

# Bayesian Regression Approach for Building and Stacking Predictive Models in Time Series Analytics

Bohdan M. Pavlyshenko

Ivan Franko National University of Lviv  
Lviv, Ukraine

b.pavlyshenko@gmail.com, [www.linkedin.com/in/bpavlyshenko/](http://www.linkedin.com/in/bpavlyshenko/)

**Abstract.** The paper describes the use of Bayesian regression for building time series models and stacking different predictive models for time series. Using Bayesian regression for time series modeling with nonlinear trend was analyzed. This approach makes it possible to estimate an uncertainty of time series prediction and calculate value at risk characteristics. A hierarchical model for time series using Bayesian regression has been considered. In this approach, one set of parameters is the same for all data samples, other parameters can be different for different groups of data samples. Such an approach allows using this model in the case of short historical data for specified time series, e.g. in the case of new stores or new products in the sales prediction problem. In the study of predictive models stacking, the models ARIMA, Neural Network, Random Forest, Extra Tree were used for the prediction on the first level of model ensemble. On the second level, time series predictions of these models on the validation set were used for stacking by Bayesian regression. This approach gives distributions for regression coefficients of these models. It makes it possible to estimate the uncertainty contributed by each model to stacking result. The information about these distributions allows us to select an optimal set of stacking models, taking into account the domain knowledge. The probabilistic approach for stacking predictive models allows us to make risk assessment for the predictions that are important in a decision-making process.

**Keywords:** Time Series, Bayesian Regression, Machine Learning, Stacking, Forecasting, Sales

## 1 Introduction

Time series analytics is an important part of modern data science. The examples of different time series approaches are in [2, 3, 5, 10, 11, 21, 22]. In [6, 9, 18–20, 23], a range of ensemble-based methods for classification problems is considered. In [14], the authors studied a lagged variable selection, hyperparameter optimization, as well as compared classical and machine learning based algorithms for time series. Sales prediction is an important part of modern business intelligence

[7,13,24]. Sales can be regarded as a time series. However, time series approaches have some limitations for sales forecasting. For instance, to study seasonality, historical data for large time period are required. But it frequently happens that there are no historical data for a target variable, e.g. when a new product is launched. Furthermore, we must take into consideration many exogenous factors influencing sales. We can consider time series forecasting as a regression problem in many cases, especially for sales time series. In [16], we considered linear models, machine learning and probabilistic models for time series modeling. In [15], we regarded the logistic regression with Bayesian inference for analysing manufacturing failures. In [17], we study the use of machine learning models for sales predictive analytics. We researched the main approaches and case studies of implementing machine learning to sales forecasting. The effect of machine learning generalization has been studied. This effect can be used for predicting sales in case of a small number of historical data for specific sales time series in case when a new product or store is launched [17]. A stacking approach for building ensemble of single models using Lasso regression has also been studied. The obtained results show that using stacking techniques, we can improve the efficiency of predictive models for sales time series forecasting [17].

In this paper, we consider the use of Bayesian regression for building time series predictive models and for stacking time series predictive models on the second level of the predictive model which is the ensemble of the models of the first level.

## 2 Bayesian Regression Approach

Probabilistic regression models can be based on Bayesian theorem [4,8,12]. This approach allows us to receive a posterior distribution of model parameters using conditional likelihood and prior distribution. Probabilistic approach is more natural for stochastic variables such as sales time series. The difference between Bayesian approach and conventional Ordinary Least Squares (OLS) method is that in the Bayesian approach, uncertainty comes from parameters of model, as opposed to OLS method where the parameters are constant and uncertainty comes from data. In the Bayesian inference, we can use informative prior distributions which can be set up by an expert. So, the result can be considered as a compromise between historical data and expert opinion. It is important in the cases when we have a small number of historical data. In the Bayesian model, we can consider the target variable with non Gaussian distribution, e.g. Student's t-distribution. Probabilistic approach enables us an ability to receive probability density function for the target variable. Having such function, we can make risk assessment and calculate value at risk (VaR) which is 5% quantile. For solving Bayesian models, the numerical Monte-Carlo methods are used. Gibbs and Hamiltonian sampling are the popular methods of finding posterior distributions for the parameters of probabilistic model [4,8,12].

