Inventory Management for Spare Parts

Students: Berkay Akil – Cansu Ünal – Itır Ege Değer – İzel Özer

Faculty Members: Kamer Kaya – Kemal Kılıç – Sinan Yıldırım

Company Advisor: Barış Genç

Sabancı University – Faculty of Engineering and Natural Sciences

ABSTRACT

The efficient management of the spare parts is crucial for many companies in vehicle sector because it ensures to avoid stock-outs which will boost customer satisfaction and reduce high inventory costs. The aim of this project is to choose the most fitting model to predict the stocks of Doğuş Teknoloji using the last 8 years' masked weekly demands of three types of vehicle spare parts (A, B and C parts).

Recurrent decreases in the quantity were observed periodically and it was decided that national holidays and seasonality may have an effect on the demand. Thus, the data is processed in two different ways; forecasting with and without the holiday effects.

OBJECTIVES

- The company was applying a common model such as moving average to make predictions about the future spare part demand planning. However, other models were investigated to find a more accurate one.
- The reason why we are using 4 different libraries/models is that they all have different approaches to a given problem with different properties.
- After finding suitable models, a fitting model can be created by changing the parameters or combining the existing ones.

PROJECT DETAILS

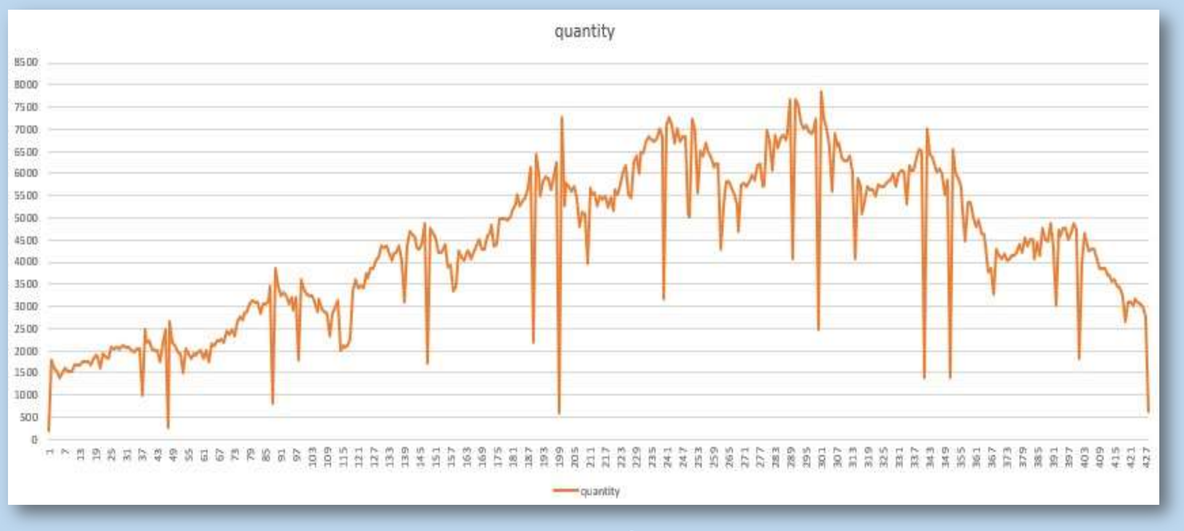

Fig 1. «Quantity» vs «yearweek» graph for Part A (similar graphs can be observed for the other parts (B and C) as well)

The approach to solve this problem is to use the following tools to come up with the most suitable models for predicting the demands that belong to last 43 weeks of the provided data.

Neural Networks: Neural Networks is an information processing approach which is inspired by the biological nervous systems. It is composed of a high number of interconnected processing elements which solve specific problems by learning from examples. [1]

Fig 2. Outcomes of Neural Networks for predictions with and without «holiday»

ARIMA: ARIMA describes a time series as a combination of autoregressive (AR) and moving average (MA) lags which detect the autocorrelation within the time series. Seasonal ARIMA with exogenous variables (SARIMAX) is a linear regression model which extends ARIMA. SARIMAX models rely on seasonal lags and differences to fit the seasonal pattern and let adding external variables to the model by generalizing the regression approach. [2]

Fig 3. Outcomes of ARIMA for predictions with and without «holiday»

Prophet: Prophet is based on an additive model where non-linear trends are fit with yearly and weekly seasonality, plus holidays. It works best with daily periodicity data with at least one year of historical data. It is robust to missing data, shifts in the trend, and large outliers. It uses times series model which has three main components: trend, seasonality and holidays. [3]

