Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Monterrey

School of Engineering and Sciences


# An Ensemble Forecasting Framework for Time Series

A thesis presented by

Alejandro Saldaña Rodríguez

Submitted to the
School of Engineering and Sciences
in partial fulfillment of the requirements for the degree of

Master of Science

in

Engineering Sciences

Querétaro, Querétaro, November 2021

# Dedication

Primarily this thesis is dedicated to Trick, without whose presence in my life, it wouldn't have been possible or as fun and enjoyable. I'll always be grateful for everything you've done.

To Mónica who dutifully cheered me on, Héctor whose support made it possible for me to achieve this dream, Lorena who positively nudged me on, Diego who spent countless hours listening to my ideas, Sofía who always heard with a smile what I was up to, Héctorin who I wish to inspire, and Emilia whose playtime included the backtrack I provided while playing with stuffed animals.

This thesis is also dedicated to those who always asked and whose questions helped me understand my own reasoning: Ales, Ariel, Mon, Lillian, Xavier, Dav, Gominoli, Pabli and Cristina who additionally inspired me to pursue a postgraduate degree. To Uge, Eugenio, Alejandro, Dani, and Mers who I hold dearly in my heart. To my colleagues Linda, Benjy, Luis, Fabián, Christián, and Alejandro.

# Acknowledgements

I would like to thank my committee, first and foremost, Juan Carlos Espinoza for his much needed guidance, his patience, and most of all his camaraderie throughout all of my ventures into the academic lifestyle. May the accomplishments we achieved together be the start of many more to come. Betty for her gentle nudge into what would be an amazing odyssey and her subtle yet firm steering inside the "thesis waters". Fabi, for her continued support and encouragement I'll be always grateful.

My deepest gratitude to Tecnólogico de Monterrey for honoring me with the scholarship that allowed me to pursue the dream of obtaining this degree and CONACyT for the continued support throughout the completion of this program.

# An Ensemble Forecasting Framework for Time Series

Alejandro Saldaña Rodríguez

November 2021

**Abstract**

Forecasting for businesses is essential and, because small to medium sized enterprises cannot afford to spend the resources on accurate forecasting, the necessity to build step-by-step procedures that aid in this process is vital. Forecasting using machine learning or more complicated models comes with its own sets of challenges as many of them have parameters that are not directly interpreted to the variables. Ensemble Forecasting is a mixture between machine learning and forecasting and it uses many proven mathematical concepts such as the law of large numbers, the Jury theorem, and proven empirical evidence of these models outperforming the single models counterparts. This thesis proposes a new methodology to modernize and include the data analytics part of the cross industry standard process for data mining described in [25] to the time series analysis methodology proposed by [7]. The ensemble methods composed of linear combinations and majority-rule voting made better predictions and the new Ensemble Forecast model proposed in this thesis proved to be more accurate and precise than any other model including the other ensembling methods.

# Contents

**List of Figures** **4**

**List of Tables** **5**

**1 Introduction** **6**

**2 Literature Review** **9**

2.1 Loss Functions . . . . . 10

2.2 Time Series . . . . . 13

2.3 Forecast Models . . . . . 15

2.3.1 Traditional Forecasting Models for Time Series . . . . . 17

2.3.2 State-of-the-Art Forecasting Models for Time Series . . . . . 19

2.3.3 Model Parameters . . . . . 21

2.4 Ensemble Techniques . . . . . 23

2.4.1 Boosting and Bagging . . . . . 23

2.4.2 Linear Combinations . . . . . 25

2.4.3 Majority-Vote Algorithms . . . . . 28

2.5 Other Works . . . . . 30

**3 Methodology** **33**

3.1 Business Understanding . . . . . 36

3.2 Data Understanding/Preparation . . . . . 37

3.3 Existing Models . . . . . 38

3.3.1 Hierarchical Clustering . . . . . 40

3.4 Ensemble Algorithm . . . . . 47

**4 Results and Discussion** **50**

4.1 Results . . . . . 50
4.2 Bias Evaluation . . . . . 51
4.3 Variance Evaluation . . . . . 54
4.4 Relative Forecast Error Evaluation . . . . . 56
4.5 AIC Against Other Criterions . . . . . 59
4.5.1 Bias Evaluation . . . . . 60
4.5.2 Variance Evaluation . . . . . 61
4.5.3 Relative Forecast Error Evaluation . . . . . 62
4.5.4 Criterion Selection Conclussion . . . . . 62

**5 Conclusion** **63**

**6 Future Work** **65**

**References** **71**

# List of Figures


1 Search Frequency Trends . . . . . 7

2 Example Time Series . . . . . 14

3 Boosting and Bagging . . . . . 25

4 2014 - 2018 CAQ Sales . . . . . 38

5 Weekly Sales by Model . . . . . 40

6 Clustering by Sales . . . . . 43

7 Clustering by Weekly Changes Percentages . . . . . 45

8 Clustering by Weekly Increase/Decrease . . . . . 46

9 State of the Art Models for the L-65-800 . . . . . 50

10 Traditional Models for the L-65-800 . . . . . 51

11 Dispersion of MBE by Model . . . . . 54

12 Dispersion of MSE by Model . . . . . 56

13 Dispersion of MPE by Model . . . . . 59

14 Clustering by Sales . . . . . 68

15 Clustering by Weekly Changes Percentages . . . . . 69

16 Clustering by Weekly Increase/Decrease . . . . . 70

# List of Tables


1 Model Parameters for Studied Models . . . . . 21

2 Monthly Data point for L-58-575: 2021-07-18 . . . . . 27

3 Monthly Data point for L-31T-900: 2021-02-07 . . . . . 29

4 Best Performing Models for Monthly CAQ's Aggregated Sales . . . . . 39

5 Monthly L-31T-900 Sales . . . . . 39

6 Monthly L-65-800 Sales . . . . . 39

7 Absolute Precision Error in Monthly Sales . . . . . 41

8 Mean Bias Error by Model . . . . . 52

9 Mean Bias Error by Battery Model . . . . . 53

10 Mean Square Error by Battery, Models Better than Ensemble Forecast Highlighted . . . . . 55

11 Mean Percentage Error by Model . . . . . 57

12 Mean Percentage Error by Battery, Models Better than Ensemble Forecast Highlighted . . . . . 58

13 Mean Bias Error by Criterion . . . . . 61

14 Mean Square Error by Criterion, Best Criterion Highlighted . . . . . 61

15 Mean Percentage Error by Criterion . . . . . 62

# 1 Introduction

Forecasting, besides allowing users and companies to maintain service levels and stock, facilitates goal setting and capacity planning. Both of these additional benefits of accurate forecasting result in better key performance indicators and additional resources for the decision maker to use. There are many challenges to creating reliable and high quality forecasts, especially considering the scarcity of analysts with expertise in time series and modeling [31]. Along the issue of not having enough time series forecasting analysts, the subsequent problem of the absence of expert forecasters: the interest in publishing, searching, and open sourced projects has maintained there low levels. In Figure: 1, with data compiled from Google Trends (https://www.google.com/trends), it can be seen that the amount of people searching for *machine learning* greatly surpasses the amount of people using the words *forecasting* or *time series analysis*. Forecasting is a skill where deep knowledge is required to build, or even in some cases at least understand, some of the new models and application methodologies. Because of these reasons, qualitative and simple quantitative models are still being used in many companies, specially small and medium sized companies (PYMES for its acronym in spanish). According to the Institute of Business Forecasting and Planning (IBF): "Popular doesn't always equate to quality, and an ERP or an advanced planning system that claims to do it all may not have the forecasting tools you need." [34]. In this quote it is clearly stated that the industry standard for forecasts is not very high. [34] also writes that the typical forecasting software costs from $5,000 to $30,000 USD per user which means that for every $100,000 USD seen in revenue $2,000-$6,000 USD is spent for this purpose. If instead of buying software the company in question decides to go through consulting the rates are also described in the $110-$220 USD an hour. This, for small and medium sized companies may not be sustainable.

Selecting the correct inventory policy can have a meaningful impact on any business. From a

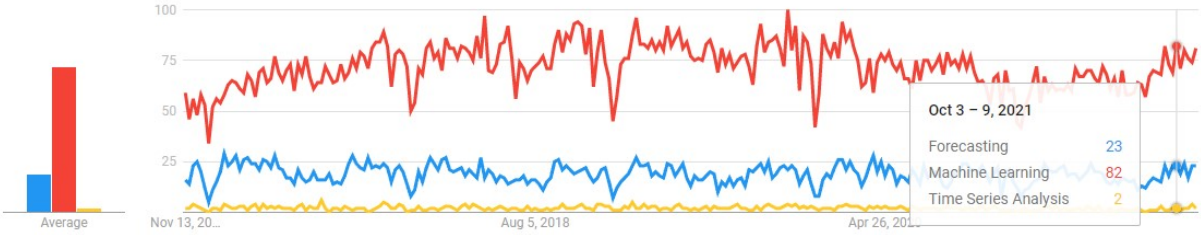


Figure 1: Search Frequency Trends

healthier cash flow to less physical space needed to provide the target service level, inventory policies can help companies manage and order different aspects of their business to become more profitable. Having an inventory policy that is optimal for every business is critical as it allows the management to select an evaluation criterion or key performance indicator from which to understand and analyze where it's located in regards to inventory management. To select and apply a proper inventory policy one must be able to correctly and aptly create accurate forecasts. Inventory management has allowed companies to remain profitable by analyzing lead times, physical space, holding costs, service levels, amongst other variables and give management the information needed for them to take action. Once an inventory policy is selected, be it by an empirical or analytical decision, management must be able to comply with it and forecast the changes in their demand for them to manage their inventory in optimum levels.