Bayesian inference makes it possible to do nonlinear regression. If we need to fit trend with saturation, we can use logistic curve model for nonlinear trend

and find its parameters using Bayesian inference. Let us consider the case of nonlinear regression for time series which have the trend with saturation. For modeling we consider sales time series. The model can be described as:

$$\begin{aligned} \log(\text{Sales}) &\sim \mathcal{N}(\mu_{\text{Sales}}, \sigma^2) \\ \mu_{\text{Sales}} &= \frac{a}{1 + \exp(bt + c)} + \beta_{\text{Promo}}\text{Promo} + \\ &\beta_{\text{Time}}\text{Time} + \sum_j \beta_j^{wd}\text{WeekDay}_j, \end{aligned} \quad (1)$$

where  $\text{WeekDay}_j$  - are binary variables, which is 1 in the case of sales on an appropriate day of week and 0 otherwise. We can also build hierarchical models using Bayesian inference. In this approach, one set of parameters is the same for all data samples, other parameters can be different for different groups of data samples. In the case of sales data, we can consider trends, promo impact and seasonality as the same for all stores. But the impact of a specific store on sales can be described by an intersect parameter, so this coefficient for this variable will be different for different stores. The hierarchical model can be described as:

$$\begin{aligned} \text{Sales} &\sim \mathcal{N}(\mu_{\text{Sales}}, \sigma^2) \\ \mu_{\text{Sales}} &= \alpha(\text{Store}) + \beta_{\text{Promo}}\text{Promo} + \\ &\beta_{\text{Time}}\text{Time} + \sum_j \beta_j^{wd}\text{WeekDay}_j, \end{aligned} \quad (2)$$

where intersect parameters  $\alpha(\text{Store})$  are different for different stores.

Predictive models can be combined into ensemble model using stacking approach [6, 9, 18–20, 23]. In this approach, prediction results of predictive models on the validation set are treated as covariates for stacking regression. These predictive models are considered as a first level of predictive model ensemble. Stacking model forms the second level of model ensemble. Using Bayesian inference for stacking regression gives distributions for stacking regression coefficients. It enables us to estimate the uncertainty of the first level predictive models. As predictive models for first level of ensemble, we used the following models: 'ARIMA', 'ExtraTree', 'RandomForest', 'Lasso', 'NeuralNetowrk'. The use of these models for stacking by Lasso regression was described in [17]. For stacking, we have chosen the robust regression with Student's t-distribution for the target variable as

$$y \sim \text{Student}_t(\nu, \mu, \sigma) \quad (3)$$

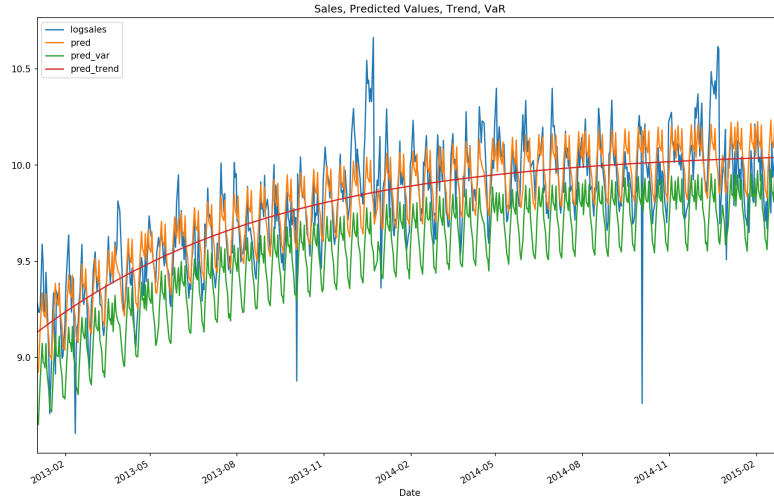
where

$$\mu = \alpha + \sum_i \beta_i x_i, \quad (4)$$

$\nu$  is a distribution parameter, called as degrees of freedom,  $i$  is an index of the predictive model in the stacking regression,  $i \in \{ \text{'ARIMA'}, \text{'ExtraTree'}, \text{'RandomForest'}, \text{'Lasso'}, \text{'NeuralNetowrk'} \}$ .

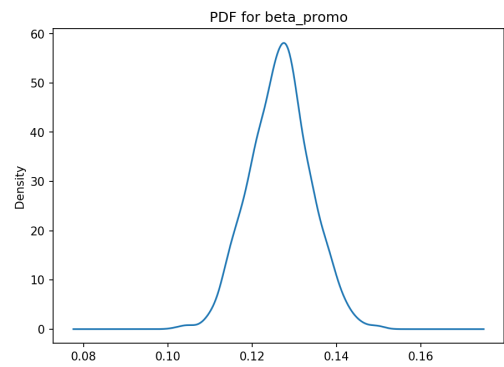
### 3 Numerical Modeling

The data for our analysis are based on store sales historical data from the “Rossmann Store Sales” Kaggle competition [1]. For Bayesian regression, we used Stan platform for statistical modeling [4]. The analysis was conducted in Jupyter Notebook environment using Python programming language and the following main Python packages *pandas*, *sklearn*, *pystan*, *numpy*, *scipy*, *statsmodels*, *keras*, *matplotlib*, *seaborn*. For numerical analysis we modeled time series with multiplicative trend with saturation. Figure 1 shows results of this modeling where mean values and Value at Risk (VaR) characteristics are given. VaR was calculated as 5% percentile. Figure 2 shows probability density function of regression coefficients for Promo factor. Figure 3 shows box plots for probability density function of seasonality coefficients.

