A Comparative Case Study on Time Series Prediction

Anıl Ozdemir
Faculty of Computer Science and Engineering
Sabancı University, İstanbul, Turkey
aozdemir@sabanciuniv.edu

Furkan Coskun
Faculty of Computer Science and Engineering
Sabancı University, İstanbul, Turkey
furkancoskun@sabanciuniv.edu

Selim Balcisoy
Faculty of Computer Science and Engineering
Sabancı University, İstanbul, Turkey
balcisoy@sabanciuniv.edu

***Abstract*— A time series is a sequence collected at consecutive equally spaced points in time. The basic idea behind the time series forecasting is the use of a model to estimate future values based on previously observed ones. Traditionally, statistical methods are used to forecasting time series however, Machine Learning (ML) algorithms have been also proposed as alternatives to statistical methods in past decades. In this paper, we evaluate forecasting performance of different ML algorithms and statistical methods on Turkey automobile sales. Recently, various of work has claimed that traditional statistical methods dominate the ML solutions in terms of time series forecasting. This study discusses different aspects of ML and statistical methods and compare their performance on different time series.**

Keywords—Time series forecasting, machine learning regression, statistical models.

I. INTRODUCTION

Time series forecasting has been always a challenging problem that pushes researchers to find better models. The main purpose of time series modeling is to collect and observe the past observations to estimate future values. Time series forecasting has attracted many researchers in last few decades since the importance in various fields such as economics, business, finance, science and engineering [1]. As a result, various important time series forecasting models have been proposed in literature. In early dates, it started with the exponential smoothing for dealing with inventory control [2]. Afterwards, autoregressive–moving-average (ARMA) has been described in the early 1950's by Peter Whittle [3] and became popular in 1970 by George E. P. Box and Gwilym Jenkins [4]. This leads to introduce the Box-Jenkings methodology to Autoregressive Integrated Moving Average (ARIMA) which is one of the most frequently used models in time series. Eventually, multivariate GARCH models were also proposed and enlarged this field. [7-8]

On the other hand, Machine Learning (ML) methods have been appeared in the academic literature as alternatives to statistical method to improve time series estimations [9]. Many of works propose new ML models to advancing in the field of forecasting [10-12]. However, there are several of limitations have been mentioned before in terms of ML models on time series forecasting. The one example is outcomes of these models base on a few time series which leads to decrease their generalization. Another is the models are interpreted for short-term forecasting horizons, usually one-step-ahead [8] [12].

There are several studies exist for comparing ML and statistical methods to time series forecasting. The famous one is Makridakis Competitions (M- Competitions) which is series of open competitions organized by teams led by forecasting researcher Spyros Makridakis to evaluate and compare the performance of different forecasting methods [14] [15] [16].

In this study, we benefited of M-3 Competition which based on analysis of 24 methods and 3003 time series [14]. The data used in this study is monthly automobile sales of Turkey which consist of six different Time series in the years of 2004 to 2018. We used six Machine learning algorithms, which are Random Forest (RF), Decision Tree (CART), Multi-Layer Perceptron (MLP), Bayesian Ridge Regression (BRR), Support Vector Machines (SVR) and K- Nearest Neighbor (KNN), and four statistical methods which are Linear Regression (LR), Logistic Regression (LOGR), Triple Exponential Smoothing (Holts-Winters) and Autoregressive Integrated Moving Average (ARIMA).The observations from 146 months are used in these models to estimate rest of 24 months and only the one-step-ahead forecasting is considered. Also, two accuracy metrics are performed to understand compare the accuracy of different models: The symmetric Mean Absolute Percentage Error (sMAPE) and the Root Mean Squared Error (RMSE).