CAQ, a regional distributor for LTH batteries, has the goal of reducing the inventory levels through a decrease in forecasting errors by using an industry standard level forecasting method instead of the integrated ERP that uses information from sales in the previous weeks to accurately estimate future periods. Given that the company has allowed the use of their data, this thesis will explore of some ways to reducing the forecasting errors as well as developing a framework for time series analysis. CAQ has many different products, a list

which goes over the 200 different battery models, and uses heuristic modelling of their sales every week in order to replenish stock. The company's director has established that there are no battery specific forecasting policies for any of the models and that in many cases errors could be dire. Because CAQ belongs to the PYME category, it does not have the means to hire or train an employee with the task of producing better fitting models for forecasting or data analysis purposes. This type of companies would benefit from easy graphic user interfaces or a simple software/program that allows them to obtain better forecasts with ease and precision.

In the case of CAQ, inventory levels are critical because they sell car batteries in a metropolitan region in Queretaro, Mexico and keeping the correct stock levels aids them in providing a high service rate while keeping costs down. CAQ's sales data will be analyzed and subjected to different forecasting methods. Inventory levels must not be so high as to fall into unnecessary holding costs yet not so low as to lose sales by under inventorizing. Forecasting techniques aid management to determine these levels and prevent some of the inherent risks of not inventorizing correctly. Every forecasting technique comes with its advantages and disadvantages, which will be explained in section 2.3, however given the fact that state-of-the-art for non-time-series applications have encountered that ensemble methods for machine learning applications result in great prediction results, ensemble techniques for forecasting techniques will be analyzed to observe if these predictions conserve the advantages of the models to allow for a smaller error in predictions. Ensemble techniques have appeared throughout machine learning applications long before open sourced resources made them available to anyone searching. Their great performance have even seen them win many Kaggle competitions, even inspiring the article [24]. Given the effectivity seen in other data science applications, ensemble techniques should be studied through the time series lenses and applications.

This thesis will propose an ensemble method for forecasting after analyzing if different products in the same categories, be it magnitude or demand shape, result in better fits for different models and then proving this forecasting method can provide better results than the analyzed models.

# 2 Literature Review

In this section the different loss functions, the data type: time series, forecasting models, and ensemble techniques will be explained. Forecasting models that are going to be explored include traditional methods that come from the XXth century as well as the newest forms of applications. Even though forecasting has been around for a long time, machine learning applications and current trends are starting to explore the different possibilities that time series' studies have on academic and industrial applications. Moez Ali, creator of the PyCaret Open source python package, said on October 22nd 2021 the beta release of PyCaret's Time Series Module for a soon to be announced date. His profile has over 49,000 followers and the github repo has over 4,000 stars and 965 forks. A famous data science corporation, Towards Data Science Inc., published [32] where the author explained a new python package that envelops many useful forecasting techniques into a single coding format. Many different open source programs and codes are being pushed forward by many technological writers in different non-peered-reviewed journals and web publications were they share tips, tricks, codes and how-to's for scientists. This technological innovation sharing is growing popular among researchers because many of the applications can be downloaded as open source code and the developers can get feedback on their packages.

## 2.1 Loss Functions

Models use different loss functions to evaluate how well they fit any given data. Different loss functions are used depending on the different type of fit that is wanted, for example inside the forecasting application in the industry if the holding costs are high for certain products, managers might choose a loss function that does penalize outliers heavily. This loss functions selection results in models that have certain context and perform better than box models. Optimization functions allow loss functions to reduce errors in the model's predictions. There are no right answers when picking a loss function due to the differences between them. Regression losses that are commonly used in forecasting applications, as seen in [22] and [17] where the authors compare some of the accuracy measurements, include but are not limited to:

- Mean Square Error/Quadratic Loss/L2 Loss: Mean squared error is the squared measured difference between forecasted predictions and real values. This measurement results in heavy penalization of predicted values that are far away from the real values compared to nearer predictions.

$$MSE = \frac{\sum_{t=1}^{n}(Y_t - \hat{Y}_t)^2}{n} \tag{1}$$

- Mean Absolute Error/L1 Loss: Mean absolute error is the measured absolute difference between forecasted predictions and real values. This measurement, just like MSE, does not consider the direction of the errors. Because it does not square errors, it is more robust to outliers.

$$MAE = \frac{\sum_{t=1}^{n}|Y_t - \hat{Y}_t|}{n} \tag{2}$$

- Mean Bias Error: Mean bias error is less common than both of the loss functions de-

scribed above. This error is calculated by measuring the difference between forecasted predictions and real values.

$$MBE = \frac{\sum_{t=1}^{n}(Y_t - \hat{Y}_t)}{n} \tag{3}$$

- Mean Precision Error: Mean precision error is a relative forecast error that is used to obtain as a percentage the difference between the real values and forecasted predictions. This measurement has the possibility of weighing different errors, positive and negative, differently if such case might present a better fit.

$$MPE = \frac{(\frac{\sum_{t=1}^{n}(Y_t - \hat{Y}_t)}{Y_t}) * 100}{n} \tag{4}$$

- Mean Absolute Precision Error: Mean absolute precision error is another form of explaining relative forecast error. This measurement is taken by calculating the absolute difference between forecast and real value and is commonly used when the type of error, be it positive or negative, has the same weight. This error results in a great measurement for when analyzing total error, in this case the average amount of deviation from the actual values.

$$MAPE = \frac{(\frac{\sum_{t=1}^{n}|Y_t - \hat{Y}_t|}{Y_t}) * 100}{n} \tag{5}$$

Additional loss functions for regression include the Huber Loss (Smooth Mean Absolute Error), Log-Cosh Loss, and the Quantile Loss. All of them have different pros and cons. Huber Loss could be described as the middle ground between MSE and MAE however, the hyperparameter of the equation requires iteration to obtain the best values for it. The Log-Cosh Loss has most of the same advantages the Huber Loss but it suffers from the problem of

hessian and gradient for very large errors being constant. Quantile Loss works by providing intervals instead of point based values. This last model works well with heteroscedastic data.

In [17] the authors analyze and compare MSE, MAE, MAPE, their own creation Mean Absolute Scaled Error (MASE), and other measurements to show that some of these measurements should not be applied globally. To measure the differences the authors use 4 simple methods: historical mean, naïve or random-walk method, SES and HWES. MASE builds from the strengths of MAE and it follows some of the same logic of scaling errors using a benchmark model and comparing it to the studied model. The authors conclude that their loss function has more steps and that there are situations were easier and more directly interpretable measurements such as MAE and MAPE should be used. MASE depicts the forecast accuracy best when dealing with data that holds many different scales.

[18] proposed, alternatively, in the *International Journal of Production Economics* that there is merit in optimizing inventory targets instead of forecast accuracy as many of their objective functions resulted in minor gains when lead times were considered. The authors also propose an optimization over the gap in service level: $Cost = (\text{Target CSL} - \text{Real CSL})$ where CSL is *Cycle Service Level* or cost based: $Cost = \text{Lost Sales} + \lambda(\text{Stock on Hand})$ where $\lambda$ is any given cost ratio. The authors used these formulas in order to diminish errors when comparing to mean squared forecast error in its one-step ahead ($t+1$) and with L-steps ahead ($L$):

$$MSE_{t+1-t+L} = \frac{1}{L}\sum_{i=1}^{L} MSE_{t+i} \tag{6}$$

And the other alternative equation, cumulative mean squared error over L which results in a minimization of the total demand over lead time, not by period:

$$cMSE_{t+L} = \frac{1}{n-L+1} \sum_{t=L+1}^{n} ( \sum_{j=t-L+1}^{t} y_j - \sum_{j=t-L+1}^{t} \hat{y}_{j|t-L}) \tag{7}$$

This equation implies the smoothing of data, it's equivalent to overlapping temporal aggregation, lessens training sample and data becomes less volatile.

For each specific forecasting application a different loss function might be required, e.g. [20] established that in a financial context investors are ultimately trying to maximize profits more than they are minimizing risk thus, MSE and MAE loss functions would not work because it penalizes errors above and below the real value by the same margin. For inventory policies each company should determine their own politics so as to define whether the objective function should penalize the same way over stocking a product or facing having a lower service level.

## 2.2 Time Series

Time Series Analysis is the analysis of a special type of data that was defined by Box et al. in *Time Series Analysis: Forecasting and Control* as a sequence of observations taken in given steps over time. Many sets of data contain this format and are applied over distinct areas throughout humankind's history. Be it the gross domestic product (GDP) in economics, to inventories in business and from climate in meteorology to bacterial colonies growth in biotechnology. From [14] the Figure 2, the regular structure for time series can be observed. The time series can be structured by any given time frame, be it intradaily, weekly, monthly, yearly or any other time frame. On the vertical axis any numerical, continuous or discrete, variable is plotted over the selected time frame. The elements of a time series are comprised of seasonality which depicts the regular cycles over inter-periods, the trend that calculates the long period movement over time, and the noise which captures the variance not obtained through the other two terms. Different time series can have seasonality and trend but, it

can also be stationary. Depending on the data time series can be stationary, which can have cyclic behaviour, or have trends throughout time and this should be considered in the models. Time series are usually time dependent observations and based on that nature many different tools have been developed, some of them described in 2.3, to produce forecasts, analyse interrelationships, and to try and find noise in the dataset. In a business setting forecasts have long been used since before [15], originally published in 1957, his own very popular and still used method for developing a systematic forecast expression for exponential weighted moving averages (see section 2.3) and as stated in the introduction, there are still many different areas of opportunity improve forecasting and its applications in time series.