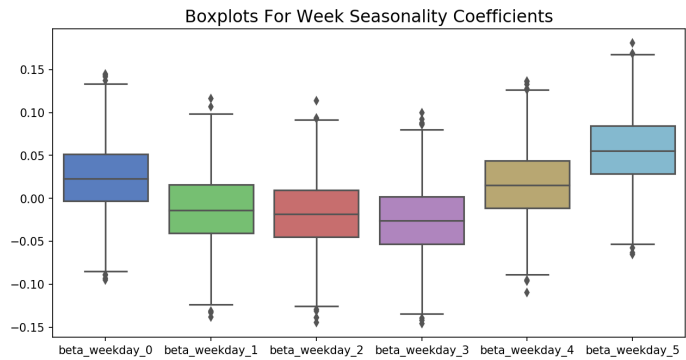


**Fig. 1.** Time series and trend forecasting

Let us consider Bayesian hierarchical model for time series. Intercept parameters  $\alpha(Store)$  in the model (2) are different for different stores. We have considered a case with five different stores. Figure 4 shows the boxplots for the probability density functions of intercept parameters. The dispersion of these distributions describes an uncertainty of influence of specified store on sales. Hierarchical approach makes it possible to use such a model in the case with short historical data for specific stores, e.g. in the case of new stores. Figure 5 shows the results of the forecasting on the validation set in the two cases, the first case is when we use 2 year historical data and the second case is with historical data for 5 days. We can see that such short historical data allow us to estimate sales dynamics correctly. Figure 6 shows box plots for probability density function

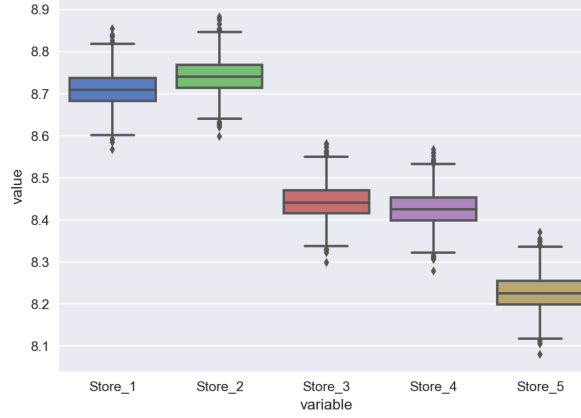


**Fig. 2.** Probability density function of regression coefficients for Promo factor



**Fig. 3.** Box plots for probability density function of seasonality coefficients

of intersect parameters of different time series in case of short time period of historical data for a specified store *Store\_4*. The obtained results show that the dispersion for a specific store with short historical data becomes larger due to uncertainty caused by very short historical data for the specified store.

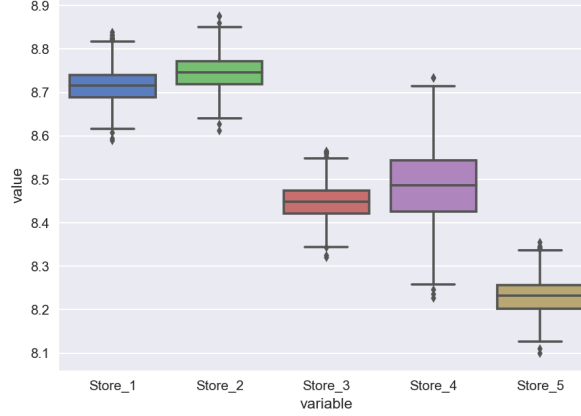


**Fig. 4.** Boxplots for probability density functions of intersect parameters



**Fig. 5.** Forecasting on the validation set for specified time series in cases with different size of historical data

Let us consider the results of Bayesian regression approach for stacking predictive models. We trained different predictive models and made the predictions on the validation set. The ARIMA model was evaluated using *statsmodels* package, Neural Network was evaluated using *keras* package, Random Forest and



**Fig. 6.** Box plots for probability density function of intersect parameters of different time series in case of short time period of historical data for a specified store *Store\_4*.

Extra Tree was evaluated using *sklearn* package. In these calculations, we used the approaches described in [17]. Figure 7 shows the time series forecasts on the validation sets obtained using different models.

