# Experimenting with Scaleable Forecasting Models

Matthew Coghlin Pelumi Dacosta Joseph Despres Aneesh Gahi

Joseph Sigler

December 3, 2021

## Introduction

Planning any organizational activities such as inventory planning, staffing decisions, budgeting, all depend on some kind of expectations for the future. Forecasting the practice of using current data available to make a prediction about the future value of a variable. Decision makers in government, businesses, and non-profit organizations all require accurate and reliable forecasts when planning their various activities. There are tremendous costs associated with forecasts that are either too high or too low. Therefore, as practitioners, it is essential to minimize forecasting error.

Organizations are collecting more and more data as it becomes cheaper to store and more convenient to collect. This presents the opportunity to use more and more data driven forecasts and less judgment based forecasts. A time-series expert is generally required to carefully tune ARIMA model parameters (Taylor & Letham, 2018). Although, this approach is riggerois, it does not scale, any standard grocery store has more items than you could feasibly tune manually. Analysts could be asked to generate high quality forecasts for thousands, and even hundreds of thousands of series at a time. The goal of this project is to implement algorithms that generate high quality forecasts with minimal involvement, validate them with a training and testing partisan, and generate ensemble predictions of the different methods.

## Data

The data obtained for this project are provided as part of a Kaggle challenge where participants are to forecast daily retail sales demand (Kaggle, 2018). As contestants, we are given 5 years of training data, with the daily sales of 50 different products from ten different stores. This is a total of 913,000 data points to train forecasting models. The goal is to forecast the next 90 days for each of the 50 products and 10 stores. Judged by Scaled Mean Absolute Percentage Error (SMAPE) shown in Equation 1

$$\text{SMAPE} = \frac{100\%}{h} \sum_{t=1}^{h} \frac{|\hat{y}_t - y_t|}{(|y_t| + |\hat{y}_t|)/2}. \tag{1}$$

where $y_t$ is the actual value, $\hat{y}_t$ is the forecast, and $h$ is the forecast horizon.

The data required very little preparation. There were several data points that were zero we switched to a one because the one of the algorithms did not support 0 values. After that we combined the stores and items into one string column to avoid nesting loops when iteration over stores and items.

After that, we separated into training and testing partisans to get an idea of what kind of model performance we can expect. We selected the first 1279 data points or 80% training set of our time-series. Then we separated the remainder of the data as a testing set. This prepares us for running the experiments.

(NEED PLOTS)

These data are highly seasonal with a slight upward trend. These data have significant noise. the vast majority of data involving human activities have a seasonal and trend component (Taylor & Letham, 2018).

## Requirements

Using algorithms to generate high quality forecasts will have strict requirements. These need to be fast, accurate, and interpretable. To be fast, the models must have minimal parameters to tune or be accurate with some specified default parameters. Grid searching model parameters over many series is not feasible. If these alogrythms are fast, many can be ran. Therefore, you can have a very good idea of what kind of performance to expect. Due to the high costs associated with forecasting errors, these models

must be interpretable in the event a steakholder is undertain about the model. For obvious reasons, stakeholders should not be asked to place their faith in black-box forecasting models.

# Models

There are many forecasting models, however we selected models that have been shown to perform well and fit our project requirements. We choose to test two categories of models. First, classical models that have been successfully implemented with years of good results. They are derrived with statistical methods and have strong theoretical justifications. Second, are newer promising machine learning based models. This is an open field and we are going to test different models and evaluate them strictly on their performance on testing data.

1. Seasonal Exponential Smoothing
2. Vector Autoregression
3. Autoregressive Distributed Lag
4. XGBoost
5. Prophet
6. Neural Prophet

## 1 Seasonal Exponential Smoothing

Exponential smoothing, or this is sometimes known as the holt winters method. decomposes the timeseries into three components: Seasonality, Trend, and slope. With the effevt of seasonality and trend to be linear. (Hyndman & Athanasopoulos, 2018)

## 2 Vector Auto Regression

The first algorithm we implement, is an autoregressive model. This takes the first 5 lag positions and uses them as regressors, then using timeseries decomposition, it models the seasonality. This model is commonly used to forecast economic variables.

## 3 Auto Regressive Distributed Lag

ARDL models add to the above auto regressive model, however in addition to seasonality with is fit with a vector of indicator variables. and trend, in this case we are adding an explanatory variable of time and fitting the model to laggs of time.

Although there are many forecasting models to choose from, there is not much research on when a given forecasting model outperforms another. Due to the data having strong weekly swings, we implement several models with an autoregressive terms.

## 4 XGBoost

XGBoost is an implementation of a tree boosting system. This uses decsision tree regression, and fits an ensamble of models fit to the data, then the residuals, then fits the residuals residuals. This ensamble of tree boosting is quite robust and is useful for a variety of different regression and classification tasks.

## 5 Facebook's Prophet

Facebook released a forecasting library designed specifically to meet the challenges of generating many high quality forecasts. The model prophet is a General Additive model, that consists of three functions, trend which fits an aperiodic logistic population growth model (we did not limit the growth, however that is a parameter), seasonality is a fouire series fit to the remaining seasonal component, and a holiday parameter which is a vector of user specified holliday periods, the holliday periods saw a drop in sales, however not enough to justify us specifying specific dates.

## 6 NeuralProphet

NeuralProphet, is a forecasting library that expands on facebook prophet and includes an autoregressive term in the general additive model and uses neural networks to generate the autoregressive terms in the model.

# Performance

# Kaggle Challenge

Feature Detection

There are many different shapes and patterns a timerseries plot can take. Tsfeatures (Montero-Manso, Athanasopoulos, Hyndman, & Talagala, 2020) quantifies different features such as the number of times it crosses the median, degree of seasonality, the number of flat spots, entropy, and many more.

We ran this algorythm on each series collecting 30 quantified features. The central question of this study is to determine if we can determine which forecast will perfrom the best given these features.

Predict Which Day the Model Has the Smallest Error Given the Features

Using different models, we tried knn, logistic regression, XGBoost, and an artificial neural network.

Due to the way kaggle scores, final Predictions could not be a weighted average of the results. We selected which was most likely to be the final.

The project will use that clean data and input it into a feature detection algorythm which quantifies various features of each series. Then Takes the data and runs six different forecasting algorythms to generate a prediction for the remainder of the testing data. Then analyze forecasting errors. Then attempt to use machiene learning methods to select the best model for the data given features

# Conclusions

The models ensamble was able to outperform each of the individual. In practice, it is likely that we are too use these methods to forecast a months sales. this would be aggrigated.

# References

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: principles and practice*. OTexts.

Kaggle. (2018). *Store item demand forecasting challenge.*

Montero-Manso, P., Athanasopoulos, G., Hyndman, R. J., & Talagala, T. S. (2020). Fforma: Feature-based forecast model averaging. *International Journal of Forecasting*, *36*(1), 86–92.

Taylor, S. J., & Letham, B. (2018). Forecasting at scale. *The American Statistician*, *72*(1), 37–45.