Results demonstrated that average scores from statistical models works better than ML methods. Only the RF is slightly close to the ARIMA and Holts-Winters. Also, we conclude that RMSE and SMAPE scores are varies in some cases. To be clearer, even if sMAPE shows successful results, RMSE score showed the different results as a complement in some cases. Outcomes from our results also supports the conclusions from the M-3 Competitions. Recent works from M-3 competitions stated that statistically complicated models are do not essentially produce more accurate predictions than simpler models, and the rankings of performance of models could be change depending on the accuracy measure used [14].

From the conclusions we made, neither ML nor statistical methods dominates each other. Even though statistical techniques are more prominent in the results, estimation of ML is very strong as well and even more successful in estimating outlier points.

II. MATERIALS & METHODS

A. Data Preprocessing

One method we applied to clean raw data is segmentation. The segmentation was performed according to the Euro Car Segment Protocol (ECSP) and six segments appeared in our data [17]. These segments are A, B, C, D, E and F which means mini cars, small cars, medium cars, large cars, executive cars and luxury cars, respectively. Every car segment corresponds different time series. Thus, we have 6 different time series data. Data cleaning was made by replace the null values with corresponding averages of features. Seasonality and trends can also be observed in our data by examined local maximums in the first period of every year.In our case, normalization has been used for all models, also deasonalization used in some cases in ARIMA.

B. Feature Engineering

ML methods requires more extra steps than statistical methods due to lack of information in data (# of inputs in neural networks) [18]. In other words, we expanded our data with combined with additional data sources for only ML models. These are dollars and euros to Turkish lira exchange rate, the

total value of Turkey trade in goods (export and import), and Turkey stock market index for every month.

Furthermore, new parameters created from the parameters that was already available, which are the previous values of features. Main reason extraction is to determine which parameter impacts the sales numbers of an automobiles in a month. After we made the improvements mentioned above, the number of features in our data goes from 6 to 85.

C. Measure of Accuracy

Two different accuracy measures are performed in this study: The symmetric Mean Absolute Percentage Error (sMAPE) and the Root Mean Squared Error (RMSE). The sMAPE is defined as follows:

$$SMAPE = \frac{100\%}{n} \sum_{t=1}^{n} \frac{| F_t - A_t |}{| A_t | + | F_t |}$$

where A_t is the actual value and F_t is the forecast value. As sMAPE score decreases, model is become more successful.

The root-mean-square error (RMSE) is used to measure of the differences between values predicted by a model (estimator) and the values observed. As RMSE score decreases, model is become more successful. The RMSE is defined as follows: [19] [20]

$$RMSE = \sqrt{\frac{\sum_{t=1}^{T} \hat{(y_t - y_t)^2}}{T}}$$

D. Applying ML Models

2.d.1 Bayesian Ridge Regression (BRR)

BRR is sub method of traditional Bayesian regression and described as follows:

$$P(w|\lambda) = N(w|0, \lambda^{-1} I_p)$$

The regularization parameter used in Ridge Regression is equal to finding a maximum a posterior probability estimation under a Gaussian before over the parameters with precision. Instead of setting lambda manually, to treat it as a random variable to be estimated from the data. The BRR method is constructed using BayesianRidge function of scikit-learn v0.20.2 Python package. [21] [22]

2.d.2 Decision Tree Regression (CART)

CART uses a decision tree as a predictive model to make observations about an item (Corresponding month of a year in our study) to conclusions about the item's target value (# of sales in our study). Target variable can take either discrete set and continuous values (Regressors). The purity measure we used in building decision tree is Gini Index. As a tree growing, model decides on which parameters or features to select to reach specific output. We increased the success rate of a tree by pruning which is the way that removing the parameters that having low importance. By doing this, we decreased the complexity and increased its forecast power. The CART method is constructed exploiting DecisionTreeRegressor function of scikit-learn v0.20.2 Python package. [22] [23]

2.d.3 K-Nearest Neighbor Regression (KNN)

KNN is a non-parametric method for both regressions and classifications. Its estimations based on a similarity measures, such as the Euclidean distance, Manhattan distance. In our study, Euclidean distance used for both training and testing the method. In this way, given the N inputs, KNN picks the closest K training data points and sets the prediction as the average of the target output values for these points. The KNN method is constructed using KNeighborsRegressor function of scikit-learn v0.20.2 Python package. [22] [24].