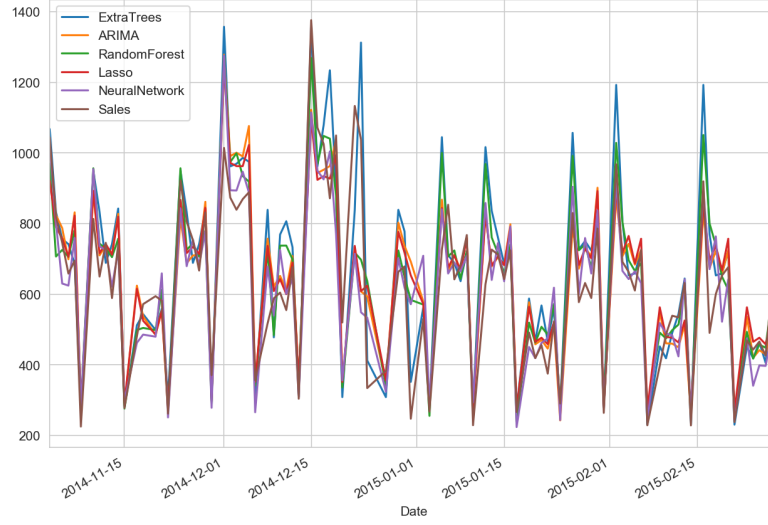
The results of prediction of these models on the validation sets are considered as the covariates for the regression on the second stacking level of the ensemble of models. For stacking predictive models we split the validation set on the training and testing sets. For stacking regression, we normalized the covariates and target variable using z-scores:

$$z_i = \frac{x_i - \mu_i}{\sigma_i}, \quad (5)$$

where  $\mu_i$  is the mean value,  $\sigma_i$  is the standard deviation. The prior distributions for parameters  $\alpha, \beta, \sigma$  in the Bayesian regression model (3)-(4) are considered as Gaussian with mean values equal to 0, and standard deviation equal to 1. We split the validation set on the training and testing sets by time factor. The parameters for prior distributions can be adjusted using prediction scores on testing sets or using expert opinions in the case of small data amount. To estimate uncertainty of regression coefficients, we used the coefficient of variation which is defined as a ratio between the standard deviation and mean value for model coefficient distributions:

$$v_i = \frac{\sigma_i}{\mu_i}, \quad (6)$$

where  $v_i$  is the coefficient of variation,  $\sigma_i$  is a standard deviation,  $\mu_i$  is the mean value for the distribution of the regression coefficient of the model  $i$ . Taking into account that  $\mu_i$  can be negative, we will analyze the absolute value of the coefficient of variation  $|v_i|$ . For the results evaluations, we used a relative mean absolute error (RMAE) and root mean square error (RMSE). Relative mean absolute error (RMAE) was considered as a ratio between the mean absolute



**Fig. 7.** Forecasting of different models on the validation set

error (MAE) and mean values of target variable:

$$RMAE = \frac{E(|y_{pred} - y|)}{E(y)} 100\% \quad (7)$$

Root mean square error (RMSE) was considered as:

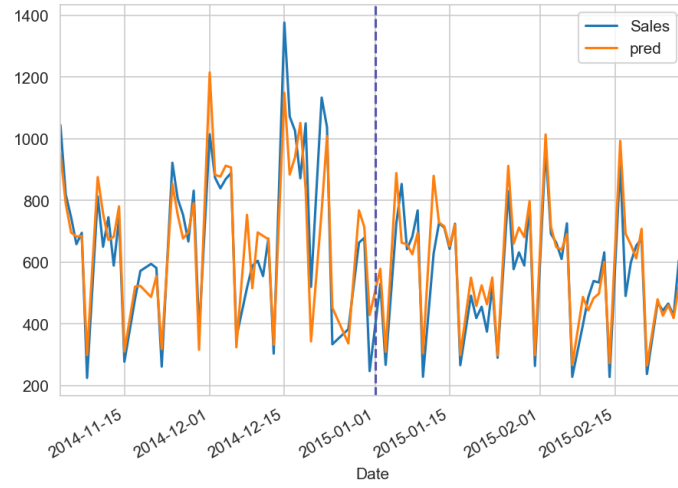
$$RMSE = \sqrt{\frac{\sum_i^n (y_{pred} - y)^2}{n}} \quad (8)$$

The data with predictions of different models on the validation set were split on the training set (48 samples) and testing set (50 samples) by date. We used the robust regression with Student's t-distribution for the target variable. As a result of calculations, we received the following scores:  $RMAE(\text{train})=12.4\%$ ,  $RMAE(\text{test})=9.8\%$ ,  $RMSE(\text{train})=113.7$ ,  $RMSE(\text{test})=74.7$ . Figure 8 shows mean values time series for real and forecasted sales on the validation and testing sets. The vertical dotted line separates the training and testing sets. Figure 9 shows the probability density function (PDF) for the intersect parameter. One can observe a positive bias of this (PDF). It is caused by the fact that we applied machine learning algorithms to nonstationary time series. If a nonstationary trend is small, it can be compensated on the validation set using stacking regression.