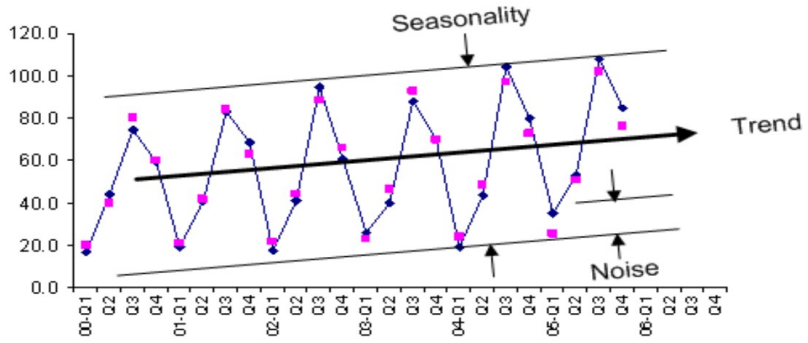


Figure 2: Example Time Series

Since time series data has been used, different project management methodologies have been developed to better get information out of time series analysis. Many scholars have proposed methodologies that work with their own models and some have developed models to work with certain given methodologies such as [31] that developed a model to work for Facebook's data and work environment (more on 2.3), and [7]. To work with time series one can follow the steps of one of the more popular and easy methodologies out there identified in [7] which are:

1. Model Specification: the model types are selected depending the kind of dataset obtained. During this phase one can also compute different statistics and incorporate

experience from the application's experts. The selected model/models are just tentative at this stage.

2. Model Fitting: consists in finding the best parameters and hyperparameters, if applicable, of each model. Possible criteria includes: Akaike information criterion (AIC), Bayesian information criterion (BIC), Maximum likelihood estimation (MLE), least squares, amongst others. These measurements will be explained further in 2.3, what is important to note is that these measurements penalize overfitting as harshly as underperforming because overfitted models lose the ability to generalize correctly and perform worse on test settings.

3. Model Diagnostics: best explained as the assessment of the quality of the model that has been parametrized and applied. It is important to not only check the model but to understand the implications of our assumptions and if these assumptions deter the correct application of the model.

Once the steps above are fulfilled one may decide to loop through them again if any inadequacy has been found, if not the model may then be used to forecast future values. Some time series analysis are computationally intensive and more so modern ones and hybrid models that work hand in hand with an optimizer to find the best parameters and hyperparameters of a model so it is best to plan ahead when coursing through a time series application.

## 2.3 Forecast Models

Before getting into some of the forecasting models, it is important to establish that even though time series analysis has been studied for some time now, some of the models that are used as industry standard have not changed much. In this subsection, the forecast models selected for the analysis of this thesis will be explained and the chosen parameters will be explained. Forecasting techniques with a machine learning focus are starting to get developed

by diverse technological companies as is the case of one of the selected models. Statistics are so ingrained in data and machine learning that many data scientists are studying the interaction of the intersection between time series analysis and machine learning as many of the conventional tools of time series analysis are based on statistical methods.

Forecasts have been used for many applications throughout history. Weather predictions, albeit in the early stages non-numerical, have been used in civilizations throughout time. From the ancient greek sanctuary, Delphi is one of the most common forecasting methods. The Delphi method relies on an expert panel, or people knowledgeable on the subject, to make a consensus given that it assumes that the groups answers will be better than any individual. This model along with market research and historical life-cycle analogy. The Delphi method has been long studied such as in [35] where he found that no evidence linked the Delphi method to producing more accurate judgements and forecasts or that consensus is better than single estimation. The data the author presented states that consensus is usually achieved because of group pressure to conformity or statistics fed back to participants by the response. This paper researched different articles and studies and summarised the information in three points, the first stating that the aggregation of several individuals was indeed more accurate than the judgement of a random individual, though not by statistical significance, second, the judgements from interacting groups perform better than a statistically aggregated judgement, and last that direct interaction has the disadvantage of leading to suboptimal accuracy of judgements. In ensemble forecasts, as there is no one "expert"the interaction amongst models is difficult to understand due to the inner workings of each of the components. Inside the Delphi method, there are different workings that the academy has studied: the Delphi to estimate unknown parameters, the policy Delphi, and the decision Delphi. The market research method has been long used to forecast demand of different products and services however, not many academic publications are obtained in

the quest of finding these articles.

### 2.3.1 Traditional Forecasting Models for Time Series

There are many forecasting techniques that have been around since the 20th century (from hence forth classified as traditional models): The models in this section are described and applied in [16]

- Moving Average (MA): This model serves as a smoothed version of the original time series because it takes a rolling average over the data. The parameter that is moved is the length of time frame from which the average is taken. The formula is as follows:

$$m_t = \frac{1}{N} \sum_{t=T-N+1}^{T} y_t \tag{8}$$

- Simple Exponential Smoothing (SES): This model uses an exponential smoothing factor in the form of $\alpha \in [0, 1]$, and it acts as a filter for outliers as it smooths the shape of the prediction reducing noise in the data. The simplest form of SES is as follows:

$$s_t = \alpha x + (1 - \alpha) s_{t-1}, t > 0 \tag{9}$$

  The SES can also be expressed as a recursive form or as an expression with a discount factor $\lambda$.

- Holt-Winters (HWES): This model uses an overall smoothing technique ($s_t$), a trend smoothing technique ($b_t$), and seasonal smoothing technique ($c_t$). These three components interact with the following formulas where $L$ is cycle length, $t$ is the period, smoothing factor $\alpha \in [0, 1]$, trend smoothing factor $\beta \in [0, 1]$, and seasonal change

smoothing factor $\gamma \in [0, 1]$.

$$s_t = \alpha \frac{x_t}{c_{t-L}} + (1 - \alpha)(s_{t-1} + b_{t-1}) \tag{10}$$

$$b_t = \beta(s_t - s_{t-1}) + (1 - \beta)b_{t-1} \tag{11}$$

$$c_t = \gamma \frac{x_t}{s_t} + (1 - \gamma)c_{t-L} \tag{12}$$

- Autoregressive (AR): This model works by using a linear combination of past values of the studied variable. The term autoregression stems from the concept that it is a regression of the variable against itself. An autoregressive model with order $p$ can be written as:

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + ... + \phi_p y_{t-p} + \epsilon_t \tag{13}$$

  where $\epsilon_t$ is the error term for white noise.

- Autoregressive Moving Average (ARMA): This model is much like the AR with the difference that instead of using past values of the forecasted variables, the model uses past forecast errors. The model can be thought of as a weighted moving average of the past forecast errors. The formula for an order $q$ is denoted as follows:

$$y_t = c + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + ... + \theta_q \epsilon_{t-q} + \epsilon_t \tag{14}$$

- Autoregressive Integrated Moving Average (ARIMA): This model results as a combination of both of the previous models. the model has both parameters of $p$ and $q$ and it has the added degree of $d$ that denotes the degree of first differencing involved. The

model can be written as follows:

$$y'_t = c + \phi_1 y'_{t-1} + ... + \phi_p y'_{t-p} + \theta_1 \epsilon_{t-1} + ... + \theta_q \epsilon_{t-q} + \epsilon_t \tag{15}$$

where $y'_t$ is the differenced series.

### 2.3.2 State-of-the-Art Forecasting Models for Time Series

Many of the state-of-the-art models are being developed by open sourced projects and some of the technological industries. These developments are being tested in thousands upon thousands of datasets. searching "Prophet Forecasting"in Google scholar you get around 16,200 results including many comparisons of the Prophet model and the seasonal version of the ARIMA such as [28]. It is just recently that these papers, thesis, and in proceedings are being published and the way analysts are performing time series analysis is changing. The studied state-of-the-art forecasting models are listed below:

- Seasonal Autoregressive Integrated Moving Average with Exogenous Factors (SARIMAX): This model was developed to handle seasonality because regular ARIMA models cannot use it. SARIMAX comes from the SARIMA model but includes the ability to consider exogenous factors in its application. The SARIMA model can be found in [16] and it writes the following notation for SARIMA:

  $$SARIMA(p, d, q) \times (P, D, Q, S)$$

  The SARIMAX model can be found along with its documentation in [30].

SARIMAX is a model that besides handling seasonality it can handle exogenous variables such as holidays if these or other additional variables were relevant to an application. Because the python package uses SARIMA and SARIMAX in the same wrapper it will be continued to be shown as SARIMAX even if no exogenous variables were considered. The seasonal

parts of the model $(P, D, Q, S)$ consists of terms that are similar to the orders and degrees of the ARIMA model but involve backshifts/lags of the seasonal period, i.e. the backshift is calculated for each seasonal period. For the final forecast, the seasonal terms are simply multiplied by the non-seasonal terms.

