g\left(\sum_{i=1}^{p} w_{ij} x_{t-i} + w_{0j}\right) \tag{12}$$

Where $\{x_{t-i}; i = 1,\ldots, p\}$, $\{w_{ij}; i = 1,\ldots, p; j = 1,\cdots, q\}$, $\{\beta_j; j = 1,\ldots, q\}$. The weights connecting the input with the hidden layer are denoted by w_{ij}, while g is the non-linear function of the neurons in the hidden layer. The number of neurons is estimated by cross-validation. A complete summary on the implementation of NNs can be found in Haykin (2008).

III. Results

The data set used in the empirical experiment are collected from the Spanish Statistical Office (National Statistics Institute – INE – www.ine.es). It covers 183 monthly observations of monthly tourist arrivals at a regional level from 1999:01 to 2014:03. The first 52% observations are selected as the initial training set, the next 33% as the validation set, and the last 15% as the test set. For an iterated multi-step-ahead forecasting comparison the partition between train and test sets is done sequentially.

The forecasting performance of the different models is assessed for different time horizons (1, 2, 3, 6 and 12 months) by computing the Relative Mean Absolute Percentage Error (rMAPE). The rMAPE ponders the MAPE of the model under evaluation against the MAPE of the benchmark model. We use an ARMA model as a benchmark. Table 1 presents the results obtained with the direct approach, while Table 2 shows the results obtained by aggregating the regional forecasts by summation.

Table 1

Table 2

With both forecasting approaches the lowest rMAPE values are obtained for longer forecast horizons. We do not find significant differences between the different techniques. This result indicates that ML methods improve their forecasting performance with respect to linear models as forecast horizons increase.

In line with previous research, the disaggregated approach yielded lower forecast errors than direct predictions, but when comparing the improvement of ML methods over the benchmark, we find that ML models outperform the benchmark for six- and twelve-month ahead forecasts, while with the direct approach the improvement occurs from three-month ahead forecasts on.

IV.   Conclusion

Artificial intelligence methods based on ML have attracted increasing interest to refine predictions. This study analyses the forecasting performance of SVR, GPR and NN models when combining forecasts at a regional level. This is the first study that compares both forecasting approaches with ML techniques at a regional level.

When comparing the improvement of ML methods over the benchmark, we find that with the direct approach the relative gain occurs one step-ahead before than when combining regional forecasts.

With both approaches we obtain major improvements of forecast accuracy as forecast horizons increase, but we do not find significant differences between the different techniques. This result suggests that ML techniques are especially suitable for mid- and long-term forecasting.

References

Akin, M. 2015. A novel approach to model selection in tourism demand modeling. Tourism Management 48, 64–72.

Banerjee, S., Gelfand, A. E., Finley, A. O, Sang, H. 2008. Gaussian predictive process models for large spatial data sets. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 70, 825–848.

Bates, J. M, Granger, C. W. J. 1969. The combination of forecasts. Operational Research Quarterly 20, 451–468.

Cang, S. 2014. A comparative analysis of three types of tourism demand forecasting models: individual, linear combination and non-linear combination. International Journal of Tourism Research 16, 596–607.

Chen, K. Y., Wang, C. H. 2007. Support vector regression with genetic algorithms in forecasting tourism demand. Tourism Management, 28: 215–226.

Claveria, O., Torra, S. 2014. Forecasting tourism demand to Catalonia: Neural networks vs. time series models. Economic Modelling 36, 220–228.

Cristianini, N., Shawhe-Taylor, J. 2000. An introduction to support vector machines and other kernel-based learning methods. Cambridge: Cambridge University Press.

Drucker, H., Burges, C. J. C., Kaufman, L. Smola, A., Vapnik, V. 1997. Support vector regression machines. In M. Mozer, M. Jordan, and T. Petsche (Eds.), Advances in Neural Information Processing Systems, Vol. 9 (pp. 155–161). Cambridge, MA: MIT Press.

Haykin S. 2008. Neural networks and learning machines (3rd Edition). New Jersey: Prentice Hall.

Hong, W., Dong, Y. Chen, L., Wei S. 2011. SVR with hybrid chaotic genetic algorithms for tourism demand forecasting. Applied Soft Computing 11, 1881–1890.

Huang, W., Nakamori, Y., Wang, S. Y. 2005. Forecasting stock market movement direction with support vector machine. Computers & Operations Research 32, 2513–2522.

Kim K. 2003. Financial Time Series Forecasting using support vector machines. Neurocomputing 55, 307-319.

Lin, C., Chen, H., Lee, T. 2011. Forecasting tourism demand using time series, artificial neural networks and multivariate adaptive regression splines: Evidence from Taiwan. International Journal of Business Administration 2, 14–24.

MacKay, D. J. C. 2003. Information theory, inference, and learning algorithms. Cambridge: Cambridge University Press.

Padhi, S. S., Aggarwal, V. 2011. Competitive revenue management for fixing quota and price of hotel commodities under uncertainty. International Journal of Hospitality Management 30, 725–734.

Rasmussen, C. E. 1996. The infinite Gaussian mixture model. Advances in Neural Information Processing Systems 8, 514–520.

Ruth, K. 2008. Macroeconomic forecasting in the EMU. Does disaggregate modeling improve forecast accuracy? Journal of Policy Modeling 30, 417–429.

Stock, J. H., Watson, M. W. 2004. Combination forecasts of output growth in a seven-country data set. Journal of Forecasting 23, 405–430.

Williams, C. K. I., Rasmussen, C. E. 2006. Gaussian processes for machine learning. Cambridge, MA: The MIT Press.

Wu, Q., Law, R., Xu, X. 2012. A spare Gaussian process regression model for tourism demand forecasting in Hong Kong. Expert Systems with Applications 39, 4769–4774.

Zhang G, Putuwo, B. E., Hu, M. Y. 1998. Forecasting with artificial neural networks: the state of the art. International Journal of Forecasting 14, 35–62.

Table 1

Forecast accuracy – Machine learning models with respect to ARMA -rMAPE (2013:01-2014:01)

Direct forecasts					
Forecasting horizon					
	h=1	h=2	h=3	h=6	h=12
Method					
L-SVR	2.688	1.759	1.007	0.612	0.665
P-SVR	3.275	2.165	0.840	0.641	0.756
G-SVR	2.447	1.818	0.949	0.535	0.614
GPR	2.384	1.657	0.954	0.489	0.643
RBF NN	2.362	1.518	0.947	0.568	0.635
MLP NN	2.384	1.570	0.922	0.461	0.659

Note: The rMAPE ponders the MAPE of the model under evaluation against the MAPE of the benchmark model. We use an ARMA model as a benchmark.

Table 2

Forecasting performance – Machine learning models with respect to ARMA -rMAPE (2013:01-2014:01)

Disaggregated forecasts (Simple mean combination forecast)					
Forecasting horizon					
	h=1	h=2	h=3	h=6	h=12
Method					
L-SVR	2.219	1.762	1.173	0.617	0.681
P-SVR	2.782	1.855	1.006	0.628	0.880
G-SVR	2.213	1.721	1.066	0.587	0.678
GPR	2.167	1.697	1.173	0.568	0.652
RBF NN	2.135	1.547	1.160	0.631	0.652
MLP NN	2.038	1.575	1.136	0.545	0.619

See notes at bottom of Table 1.