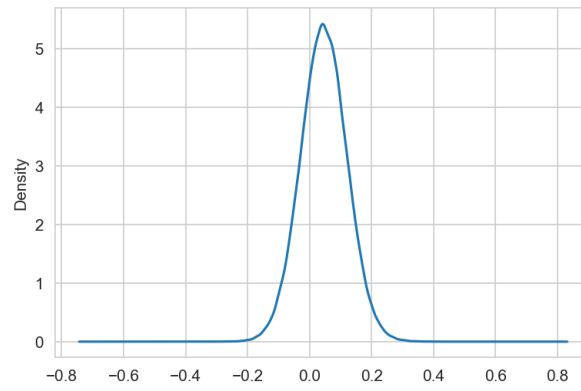
Figure 10 shows the box plots for the PDF of coefficients of models. Figure 11 shows the coefficient of variation for the PDF of regression coefficients of models.

We considered the case with the restraints to regression coefficient of models that they should be positive. We received the similar results:  $RMAE(\text{train})=12.9\%$ ,

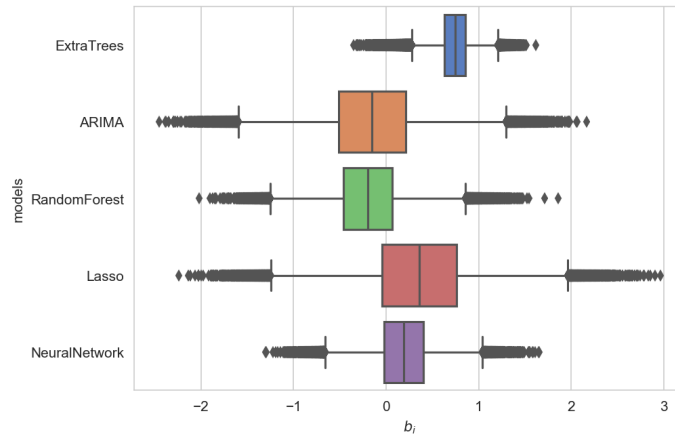




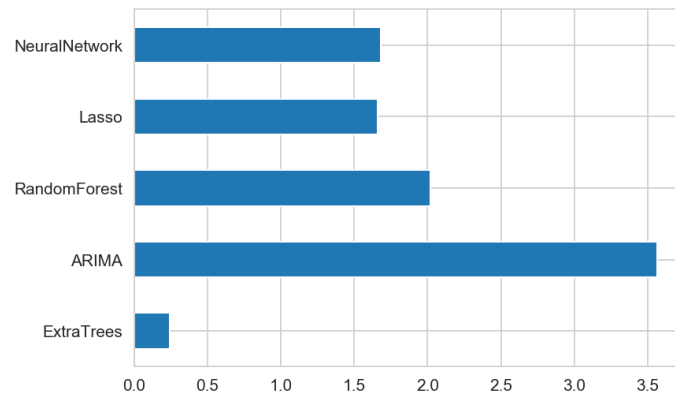
**Fig. 8.** Time series of mean values for real and forecasted sales on validation and testing sets



**Fig. 9.** The PDF for intersect parameter of stacking regression

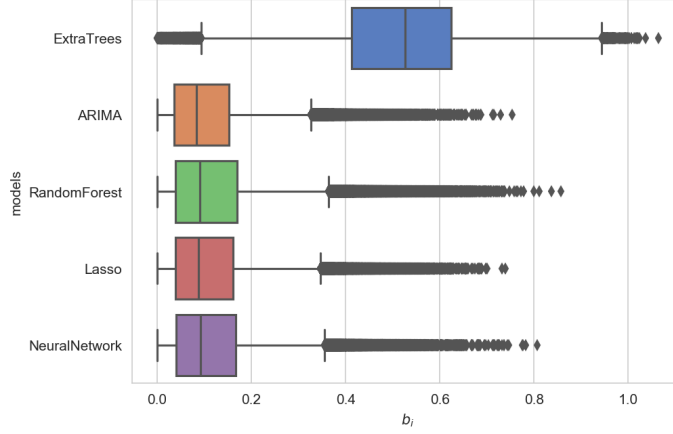


**Fig. 10.** Box plots for the PDF of regression coefficients of models



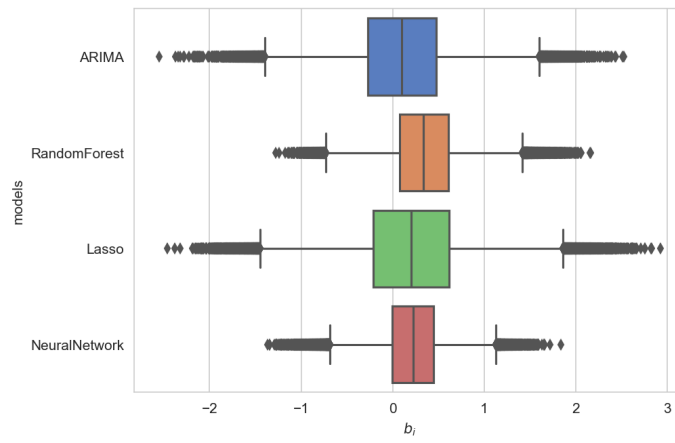
**Fig. 11.** Absolute values of the coefficient of variation for the PDF of regression coefficients of models