2.d.4 Support Vector Regression (SVR)

SVR is the regression method performed by a traditional Support Vector Machine which is also used as a regression model. Basically, it uses the same principle as the SVM for classification which is gradually identify the decision boundary by maximizes the margin between different classes and minimize the error under specific tolerance. Since output is a real number in SVR, a margin of tolerance is set in approximation to the SVM. The SVR method is constructed using svm.SVR function of scikit-learn v0.20.2 Python package. [22] [25].

2.d.5 Random Forest Regression (RF)

RF is a method for both classifications and regressions based on ensemble learning which aims improve generalizability and reduce the error. It combines the predictions of several base estimators built with a given learning algorithm. In our random forests, each tree in the ensemble is constructed from a sample drawn with bootstrapping from the training set. Thanks to the randomness, the bias of the forest usually lightly increments with compare to the bias of a single non-random tree. Since its variance also decreases due to averaging, this leads to results a better model in overall. The RFS method is constructed exploiting RandomForestRegressor function of scikit-learn v0.20.2 Python package. [22] [26].

2.d.6 Multi-Layer Perceptron (MLP).

MLP is one of the fully-connected feedforward neural network which can be used as a method for both regressions and classifications. The network consists of at least one hidden layer with one input and output layer. The calculation of output based on the weighted sum of inputs and the activation function. In our study, a single hidden layer with 2N-1 nodes and hyperbolic tangent activation function are used to construct a neural network for both training and testing the method. For the weight optimization, L-BFGS solver is used which outperforms Stochastic Gradient Descent (SGD) in smaller datasets. The MLP method is constructed using MLPRegressor method of scikit-learn v0.20.2 Python package. [22] [31].

E.Applying Statistical Models

In our study, four traditional statistical model used for forecasting as follows:

2.e.1 Linear Regression (LR)

Linear Regression fits a linear model with coefficients to minimize the remaining sum of squares between the observed values in the dataset. Also, linear approximation used to predict values.

Basically, it tries to solve a problem of the following form:

$$\min_{w} \| Xw - y \|_2^2$$

The LR method is constructed using LinearRegression function of scikit-learn v0.20.2 Python package. [22] [27].

2.e.2 Logistic Regression (LogR)

LogR, is a linear model for classification compare with regression. Its mechanism based on using a logistic function to model a binary dependent variable. Estimated values based on logistic distribution function and the estimated values are probabilities which restricted to (0,1). Therefore, LogR predicts the probability of particular outcomes rather than the outcomes themselves.The LogR method is constructed using LogisticRegression function of scikit-learn v0.20.2 Python package. [22] [32].

2.e.3 Triple Exponential Smoothing (Holt-Winters' Method)

Exponential smoothing is a method for smoothing time series data by using the exponential window function, which is a mathematical function with symmetric, local maximum at the middle and zero-valued outside of chosen interval in signal processing and statistics [28]. In in the simple moving average, the past observations are weighted equally while exponential functions are used to assign exponentially decreasing weights over time. The simplest exponential smoothing formulas described as follows:

$$\begin{aligned} s_0 &= x_0 \\ s_t &= \alpha x_t + (1-\alpha)s_{t-1}, t > 0 \end{aligned}$$