- Prophet: This model was developed by [31] and published in 2017. This model uses modular regression and its available free in different Python and R packages. The model uses the following general equation:

$$y(t) = g(t) + s(t) + h(t) + \epsilon(t) \tag{16}$$

  Where $g(t)$ is the trend function which models non-periodic changes in the value of the time series, $s(t)$ represents periodic changes and seasonality, $h(t)$ represents the effects of holidays which occur on potentially irregular schedules over one or more days, and last $\epsilon(t)$ are the changes not captured by the model.

The Prophet model was developed with a human-in-the-loop approach to address the challenges faced when there are a variety of time series and analysts and it uses interpretable parameters so as to aid in the analysis and testing endeavours. Their idea to create this model was to target two problems, the first being that in business applications automatic forecasts are hard to tune or are too inflexible to incorporate heuristics. The second issue they wanted to solve was that the analysts that create forecasts usually have no training in time series forecasting, even if they do possess the deep domain expertise about the company's products and or services. The authors of this model compare the effectivity of their model with ARIMA in different situations. Some of the pros of the Prophet against ARIMA is that measurements do not need to be regularly spaced, fitting is very fast, and because it treats time series as a curve-fitting exercise it has easily interpretable parameters. This

does not come without cons, one of them is that the model loses some of the inferential advantages.

[26] developed a comparison of forecasts' accuracy, SARIMA and Holt-Winters, using data from the Indian motorcycle industry. Due to its market size and influence over Indian economy, the authors wanted to understand and forecast the tendencies of this market and observe the differences between the Holt-Winters' and the SARIMA model. In this paper the authors utilize MSE, MAE, and MAPE as the measure of errors of each of the models and use them all as metrics as to select the best model. Puthran et al. found that even though the models resulted in very close values, Holt-Winters was the better model at forecasting Indian motorcycle industry and it was easier to use thus their recommendation did not change for this application. This study shows how different metrics might give different results and how forecasting techniques for time series data are still being tested.

The following parameters are defined for the application of the models:

### 2.3.3 Model Parameters

Table 1: Model Parameters for Studied Models

| Model | Parameters |
|---|---|
| MA | Rolling Mean = 3 |
| SES | Smoothing Factor =0.2 |
| HWES | $\alpha, \beta$, and $\gamma$ are selected for each dataset |
| AR | $p = 1$ |
| ARMA | $p = 2, d = 0, q = 1$ |
| ARIMA | $p = 1, d = 1, q = 1$ |
| SARIMAX | Seasonality = 7, Dynamic Parameters $p, d, q, P, D, Q \in [0, 2]$ |
| Prophet | Changepoints = 15 |

Each parameter from the table had a distinct reason to why it was chosen. The MA's rolling mean and SES' smoothing factor were chosen based on performance at the aggregated monthly level and to have smooth forecasts predictions to counteract more volatile models. The AR's, ARMA's, and ARIMA's parameters were selected using [16] recommendations including: "In practice, it is almost never necessary to go beyond second-order differences". The ARIMA model has some special cases enlisted below:

- White Noise: ARIMA(0, 0, 0)
- Random Walk: ARIMA(0, 1, 0) with no constant
- Random Walk with drift: ARIMA(0, 0, 0) with a constant
- Autoregression (model AR): ARIMA($p$, 0, 0)
- Autoregressive Moving Average (model ARMA): ARIMA(0, 0, $q$)

These special cases are described in detail in [16]. For the HWES parameter the model was able to select, using SSE, the parameters $\alpha, \beta$, and $\gamma$ depending on each of the time series behaviour; this helped to keep some of the wanted smoothing while allowing more flexibility. The Prophet model uses a default parameter of 20 changepoints however, given that the training size was 20, the Prophet model changed its parameter to 15 changepoints. The last model, SARIMAX, worked in a hybrid way as it includes an optimization cycle where it trains values for $p, d, q, P, D, Q \in [0, 2]$ which means that a total of 729 combinations of the model would be evaluated each period. Each period SARIMAX's parameters could take different values, which kept the volatility that SARIMAX offered to the forecasting method proposed in this thesis.

## 2.4 Ensemble Techniques

Ensemble techniques are becoming increasingly popular in machine learning applications from classification to regression and clustering to feature selection. The logic behind ensemble techniques can be traced back to [13] through his jury theorem where he studied the relative probability of a given group of individuals arriving, by means of majority vote, to a correct decision. Even if his theorem most likely resembles a binary classification problem, his ideas have permeated throughout machine learning. Many relaxations of his Jury theorem exist e.g.: the random voter selection, weighted majority rules, amongst many others that have been used for different applications. In [23] the authors describe different methods and architectures of the algorithms that produce ensemble methods in machine learning. The main methods are bagging and boosting. Bagging is the ensemble method that works by training the model on a random redistribution of the dataset while boosting works by sampling incorrectly predicted instances in the dataset. A key difference between both techniques are that boosting is dependent on the performance of earlier models while bagging is not. Other ensemble methods include the Bayes optimal classifier, the Bayesian model averaging, the bayesian model combination, stacking, etc.

### 2.4.1 Boosting and Bagging

Boosting is a machine learning ensemble of models that is best described in [10] where he defines boosting in an easy to understand manner. He states that boosting algorithms work through the following steps:

1. All observations in the dataset are, initially, equally weighted.
2. A machine learning model is trained on the dataset.
3. The model from step 2 is added to the ensemble with an error associated to the misclassified observations.

4. Misclassified observations are given greater weights.

5. Steps 2-4 are repeated for any number of epochs.

The idea from these steps is to make misclassified observations stand out as being more important in subsequent iterations. In [29] the author makes an analogy from spam emails that also help visualizing the goal of boosting. He states that junk mail can be better guessed if one were to use many rules of thumb instead of a single highly complex yet accurate rule. Every iteration of the boosting ensemble generates a single weak rule that will generally and hopefully create a more accurate model. The weighing of the misclassifications and the combining of the weak rules are the key parts of extracting a fully functional and powerful model. Even though boosting is a very powerful technique in terms of its small errors, it is very prone to overfitting.

Bagging is the merging of machine learning models trained on bootstrap samples that are independent and identically distributed. Bagging, from Bootstrap Aggregation, was introduced by [8] were the results of bagging showed that bagging could give substantial gains in accuracy. Bootstrap is a type of sampling with replacement where all samples are drawn from a single original sample. Bootstrapping allows the analyst to estimate statistics of a population by averaging estimates from multiple small data samples. The bagging method uses averages when predicting a numerical outcome and a plurality vote when predicting a class. Originally performed on classification trees, bagging has become a popular ensembling method in machine learning. The bagging methodology is comprised of two main components, bootstrapping and a set of homogeneous machine learning algorithms. From the population $n$ amount of subsets are extracted from which $n$ models are trained on.

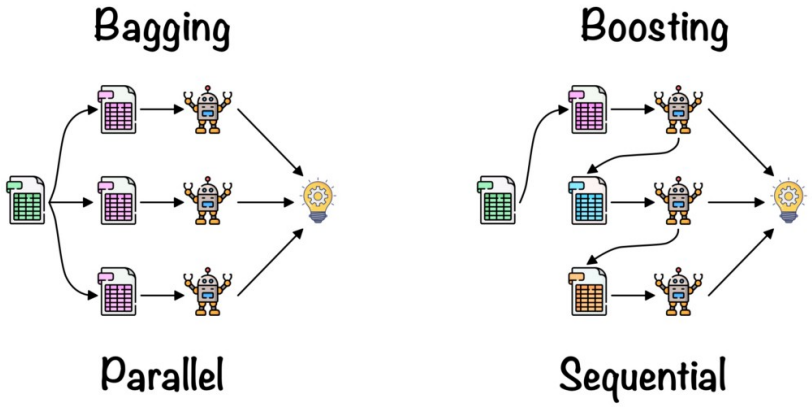


Figure 3: Boosting and Bagging

In Figure 3, taken from [21], in the bagging model the robots represent the machine learning algorithms and from the green file the three pink subsets are created to generate three different results from which a final result will be extracted. In the boosting model each iteration is represented by the robot that sequentially perform their tasks, i.e. making predictions.

### 2.4.2 Linear Combinations

Linear combinations can be seen in chapter 7 of [22], where the authors talk about combining using linear weights forecasts obtained with different methods. This idea stems from the extraction of knowledge each individual model is able to compile i.e. if a model, say ARIMA, accurately identifies a trend while another model, exponential smoothing, does not it would seem inefficient to ignore the additional information. This combination can be used to produce a superior forecast when comparing forecast errors. However in [22] the actual wording for the weights is left as: "If these weights are chosen properly, the combined forecast $\hat{y}^c_{T+\tau}$ can have some nice properties."For this thesis the proposal of the linear combinations named in this thesis as Olympics, MinMax, and Mean were created and are described as algorithms below:

The Olympic ensemble works by adding all of the models and subtracting from this sum the highest and lowest models of every step. This new sum is divided by 6 to obtain the average of the used models. The idea behind this algorithm is to obtain a mean of the middle pack of studied models reducing the outliers.

**Data:** Trained Models Forecasts: $Forecast_{models}$

**Result:** Olympics Ensemble Forecast

$Sum, Ensemble = 0$

$Sum = \sum Forecast_{models}$

$Sum = Sum - \arg\min\left(Forecast_{models}\right) - \arg\max\left(Forecast_{models}\right)$

$$Ensemble = \frac{Sum}{models - 2}$$

**Algorithm 1:** Olympic Ensemble Proposed Algorithm

The MinMax ensemble is calculated by using the smallest and highest forecasted value and averaging both to obtain the middle point of the forecasts. The hypothesis is that by averaging the outliers a new value could perform better than any single models.