RMAE(test)=9.7%, RMSE(train)=117.3, RMSE(test)=76.1. Figure 12 shows the box plots for the PDF of model regression coefficients for this case.

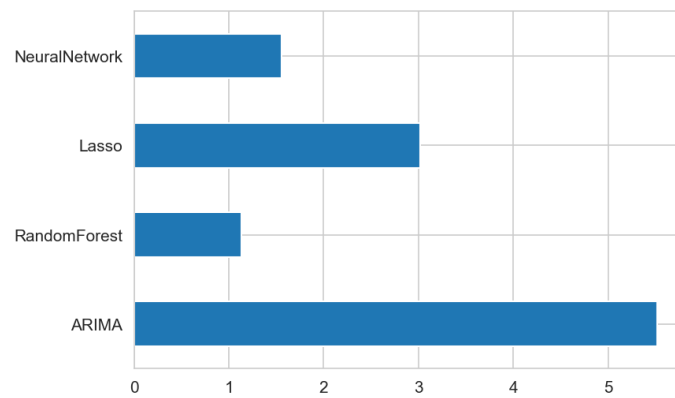


**Fig. 12.** Box plots for the PDF of regression coefficients of models

All models have a similar mean value and variation coefficients. We can observe that errors characteristics RMAE and RMSE on the validation set can be similar with respect to these errors on the training set. It tells us about the fact that Bayesian regression does not overfit on the training set comparing to the machine learning algorithms which can demonstrate essential overfitting on training sets, especially in the cases of small amount of training data. We have chosen the best ExtraTree stacking model and conducted Bayesian regression with this one model only. We received the following scores: RMAE(train)=12.9%, RMAE(test)=11.1%, RMSE(train)=117.1, RMSE(test)=84.7. We also tried to exclude the best model ExtraTree from the stacking regression and conducted Bayesian regression with the rest of models without ExtraTree. In this case we received the following scores: RMAE(train)=14.1%, RMAE(test)=10.2%, RMSE(train)=139.1, RMSE(test)=75.3. Figure 13 shows the box plots for the PDF of model regression coefficients, figure 14 shows the coefficient of variation for the PDF of regression coefficients of models for this case study. We received worse results on the testing set. At the same time these models have the similar influence and thus they can potentially provide more stable results in the future due to possible changing of the quality of features. Noisy models can decrease accuracy on large training data sets, at the same time they contribute to sufficient results in the case of small data sets. We considered the case with a small number of training data, 12 samples. To get stable results, we fixed the  $\nu$  parameter of Student's t-distribution in Bayesian regression model (3)-(4) equal to 10. We received the following scores: RMAE(train)=5.0%, RMAE(test)=14.2%, RMSE(train)=37.5, RMSE(test)=121.3. Figure 15 shows mean values time se-

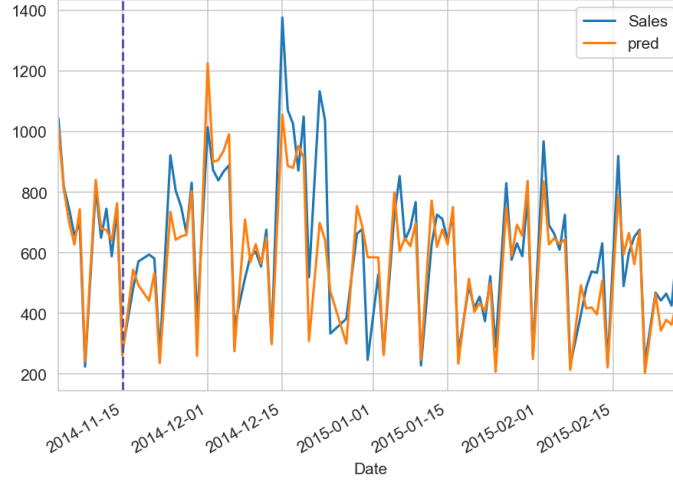


**Fig. 13.** Box plots for the PDF of regression coefficients of models



**Fig. 14.** Absolute values of the coefficient of variation for the PDF of models regression coefficients

ries for real and forecasted sales on the validation and testing sets. Figure 16 shows the box plots for the PDF of regression coefficients of models. Figure 17 shows the coefficient of variation for the PDF of regression coefficients of models. In this case, we can see that an other model starts playing an important role comparing with the previous cases and ExtraTree model does not dominate. The obtained results show that optimizing informative prior distributions of