Where the sequence of observations starts at time t=0, α is the smoothing factor and 0< α <1. Holt-Winters' method applies exponential smoothing three times. These are three frequency signals which are value, trend and seasonality to be removed in time series. The model predicts a current and future value by calculating the joined impact of these three signals. The model requires three parameters (α, β, γ) which corresponds the smoothing, length of a season and # of periods in a season. In case of seasonality of automobile sales data of Turkey, full-repetition occur at yearly level since there is a peak at every January of corresponding year. This means our season is a year. Triple exponential smoothing with multiplicative seasonality is given by the formulas:

$$\begin{aligned} s_0 &= x_0 \\ s_t &= \alpha\frac{x_t}{c_{t-L}} + (1-\alpha)(s_{t-1}+b_{t-1}) \\ c_t &= \gamma\frac{x_t}{s_t} + (1-\gamma)c_{t-L} \\ F_{t+m} &= (s_t + mb_t)c_{t_L+1+(m-1)modL}, \\ whe & re 0 < \alpha < 1, 0 < \beta < 1, and 0 < \gamma < 1 \end{aligned}$$

The Holt-Winters' method is constructed using holtwinters. ExponentialSmoothing function of statsmodels v0.10.0 Python package. [28] [29].

2.d.4 Autoregressive Integrated Moving Average (ARIMA)

ARIMA is a generalization of auto regressive moving average model (ARMA). In both case, models are fitted to time series for forecasting future values. The model used in this study is Seasonal ARIMA which denoted as follows:

$\text{ARIMA}(p,d,q)(P,D,Q)_m$

where m is the # of periods in each season (12 in our case) and P, D, Q refer to the autoregressive, differencing and moving average terms respectively. Grid search is used for finding optimal ARIMA model hyperparameters (P, D, Q). It is general procedure that tune the hyperparameters for one-step-ahead forecasting. [30] The ARIMA model is constructed using. tsa.arima_model.ARIMA function of statsmodels v0.10.0 Python package. [29].

III. RESULTS

Performance of six ML methods and four statistical methods evaluated on automobile sales numbers of Turkey for each different car segments. Major differences between segments are explained by the total numbers.

The results from 146 months are used to estimate rest of 24 months. Only the one-step-ahead forecasting is considered. Fig 3.1,3.2 and table I show the overall performance of the compared forecasting models on all the six segments with normalized and non-normalized preprocessing. RMSE and SMAPE are used to measure accuracy on all the models.

Fig 3.1: sMAPE Scores on each segment according to the models before Normalization

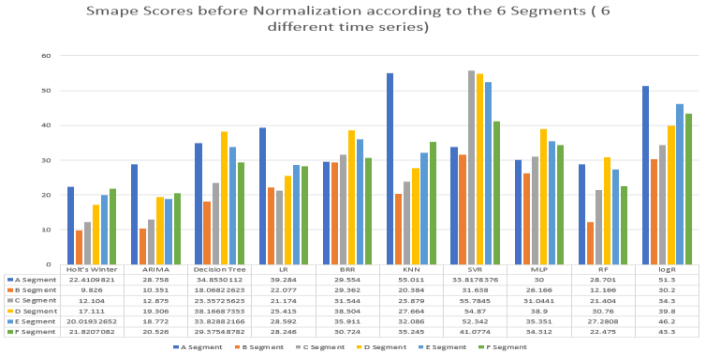

Fig 3.2: sMAPE Scores on each segment according to the models after Normalization

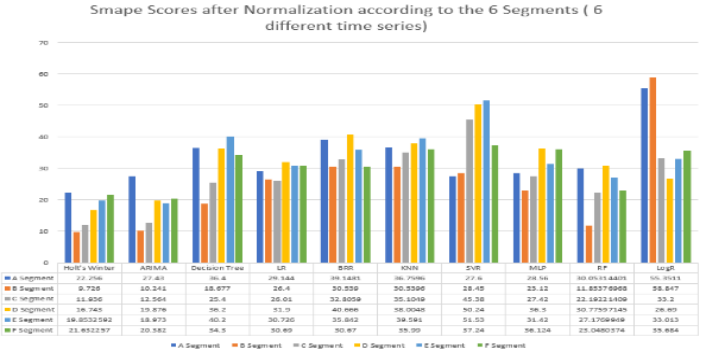

Table I. The Overall Average of the Compared Methods on All Time Series with and without normalization by sMAPE and RMSE scores.