**Data:** Trained Models Forecasts: $Forecast_{models}$

**Result:** MinMax Ensemble Forecast

$Ensemble = 0$

$$Ensemble = \frac{\arg\min\left(Forecast_{models}\right) + \arg\max\left(Forecast_{models}\right)}{2}$$

**Algorithm 2:** MinMax Ensemble Proposed Algorithm

The Mean is the easiest of the ensembles as it calculates the average of all of the models forming a smoothed version over time.

**Data:** Trained Models Forecasts: $Forecast_{models}$

**Result:** Mean Ensemble Forecast

$Ensemble = 0$

$$Ensemble = \frac{\sum Forecast_{models}}{models}$$

**Algorithm 3:** Mean Ensemble Proposed Algorithm

Using the data in Table: 2 the calculation for each of the combinations will be shown below.

Table 2: Monthly Data point for L-58-575: 2021-07-18

| Date | 2021-07-18 |
|---|---|
| Real | 102 |
| MA | 93 |
| HWES | 96 |
| SES | 100 |
| AR | 99 |
| ARMA | 91 |
| ARIMA | 101 |
| SARIMAX | 134 |
| Prophet | 104 |

$$\begin{aligned} Sum = \sum 93 + 96 + 100 + 99 + 91 + 101 + 134 + 104 = 818 \\ Sum = Sum - 134 - 91 = 593 \\ Olympic = \frac{593}{6} \approx 99 \end{aligned} \tag{17}$$

$$
\begin{aligned}
argmin &= \arg\min Forecast_{models} = 91 \\
argmax &= \arg\max Forecast_{models} = 134 \\
MinMax &\approx \frac{134 + 91}{2} \approx 113
\end{aligned}
\tag{18}
$$

$$
\begin{aligned}
Sum &= \sum 93 + 96 + 100 + 99 + 91 + 101 + 134 + 104 = 818 \\
Mean &\approx \frac{93 + 96 + 100 + 99 + 91 + 101 + 134 + 104}{8} \approx 102
\end{aligned}
\tag{19}
$$

With these algorithms the linear combinations for this thesis were created and analyzed. The results from these combinations proved to be versatile, easy to calculate, and provided better results than most of the models.

### 2.4.3 Majority-Vote Algorithms

The other types of ensemble used in this thesis pertains to the use of majority voting using different sets of rules that defines where the highest amount of votes result. This ideas spawned from [13] however due to the nature of the data not being a binary classification, the rules were defined using the structure of forecasts. Many different combinations of rules were tried and compared and amongst them the algorithms named Majority-Vote Mean (MVM), Majority-Vote Mean Median (MVMM), Majority-Vote Range (MVR), and Majority-Vote Range Median (MVRM). These rules were established using the following logic:

1. Select a point of reference: Mean or Half point
2. Count the number of predictions in each of the binary classifications
3. Select an aggregation method:
    - Mean
    - Median

Table 3: Monthly Data point for L-31T-900: 2021-02-07

| Date | 2021-02-07 |
|---|---|
| Real | 167 |
| MA | 159 |
| HWES | 164 |
| SES | 160 |
| AR | 156 |
| ARMA | 159 |
| ARIMA | 145 |
| SARIMAX | 155 |
| Prophet | 149 |

4. Aggregate the values

Using the following table the calculations for the Majority-Vote algorithms will be shown:

Using the same values from Table: 3 the following results would be obtained:

$$Mean = 155.875 \tag{20}$$

The number of forecasts bigger than the mean are 5 while smaller than the mean are 3. Because the majority of algorithms voted bigger then:

$$MVM = \frac{159 + 164 + 160 + 156 + 159}{5} \approx 160 \tag{21}$$

In the equation above the mean of all the models that forecasted values higher than 155.875 are averaged. This new value becomes the forecast for this ensemble.

$$MVMM = 159 \tag{22}$$

In the equation above the median of this points is found to be 159 therefore the forecast for

this ensemble is 159.

The same procedure operates for goes for the half point:

$$Halfpoint = 154.5 \tag{23}$$

The number of forecasts bigger than the half point are 6. Because the majority of algorithms voted bigger then:

$$MVR = \frac{159 + 164 + 160 + 156 + 159 + 155}{6} \approx 159 \tag{24}$$

$$MVRM = \frac{159 + 159}{2} = 159 \tag{25}$$

These Majority-Vote rules worked with good results as seen in the improvement over some of the models and this pattern could be seen throughout all of the data points for every dataset. This types of ensemble are easy to establish, quick to implement, and most importantly the results prove them to be an improvement over single models.

## 2.5 Other Works

[19] trained different neural networks and proved that by holding out samples for each training they could make individual networks disagree amongst themselves and this is what provided the ensemble with variation of the output for unlabeled data. They concluded that when the individual models disagree, models generalize better reducing the risk of overfitting. During their experiments the authors used a human-in-the-loop approach by measuring magnitude of disagreement as this allowed them to include samples where individuals strongly disagree thus improving the learning curve even when facing a simple test problem. This paper uses a query by committee, very much like the jury theorem in [13]. The paper con-

cluded on the idea that active learning improves the learning curve a lot for a simple test problem.

In [11] the authors use an array of methods to forecast energy consumption time series using machine learning based on usage patterns of residential householders. The authors use a mixture of single, ensemble and hybrid models to try and predict energy consumption. Amongst the models they used HWES, MA, ARIMA, and a version of SARIMAX. They concluded that simple models work great for fast estimates, ensembles work best for users who know the basics of machine learning, and hybrid models for the more experienced users.

In [5] the author explores different linear combinations and weighing benchmarks and schemes to obtain ensemble forecasts. The author found that certain degree of saturation is reached after five conceptually different forecasting models however, it is emphasized how ensembling forecasting models improves forecasting accuracy. Saturation means that when adding more single models, it does not significantly improve forecasting accuracy.

Time series analysis using an ensemble of statistical univariate time series models, multivariate models, and contemporary deep learning-based models was explored in [6]. In their paper, the authors worked from the distributor perspective (instead of endpoint retail) sales to reduce some of the unpredictable variables that consumer demand has, such as weather and the state of the economy. Their ensemble proved to reduce forecasting errors by nearly 50% of the most-sold product in their study while comparing to a statistical naïve model. The authors claim that, in their study, no single model worked for all of the products sold by the distributor; each model required an individually tuned model to significantly reduce error. The authors use the SARIMA model, the Vector Auto-regression, Long Short-Term Memory Networks, and Ensemble Models.

Forecasting sectors in the industry not only want accurate forecasts but an algorithm that is

stable in the long run. [1] proposed an averaging algorithm that penalized based on history the deviations from the actual sales. Their research and proposed algorithm is compared to existing single models, time series models and machine learning regression models alike, using two distinct forecast accuracy metrics:

- FACC:

$$FACC = 1 - \frac{\sum_{i=1}^{k}(forecast_i - actuals_i)}{\sum_{i=1}^{k} forecast_i} \tag{26}$$

- Accuracy (Named MBE in this thesis):

$$FACC = 1 - \frac{\sum_{i=1}^{k}(forecast_i - actuals_i)}{\sum_{i=1}^{k} actuals_i} \tag{27}$$

In both metrics $k$ denotes the range of the months in which the forecast accuracy is analyzed. Amidst some of the things they do is define their own methodology to produce forecasts based on their needs. The authors divided the steps into Data Processing, Creating the Validation Sets, Model Input, Model Fix and Stability test, and Forecast Generation; all of which have their inside processes. The authors were looking for low variance in their ensemble as one of the inside workings of the algorithm selected single models that specifically had low MSE. This paper proposed an ensemble using generated weights based on past deviation for each of their selected models. Their ensemble had an improvement over time series models of 3.3% and an increase over regression models of 4.3%. The authors concluded that their proposed ensemble, which took *low bias and high variance* models, and produced a *low bias and low variance* result. Their future work included the incorporation of their models into a single platform as they had to use R-Studio for the time series models and Python for the regression models.

More recently, a review of ensemble approach to forecasting was analyzed in [36]. In this

publication the authors study the effect that withdrawing from the classical single statistical or machine learning approach or single functional form. Their findings helped them conclude that ensemble forecasting generally improved forecasting accuracy and robustness in certain fields, i.e. weather forecasting. Their review mostly comprised transport related cases and in them the potential of ensemble models in improving forecast accuracy and reliability. One of their conclusions can be aptly summarised into the quote: "Ensemble forecasting cannot entirely solve uncertainties in modeling, but it gets closer to modeling real-world events...". This paper used MAE and MSE for the evaluation of the models' performance. Additionally they talk about an *Ensemble of Ensembles* which could forgo the logic behind the no size fits all rule in single models where in ensembles the reasoning would be: there is no single optimal way of combining single models. However based on the studied papers one can say it is a generality that ensemble models improve accuracy.

Other types of model such as Neuronal Networks, some hybrid models, Markov chains, and some non-gaussian mixtures models are more difficult to implement and many provide black-box solutions. The problem with black-box algorithms is that because they are not directly interpretable their application is harder to implement and understand. This poses a problem for PYMEs because they do not usually have analysts with time series understanding.

# 3 Methodology

One of the purposes of this thesis is the modification of the methodology proposed in [7] with some additions from the Cross Industry Standard Process for Data Mining (CRISP-DM), explained thoroughly in [25] with business applications. The benefit of merging both methodologies is that the tools available to the scientist of today are different than what was done at the time when Box et al. proposed the methodology.