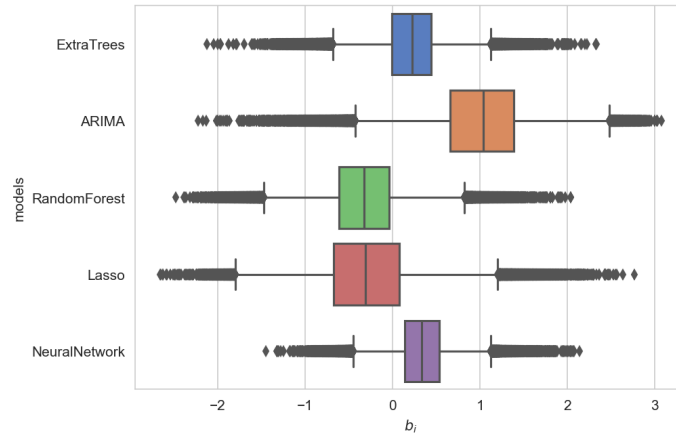


**Fig. 15.** Mean values time series for real and forecasted sales on the validation and testing sets in the case of small training set

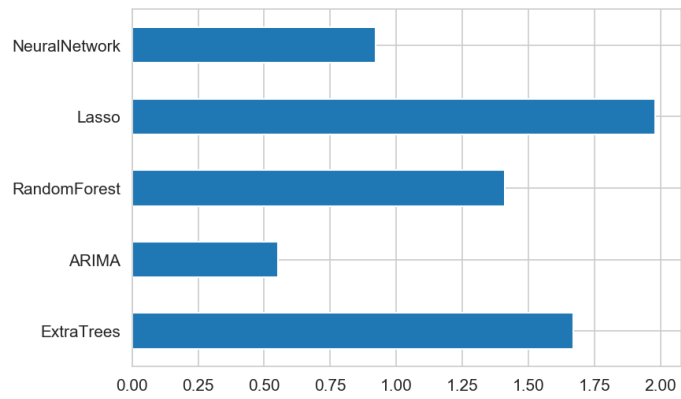
stacking model parameters can improve the scores of prediction results on the testing set.

## 4 Conclusion

Bayesian regression approach for time series modeling with nonlinear trend was analyzed. Such approach allows us to estimate an uncertainty of time series prediction and calculate value at risk (VaR) characteristics. Hierarchical model for time series using Bayesian regression has been considered. In this approach, one set of parameters is the same for all data samples, other parameters can be different for different groups of data samples. This approach makes it possible to use such a model in the case with short historical data for specified time series, e.g. in the case of new stores or new products in the sales prediction problem. The use of Bayesian inference for time series predictive models stacking has been analyzed. A two-level ensemble of the predictive models for time series was considered. The models ARIMA, Neural Network, Random Forest, Extra Tree were used for the prediction on the first level of ensemble of models. On the



**Fig. 16.** Box plots for the PDF of regression coefficients of models in the case of small training set



**Fig. 17.** Absolute values of the coefficient of variation for the PDF of regression coefficients of models in the case of small training set

second stacking level, time series predictions of these models on the validation set were conducted by Bayesian regression. Such an approach gives distributions for regression coefficients of these models. It makes it possible to estimate the uncertainty contributed by each model to the stacking result. The information about these distributions allows us to select an optimal set of stacking models, taking into account domain knowledge. Probabilistic approach for stacking predictive models allows us to make risk assessment for the predictions that is important in a decision-making process. Noisy models can decrease accuracy on large training data sets, at the same time they contribute to sufficient results in the case of small data sets. Using Bayesian inference for stacking regression can be useful in cases of small datasets and help experts to select a set of models for stacking as well as make assessments of different kinds of risks. Choosing the final models for stacking is up to an expert who takes into account different factors such as uncertainty of each model on the stacking regression level, amount of training and testing data, the stability of models. In Bayesian regression, we can receive a quantitative measure for the uncertainty that can be a very useful information for experts in model selection and stacking. An expert can also set up informative prior distributions for stacking regression coefficients of models, taking into account the domain knowledge information. So, Bayesian approach for stacking regression can give us the information about uncertainty of predictive models. Using this information and domain knowledge, an expert can select models to get stable stacking ensemble of predictive models.