Model	sMAPE(%)	sMAPE-Norm	RMSE	RMSE-Norm
Holts	17.2	17.0	1851.8	1796.8
Arima	18.4	18.2	1839.3	1791.5
RF	23.8	24.1	2538.8	2578.5
LR	27.4	29.1	2863.1	3575.1
CART	29.6	31.8	2765.2	3806.3
KNN	32.3	36.1	3812.1	4135.9
BRR	32.6	34.9	1851.8	4099.2
MLP	32.6	30.4	4758.8	4190.7
SVR	44.9	40.0	5343.7	4519.3
LogR	40.8	40.4	4271.8	6761.1

One can observe from the monitored results of the Table I the following:

- The general rank of the models very close for both normalized and non-normalized preprocessing methods.
- For average scores from sMAPE and RMSE on all segments, statistical models, such as Holts-winters and

ARIMA are at the top while ML models are left behind. Only the Random Forest is slightly close to the ARIMA and Holts. On the other hand, rest of the scores from ML models are very similar except SVR.

- In more than one cases, such as on the B, C and F segments Random forests are even equivalent with ARIMA and Holts (Fig 3.1 and 3.2).
- In case of RMSE, numbers are very different due to means from different segments quite dissimilar. However, in some cases, RMSE scores are very similar with each other (Table I).
- Results demonstrated different models differ significantly and there is an unambiguous ranking even if statistical models very likely to exceed ML models in most cases. We should refer that this ranking applies to market-type time series and it probably will change on another types.
- Preprocessing can have great impact on model's performance, we applied classical normalization in our study, and this leads to difference in all model's performance.

IV. CONCLUSION & FEATURE WORKS

Recently, many of studies proposed that ML models underperforms most noticeably in time series prediction when compare it with traditional statistical models, with taken in to consideration of they are being much more sophisticated and computationally demanding than the statistical models. Also, these works state that, understanding the motive behind their underperformance is the only way to develop ML models [8] [14]. We think that neither ML and statistical methods dominates each other and believe that we could increase our ML model's performance by adding more indicative and informative features for specific time series. When we compare ML and statistics in extraordinary cases, statistical models cannot successful for forecasting due to there lack of indicator. Here extraordinary means unexpected increase or decline in numbers. However, features, such as exchange rates in our study, can inform ML models for possible extreme conditions. In the situation like that, we cannot expect the statistical models can figure out this case because it repeats itself by observed previous experiences. Even that cases like this are not usual, they are very important when turns these cases into real life problems [31]. We believe that, with more additional indicative and informative feature selection, accuracy of ML models would be more successful. In future, we would like the develop current models and add more models to improve our study.

References

[1] Brillinger, David R. Time series: data analysis and theory. Vol. 36. Siam, 1981.

[2] Brown, Robert Goodell. Statistical forecasting for inventory control. McGraw/Hill, 1959.

[3] Whittle, P. (1951). Hypothesis Testing in Time Series Analysis. Almquist and Wicksell. Whittle, P. (1963). Prediction and Regulation. English Universities Press. ISBN 0-8166-1147-5.

[4] Hannan, Edward James (1970). Multiple time series. Wiley series in probability and mathematical statistics. New York: John Wiley and Sons.

[5] Box G, Jenkins G. Time Series Analysis: Forecasting and Control. San Francisco: Holden-Day; 1970.

[6] Brockwell, Peter J., Richard A. Davis, and Matthew V. Calder. Introduction to time series and forecasting. Vol. 2. New York: springer, 2002.

[7] Bauwens L, Laurent S, Rombouts JVK. Multivariate GARCH models: a survey. Journal of Applied Econometrics. 2006;21(1):79–109.

[8] Makridakis, Spyros, Evangelos Spiliotis, and Vassilios Assimakopoulos. "Statistical and Machine Learning forecasting methods: Concerns and ways forward." PloS one 13, no. 3 (2018): e0194889.