The CRISP-DM uses a series of steps in a reiterative methodology to understand the data, analyze its contents and deploy different models to obtain a full circle project whose objective is the continued reiteration of these steps. The methodology revolves around the central idea that data, as the object of study, provides enough information about its content, format, and context to lead the analyst through the processes required to obtain good models. This cyclical nature of the CRISP-DM pairs well with the also iterative nature of the framework proposed in [7]. The step-by-step methodology follows the steps written in [25]:

1. Business Understanding: Initially the project's problem to be solved must be correctly and aptly defined. Many times the problem and solutions must be changed as the methodology encourages that, through reiteration, the basic form of the problem is dealt with. Other times, the initial formulation may not be complete or optimal. The authors describe that it is in this stage were the analyst's creativity in presenting the problem and the solution path plays a large roll in the success of the problem.

2. Data Understanding: This step is critical because understanding the data that will be used in the problem comprises, in an analogical fashion, the raw materials available to the scientist(s) and the manner that they can be used. Many times data is gathered for exogenous purposes. As this step is travelled the solution path may change direction in response.

3. Data Preparation: Following the analogy of the raw materials, data preparation is the step where these raw materials are prepared for their application. Polished databases include those in the final format for the modeling. The data may be changed in terms of form, sometimes lineal variables are transformed into non linear versions, or even converted from numerical into categorical variables. Data that is not available at the moment of taking decisions should not be included and the amount of missing values should be considered at this moment.

4. Modeling: At this stage everything up to this point is used to select the appropriate models and predictors. This step is the one that requires the most amount of knowledge in data, statistics, and algorithms to bring out the most of the data.

5. Evaluation: The evaluation part is similar to the Model Diagnostic part in [7]. The final objective of this step can be summarised in gaining the confidence that the model or models selected are reproducible and will execute validly for their implementation.

6. Deployment: The deployment step is when the results of the project are unleashed into real world environments. Many times with the objective of obtaining some sort of return on investment, the deployment of the techniques/models/methodologies generated through the project are then verified to analyze their effectiveness and evaluate the need for a restructure of the project. There are many situations where the codes are refurbished to work in production environments or graphic user interfaces.

The combined methodology's purpose is to create a methodology that combines both structures into one. Because both methodologies use an iterative nature, the new methodology intends to loop the structures at points where it is convenient to do so. This mixed iteration process tries to facilitate both methodologies by building on their strengths and skipping repeated processes. The steps of the new framework can be enumerated as such:

1. Business Understanding

2. Data Understanding/Preparation

    (a) Model Specifications

3. Modeling

    (a) Model Fitting

    (b) Model Diagnostics

4. Reiteration (If needed)

5. Deployment

This new combination of methodologies allows the user to adapt the work of [7] to a more modern approach were data wrangling can and will have an effect on the model by building the methodology's structure into a data science project following the CRISP-DM. One of the benefits of this new approach is that it makes the process leaner and it is formatted into an industry approved methodology that can be easily implemented by businesses and PYMEs.

## 3.1 Business Understanding

Recapping from the introduction, CAQ is a regional distributor for LTH batteries that uses heuristic and an integrated forecasting model that their inventory and sales system has. The company has a weekly order for restocking and daily sales information broken down by battery and with it they wish to create forecasting models following a structured time series methodology. The company has also shared the empiric information that their sales seem to be affected by weather conditions. The first formulation of the problem included the generation of linear and machine learning models to include the weather data of Queretaro city and with it create forecasting predictions. The original model data included the monthly data provided by CAQ and the task of mining the weather data. Information from January 2010 to December 2018 was obtained from [12] and the data consists of the following variables:

- Days with Hail
- Days with Fog
- Days with Storm

- Monthly Evaporation
- Max Rain 24H
- Monthly Total Rain
- Maximum Temperature
- Maximum Mean Temperature
- Monthly Mean Temperature
- Minimum Temperature
- Minimum Mean Temperature

The last variable added to the database was the monthly aggregated sales of CAQ for this same time frame. The data was processed and the results from the different machine learning models and linear regression found no model that correctly fitted the data. After the analysis resulted in rejecting the hypothesis provided by the application's expert, moving average models were used to provide an equally weighted forecast and this concluded with a better fitting forecast. Once this was proven to be true for the aggregated database and CAQ's best-seller, the reformulation of the problem statement was generated to produce the path that ensembling methods will provide better forecasts even if they are only univariate models of time series.

## 3.2 Data Understanding/Preparation

For this reformulation the databases that were needed from the company were two. The first database is the monthly aggregated data from January 2010 to December 2018. In Figure 4 the last 50 periods are shown of this first dataset. The second dataset was ordered by daily battery sales and from there, the sales were aggregated weekly as the company orders

stock every tuesday. The company makes weekly replenishment orders so it was established that the procedure would be to combine the daily sales into weekly format to create the best possible tool for them to analyze.

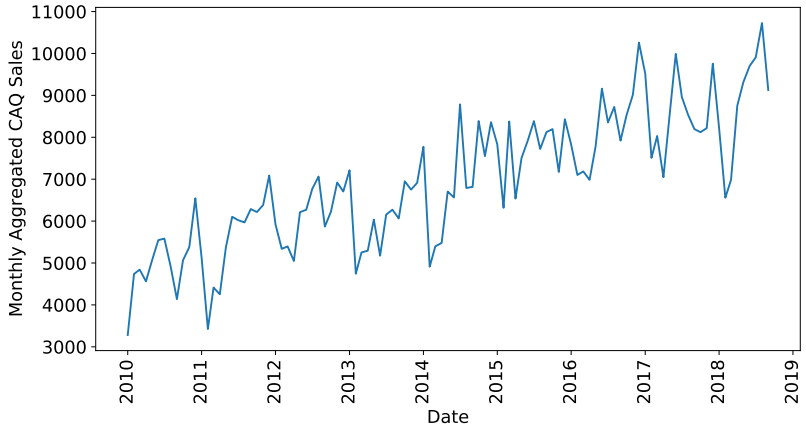


Figure 4: 2014 - 2018 CAQ Sales

## 3.3 Existing Models

In this sections the models described in sections 2.3 and 2.4 were explored and compared their performance on the aggregated and individual monthly sales datasets to evaluate the obtained results and see if any one model performs best. The traditional models were trained with their default parameters while the hybrid model SARIMAX had an additional step for optimizing its parameters in each of the training periods. To validate the necessity of an ensemble different trials were set to prove that for different databases, even amongst the same company, the best model would change. To prove this point the eight selected models were ran over the monthly aggregated sales database and its 5 top sellers, all of these databases ran in a monthly manner. Listed in the tables below, the comparisons for each of the monthly databases shows the mean absolute percentage error for a twenty month prediction period for each of the following databases: Table 4: Best Performing Models for Monthly CAQ's Aggregated Sales, Table 6: CAQ's best-seller (L-65-800), and Table 5: CAQ's 3rd best-seller.

It can be observed that every dataset has a different order on the models that performed the best. In Tables 4 and 6 SARIMAX model presented the best results while on Table 5 Prophet tested better than all of the other models. The linear combinations and majority voting algorithms performed very good when compared with traditional models following different logic for each, as described in the tables below.

Table 4: Best Performing Models for Monthly CAQ's Aggregated Sales

| Monthly CAQ Aggregated Sales | MAPE |
|---|---|
| SARIMAX | 0.080 |
| Majority-Vote Median | 0.087 |
| Mean Machine | 0.088 |
| Mean | 0.090 |
| Prophet | 0.091 |

Table 5: Monthly L-31T-900 Sales

| Monthly L-31T-900 Sales | MAPE |
|---|---|
| Prophet | 0.123 |
| SES | 0.132 |
| SARIMAX | 0.132 |
| Mean | 0.136 |
| HWES | 0.138 |

Table 6: Monthly L-65-800 Sales

| Monthly L-65-800 Sales | MAPE |
|---|---|
| SARIMAX | 0.087 |
| MinMax | 0.098 |
| Majority-Vote Median | 0.098 |
| Mean | 0.099 |
| Majority-Vote Mean | 0.100 |

These tables show how models applied to the datasets have different performance even if its during the same time period and for a similar context. Because of the distinct pattern in every battery, no single model can be used for every battery. This can be seen in Figure: 5 and in it the variations that occur for each battery.

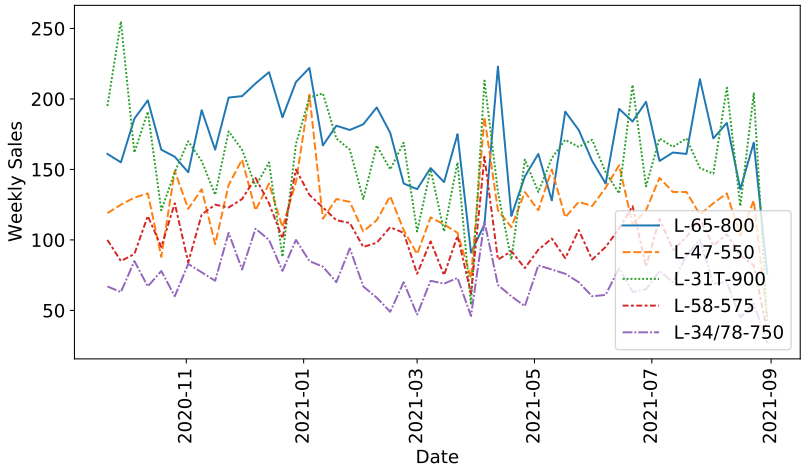


Figure 5: Weekly Sales by Model

### 3.3.1 Hierarchical Clustering

After this validation the next step for the data was the analysis and clustering of the time series using hierarchical clustering and visualizing them through dendrograms. A dendrogram is a tree diagram that is generated, amongst other uses, for hierarchical clustering. This is done by using the agglomerative clustering where using Ward variance minimization algorithm and euclidean distance to compare clusters. Once the clusters are compared it joins the next most similar clusters into a bigger one until you get to a single cluster which is also called root.