## References

1. Rossmann Store Sales. Forecast sales using store, promotion, and competitor data. Kaggle.Com. Available online: <http://www.kaggle.com/c/rossmann-store-sales>
2. Box, G.E., Jenkins, G.M., Reinsel, G.C., Ljung, G.M.: Time series analysis: forecasting and control. John Wiley & Sons (2015). <https://doi.org/10.1111/jtsa.12194>
3. Brockwell, P.J., Davis, R.A., Calder, M.V.: Introduction to time series and forecasting, vol. 2. Springer (2002)
4. Carpenter, B., Gelman, A., Hoffman, M.D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., Riddell, A.: Stan: A probabilistic programming language. *Journal of statistical software* **76**(1) (2017). <https://doi.org/10.18637/jss.v076.i01>
5. Chatfield, C.: Time-series forecasting. Chapman and Hall/CRC (2000). <https://doi.org/10.1201/9781420036206>
6. Dietterich, T.G.: Ensemble methods in machine learning. In: International workshop on multiple classifier systems. pp. 1–15. Springer (2000). [https://doi.org/10.1007/3-540-45014-9\\_1](https://doi.org/10.1007/3-540-45014-9_1)
7. Efendigil, T., Önüt, S., Kahraman, C.: A decision support system for demand forecasting with artificial neural networks and neuro-fuzzy models: A comparative analysis. *Expert Systems with Applications* **36**(3), 6697–6707 (2009). <https://doi.org/10.1016/j.eswa.2008.08.058>
8. Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A., Rubin, D.B.: Bayesian data analysis. Chapman and Hall/CRC (2013). <https://doi.org/10.1201/b16018>

9. Gomes, H.M., Barddal, J.P., Enembreck, F., Bifet, A.: A survey on ensemble learning for data stream classification. *ACM Computing Surveys (CSUR)* **50**(2), 1–36 (2017). <https://doi.org/10.1145/3054925>
10. Hyndman, R.J., Athanasopoulos, G.: *Forecasting: principles and practice*. OTexts (2018)
11. Hyndman, R.J., Khandakar, Y., et al.: *Automatic time series for forecasting: the forecast package for R*. No. 6/07, Monash University, Department of Econometrics and Business Statistics (2007)
12. Kruschke, J.: *Doing Bayesian data analysis: A tutorial with R, JAGS, and Stan*. Academic Press (2014)
13. Mentzer, J.T., Moon, M.A.: *Sales forecasting management: a demand management approach*. Sage (2005). <https://doi.org/10.4135/9781452204444>
14. Papacharalampous, G., Tyralis, H., Koutsoyiannis, D.: Univariate time series forecasting of temperature and precipitation with a focus on machine learning algorithms: A multiple-case study from greece. *Water Resources Management* **32**(15), 5207–5239 (2018). <https://doi.org/10.1007/s11269-018-2155-6>
15. Pavlyshenko, B.: Machine learning, linear and bayesian models for logistic regression in failure detection problems. In: *Big Data (Big Data), 2016 IEEE International Conference on*. pp. 2046–2050. IEEE (2016). <https://doi.org/10.1109/bigdata.2016.7840828>
16. Pavlyshenko, B.M.: Linear, machine learning and probabilistic approaches for time series analysis. In: *Data Stream Mining & Processing (DSMP), IEEE First International Conference on*. pp. 377–381. IEEE (2016). <https://doi.org/10.1109/dsmp.2016.7583582>
17. Pavlyshenko, B.M.: Machine-learning models for sales time series forecasting. *Data* **4**(1), 15 (2019). <https://doi.org/10.3390/data4010015>
18. Rokach, L.: Ensemble methods for classifiers. In: *Data mining and knowledge discovery handbook*, pp. 957–980. Springer (2005). [https://doi.org/10.1007/0-387-25465-x\\_45](https://doi.org/10.1007/0-387-25465-x_45)
19. Rokach, L.: Ensemble-based classifiers. *Artificial Intelligence Review* **33**(1-2), 1–39 (2010). <https://doi.org/10.1007/s10462-009-9124-7>
20. Sagi, O., Rokach, L.: Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* **8**(4), e1249 (2018). <https://doi.org/10.1002/widm.1249>
21. Tsay, R.S.: *Analysis of financial time series*, vol. 543. John Wiley & Sons (2005). <https://doi.org/10.1002/0471746193>
22. Wei, W.W.: Time series analysis. In: *The Oxford Handbook of Quantitative Methods in Psychology: Vol. 2* (2006)
23. Wolpert, D.H.: Stacked generalization. *Neural networks* **5**(2), 241–259 (1992). [https://doi.org/10.1016/s0893-6080\(05\)80023-1](https://doi.org/10.1016/s0893-6080(05)80023-1)
24. Zhang, G.P.: *Neural networks in business forecasting*. IGI Global (2004). <https://doi.org/10.4018/978-1-59140-176-6>