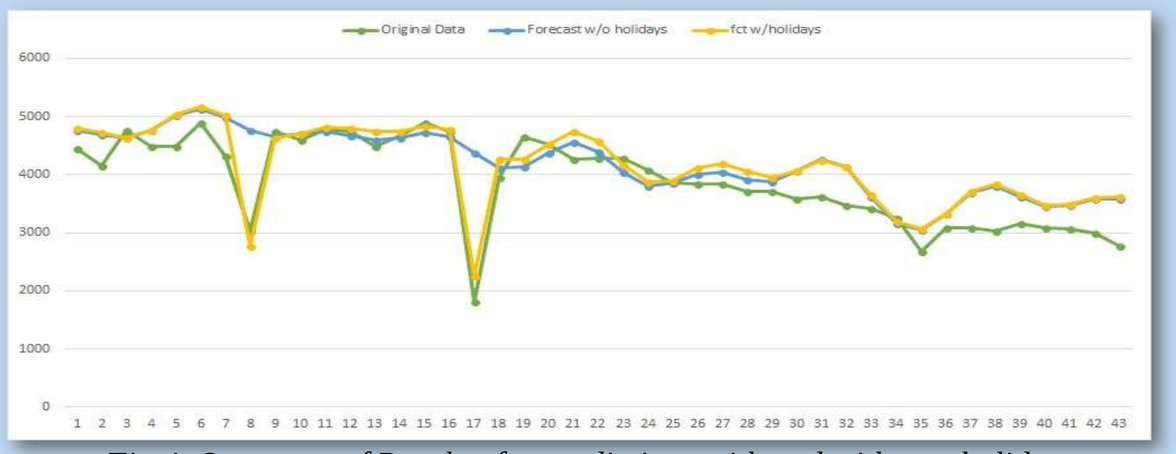

Fig 4. Outcomes of Prophet for predictions with and without «holiday»

SCkit: Sckit-learn (sklearn) is an opensource library used for data mining and data analysis, written in Python and built on NumPy, SciPy and matplotlib. It is capable of running for classification, regression, clustering, dimensionality reduction, model selection and preprocessing. [4]

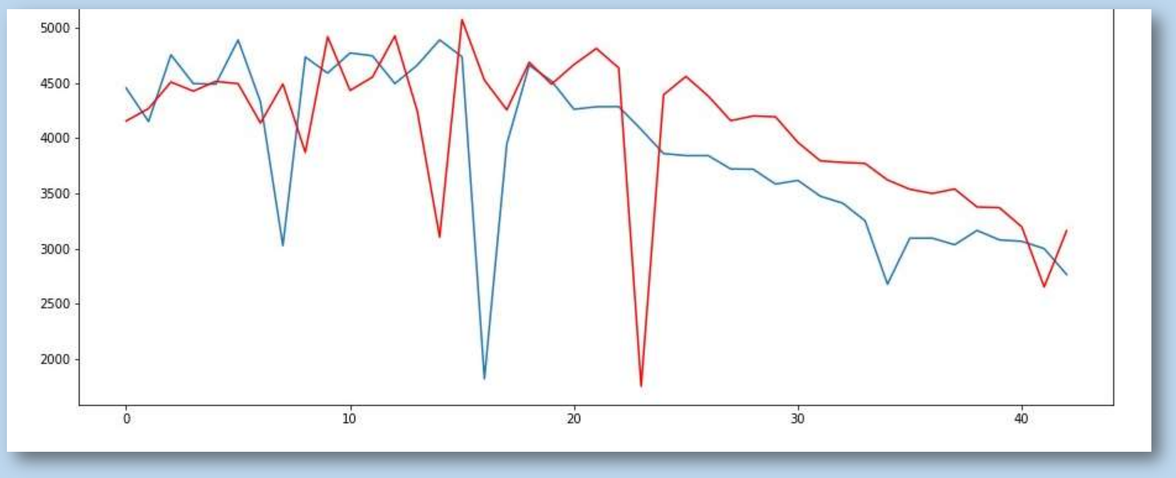

Fig 5. Outcomes of Sckit Learn for predictions without «holiday», red line indicating predictions and blue line indicating real quantity values.

CONCLUSIONS

	Neural Networks	ARIMA	Prophet	Sk-learn
With holiday	586.276	582.5763	396,39	-
Without holiday	658.890	709.255	596,76	763.061

Fig 6. RMSE results obtained with different methodologies.

As can be seen from the table above, different methods and libraries conduce toward various RMSE outcomes.

As for the future work of the project, it is planned to improve the methodologies to gain smaller RMSE results and then to decide which method fits the data better or to combine different methods to gain even better results.

REFERENCES

[1]https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html#Introduction
[2]http://pyflux.readthedocs.io/en/latest/arima.html
[3] https://github.com/facebook/prophet/blob/master/README.md
[4] http://scikit-learn.org/stable/