For the monthly aggregated sales an analysis of the performance of the traditional, SARIMAX, and Prophet models was done to observe if there were differences between the models. As seen in the highlighted rows it can be observed that there was a balance between the models' performance meaning that the possibility of combining them could result in a better performance. Combining the models would likely perform better as each individual model performed better in different points.

Table 7: Absolute Precision Error in Monthly Sales

| Date | MA | HWES | SES | AR | ARMA | ARIMA | SARIMAX | Prophet |
|---|---|---|---|---|---|---|---|---|
| 2017-01-31 | 0.03 | 0.27 | 0.18 | 0.18 | 0.21 | 0.22 | 0.03 | 0.05 |
| 2017-03-31 | 0.04 | 0.17 | 0.20 | 0.09 | 0.17 | 0.18 | 0.12 | 0.06 |
| 2017-04-30 | 0.25 | 0.11 | 0.04 | 0.18 | 0.10 | 0.10 | 0.01 | 0.02 |
| 2017-06-30 | 0.12 | 0.03 | 0.04 | 0.03 | 0.01 | 0.03 | 0.07 | 0.09 |
| 2017-07-31 | 0.15 | 0.04 | 0.02 | 0.01 | 0.02 | 0.03 | 0.06 | 0.02 |
| 2017-08-31 | 0.10 | 0.06 | 0.06 | 0.01 | 0.04 | 0.05 | 0.06 | 0.15 |
| 2017-12-31 | 0.00 | 0.07 | 0.06 | 0.11 | 0.10 | 0.11 | 0.11 | 0.01 |
| 2018-01-31 | 0.03 | 0.31 | 0.31 | 0.21 | 0.28 | 0.29 | 0.11 | 0.26 |
| 2018-02-28 | 0.05 | 0.09 | 0.17 | 0.05 | 0.08 | 0.08 | 0.10 | 0.29 |
| 2018-04-30 | 0.16 | 0.14 | 0.13 | 0.11 | 0.11 | 0.10 | 0.09 | 0.05 |
| 2018-06-30 | 0.19 | 0.12 | 0.13 | 0.08 | 0.09 | 0.08 | 0.04 | 0.02 |
| 2018-08-31 | 0.06 | 0.08 | 0.01 | 0.10 | 0.07 | 0.08 | 0.03 | 0.01 |

Given that this endeavour might be hard to prove for more than 200 stock-keeping units (SKU), the idea of clustering into clusters that should have similar statistics and behaviours becomes more and more plausible. If any company with these amount of SKUs can treat their products or services by grouping them would aid in time management as well as in simplicity, which according to Occam's razor is preferable. In the table 7 the necessity of ensembling models can also be observed. Following the MA model it had the worst performance on the date: 2017-06-30 but tied for best in 2018-02-28. This pattern can be seen in almost every case in the same table, models are not able to obtain the best results every single time. Combining models either by linear combinations, easy segmenting rules such as majority-vote

ruling or algorithmic ensembling will provide a better performance.

Dendrograms provide a clear image of which datasets most closely resemble each other. This is practical for many different applications, in this context it helps prove if a single forecasting model provides similar results for all of the batteries in a cluster. The first step was selecting all of the batteries that had more than 400 in sales throughout the whole time series because that led to the cleansing of a little bit more than 50% of the battery models. This filter left 121 battery models of the total database for which dendrograms were constructed. The dendrograms allow for companies, especially PYMEs, to focus projects and efforts into a category that usually allow for single methodologies to work for inside-a-category models. This means that a company can spend resources on projects to tackle a single problematic and it can be replicated easily for product or services inside a same category. This reduces the amount of projects needed to reduce, for example, inventory levels while maintaining service levels. The simplicity obtained by clustering reduces the workload and needed resources of data driven projects and allows for similarities to be drawn aiding for the faster reaching of objectives. Below the dendrograms are shown, were $(n)$ is the number of battery models in that line i.e. smaller clusters. The complete version can be seen in Appendix A.

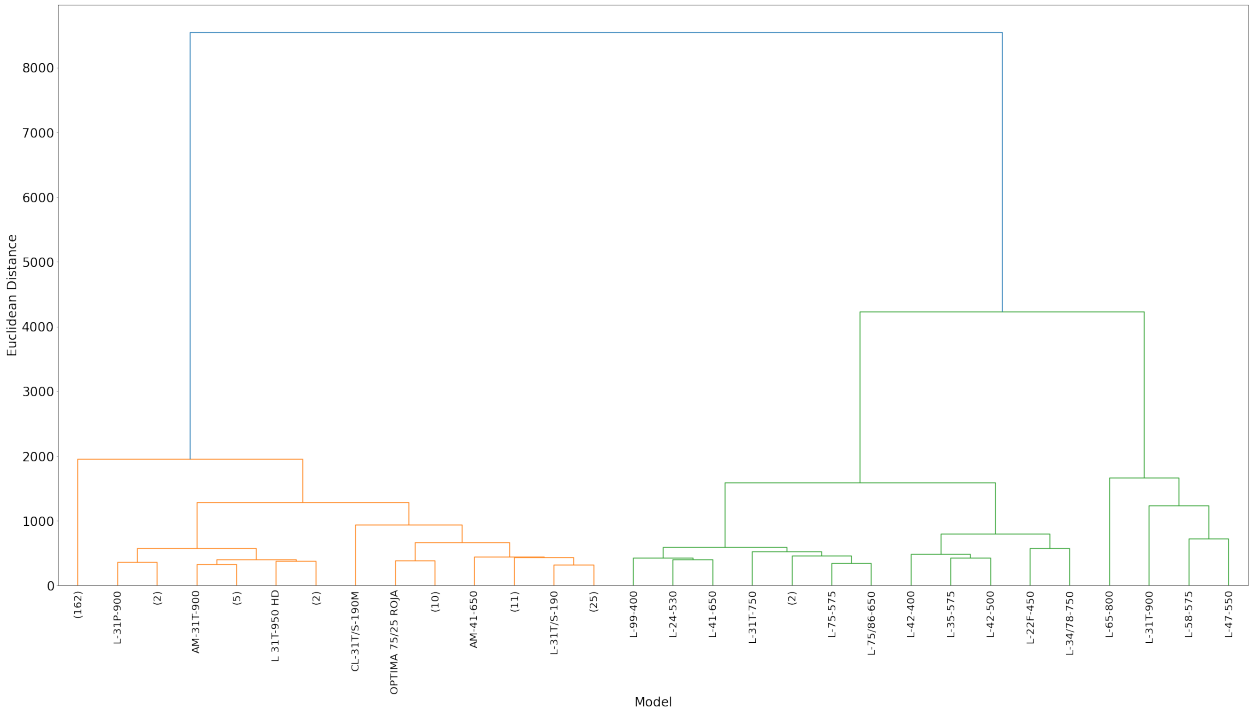


Figure 6: Clustering by Sales

In Figure:6 the batteries get clustered by magnitude of sales. The batteries that were in the same range of sales get sorted into a same cluster. Similarities in this clustering means that batteries in the same cluster have, within a range, the same magnitude. The filtered dataset was then converted from the number of sales to the rate of change week-by-week. The objective of this dendrogram is to show if there are similar behavior amongst the batteries even if their magnitudes might be different. This analysis will help conclude if a single model is best for all batteries that have similar sales patterns. This conversion was done with the following algorithm:

**if** $Week_t + Week_{t+1} \geq 1$ **then**
$\quad Week_{t+1} = \frac{Week_t + Week_{t+1}}{Week_{t+1}}$
**end**

**Algorithm 4:** Percentage Change Algorithm

Where if in algorithm 4 the denominator is 0 it would use 1 instead. With these algorithm the shape of each battery's time series can be analyzed as the shape to try and find the groups where the batteries are closely related by the Ward method and the euclidean distance metric. In Figure: 7 the categories can be seen amongst different batteries than in the first dendrogram even though, the distance between all battery models is quite far off to be statistically significant, therefore a third modification was proposed. The third alteration to this database was similar to the percentage however, in this dataset the idea was to modify the data so as to try to explain if the movement of the sales could give us further information about the batteries. The algorithm, for this modification is described in the following equations:

**if** $Week_t \leq Week_{t+1}$ **then**
| $Week_{t+1} = 1$
**else**
| **if** $Week_t \geq Week_{t+1}$ **then**
| | $Week_{t+1} = -1$
| **else**
| | $Week_{t+1} = 0$
| **end**
**end**