[9] Gilchrist, Warren. Statistical forecasting. Vol. 322. London: Wiley, 1976.

[10] Zhang G, Eddy Patuwo B, Hu Y M. Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting. 1998;14(1):35–62.

[11] Kim, Kyoung-jae. "Financial time series forecasting using support vector machines." Neurocomputing 55, no. 1-2 (2003): 307-319.

[12] Bontempi, Gianluca, Souhaib Ben Taieb, and Yann-Aël Le Borgne. "Machine learning strategies for time series forecasting." In European BusinessIntelligence Summer School, pp. 62-77. Springer, Berlin, Heidelberg,2012.

[13] Adya M, Collopy F. How effective are neural networks at forecasting and prediction? A review and evaluation. Journal of Forecasting. 1998;17(56):481–495.

[14] Makridakis, Spyros, and Michele Hibon. "The M3-Competition: results, conclusions and implications." International journal of forecasting 16, no. 4 (2000): 451-476.

[15] Hibon, Michèle, and Herman Stekler. The M-3 Competition: Statistical Tests of the Results. INSEAD, 2003.

[16] Crone SF, Hibon M, Nikolopoulos K. Advances in forecasting with neural networks? Empirical evidence from the NN3 competition on time series prediction. International Journal of Forecasting. 2011;27(3):635–660.

[17] Thiel, Christian, Johannes Schmidt, Arnold Van Zyl, and ESchmid. "Cost and well-to-wheel implications of the vehicle fleet CO2 emission regulation in the European Union." Transportation Research Part A: policy and practice 63 (2014): 25-42

[18] Hansen, James V., James B. McDonald, and Ray D. Nelson. "Time Series Prediction with Genetic-Algorithm Designed Neural Networks: An Empirical Comparison with Modern Statistical Models." Computational Intelligence 15, no. 3 (1999): 171-184.

[19] Goodwin P, Lawton R. On the asymmetry of the symmetric MAPE. International Journal of Forecasting. 1999; 15(4):405–408.

[20] Armstrong, J. Scott; Collopy, Fred (1992). "Error Measures for Generalizing About Forecasting Methods: Empirical Comparisons" (PDF). International Journal of Forecasting. 8 (1): 69–80.

[21] Tipping, Michael E. "Sparse Bayesian learning and the relevance vector machine." Journal of machine learning research 1, no. Jun (2001): 211-244.

[22] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

[23] Rokach, Lior; Maimon, O. (2008). Data mining with decision trees: theory and applications. World Scientific Pub Co Inc. ISBN 978-9812771711.

[24] Altman, N. S. (1992). "An introduction to kernel and nearest-neighbor nonparametric regression". The American Statistician. 46 (3): 175–185.

[25] Scho ¨lkopf B, Smola AJ. Learning with kernel: Support Vector Machines, Regularization, Optimization and Beyond. The MIT Press; 2001.

[26] Breiman, Leo. "Random forests." Machine learning 45, no. 1 (2001): 5-32.

[27] Yan, Xin, and Xiaogang Su. Linear regression analysis: theory and computing. World Scientific, 2009.

[28] Weisstein, Eric W. "Binet-Cauchy identity." CRC concise encyclopedia of mathematics (2nd ed.), CRC Press, ISBN 1584883472 (2003): 228.

[29] Seabold, Skipper, and Josef Perktold. "Statsmodels: Econometric and statistical modeling with python." Proceedings of the 9th Python in Science Conference. 2010.

[30] Hyndman, Rob J., and George Athanasopoulos. "8.9 Seasonal ARIMA models." Forecasting: principles and practice. oTexts. Retrieved 19 (2015).

[31] Pal, Sankar K., and Sushmita Mitra. "Multilayer Perceptron, Fuzzy Sets, Classifiaction." (1992).

[32] Kedem, Benjamin, and Konstantinos Fokianos. Regression models for time series analysis. Vol. 488. John Wiley & Sons, 2005.