**Algorithm 5:** Movement Change Algorithm

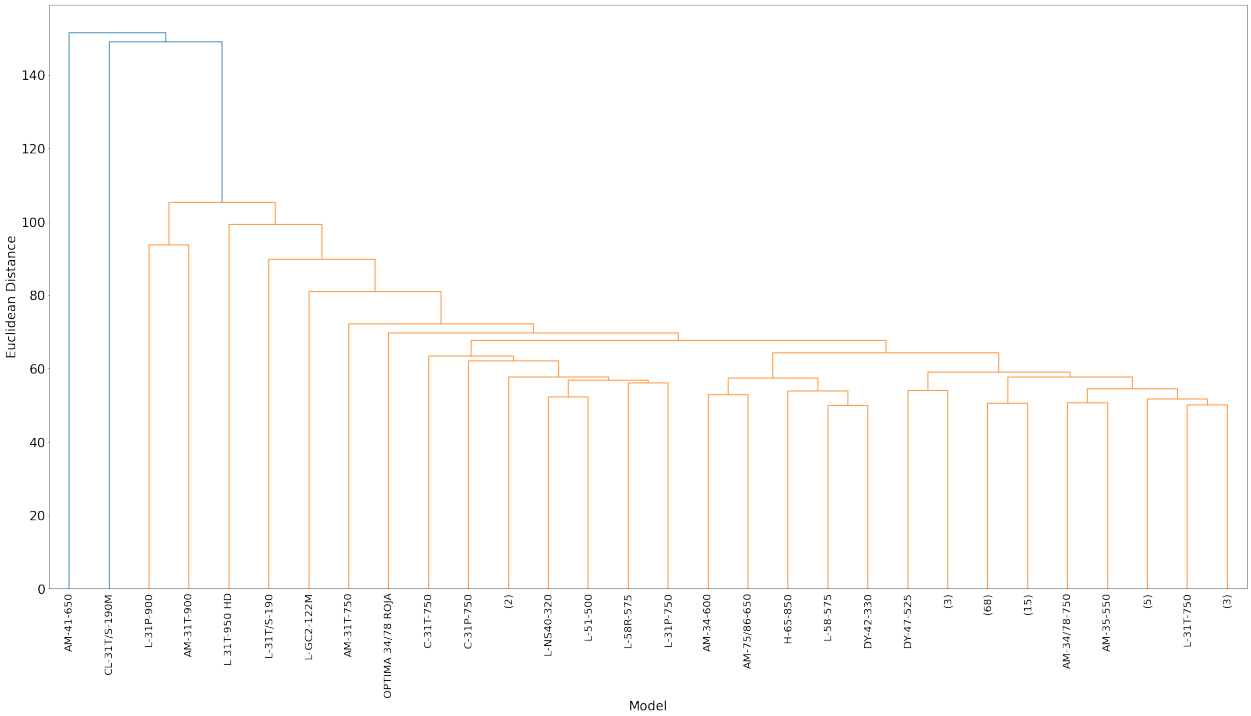


Figure 7: Clustering by Weekly Changes Percentages

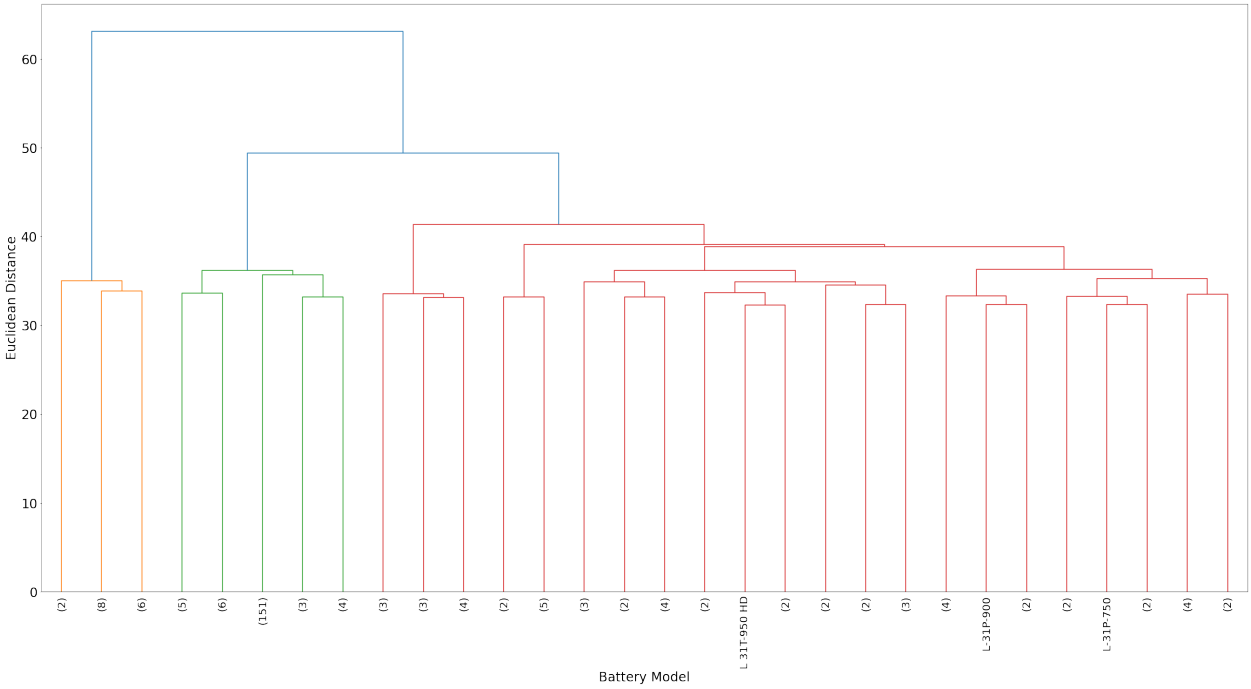


Figure 8: Clustering by Weekly Increase/Decrease

In figures 7 and 8 the differences can be seen notoriously as the groups get segmented very differently due to the adaptations to the datasets. The subtle changes in how the data is interpreted changes the analysis of the dataset, in the magnitude comparison besides amount of sales not much information is gained however, on the other datasets the groups differ significantly and provide enough information to try the different forecasting methods amongst the more diverse groups. The set obtained by the magnitude measurement differed from the other two sets so the different groups can be tested to see if any model works better in any given set.

## 3.4 Ensemble Algorithm

For this thesis the ensemble proposed works by performing SARIMAX's hyperparameter optimization, using AIC (Akaike Information Criterion) as the defining criterion, then training every other model with its default parameters using a 20 week training set size to predict the last 30 weeks in a week-to-week basis using time series cross validation. The 20 week period was obtained through a looped optimization of information gain without sacrificing computational efficiency. After the hybrid model and the rest of the models are finished training, the models are fitted and the ensemble algorithm starts combining them. This process, explained in pseudocode in 6, takes all of the training results and combines them using a weighted approach.

The idea behind combining them using this approach is to allow the ensemble to maintain certain volatility and allow each model to participate in the final forecast. Dynamic weights, meaning that every period the weights change, offer the opportunity of using different combinations without having to select models every period or, in contrast, to using the same model every period. In [2] the authors examine the performance of different weighing techniques of conceptually different forecasting models and conclude that the accuracies can be improved with many of them. In this paper they used 10 different techniques, among them the trimmed mean, an error-based measure, the least square regression, etc., to prove that most combinations work well. Given the positive results obtained the authors continued their work on [3]. In this article they present a dynamic ensemble approach that follows the past and present performance of models, using in-sample validation dataset and out-of-sample forecasts respectively, to produce the weights that will result in the final forecasts. The difference between the proposed ensemble in this thesis tho the technique in [3] is that in this thesis only present performance is considered, the metric used is different, and that the ensemble in this thesis uses each model's weight by the contribution it has to the inverse

cumulative sum of AICs. The AIC was originally defined in [4] to complement the maximum likelihood to scenarios where there are many models. AIC takes into consideration the number of factors that play a part in every model and penalizes it with the objective of getting good models that are directly applicable to the different scenarios. In the proposed algorithm 6, AIC was selected over other measurements because the risk of overfitting some of the studied models was high and because SARIMAX has the optimization loop which, if not treated correctly, could very fast become non-applicable due to its freedom to choose parameters. In [9] the authors make the comparison between AIC and BIC and they studied that most of the literature available at publication time planted the selection criterion as a Bayesian versus frequentist and that arguments for BIC in Bayesian literature that used real data recommended BIC over AIC but this conclusion was in fact flawed. Because both measurements can both be derived as frequentist or Bayesian the argument of using either one based on that factor is wrong. AIC and BIC are both good in their respective contexts but are hard to compare directly as their target objectives are different [27]. In [27] the author summarizes the goals of AIC and BIC as:

- AIC: Measurement that tries to find the smallest prediction error.
- BIC: Measurement that tries to find the model with the highest probability of having generated the data.

Based on each of the measurements goals it can be established that for the purposes of this thesis AIC is better suited for this specific job. The purpose of this measurement in this thesis is to provide a rubric based on forecasting errors for each of the models. Given that the logic for ensemble methods is to consider that all models add value to the forecasting endeavour, AIC is more appropriate in this perspective.

Using AIC makes calculating weights easily and as stated in [33] "...provide a straightforward interpretation as the probabilities of each model's being the best model in an AIC sense...".

The AIC rewards descriptive accuracy via the maximum likelihood while, as stated before, penalizes it by the amount of free parameters. In the algorithm 6 the ensembling method considers all of the forecasting models' AIC and aggregates them into a $Sum$ variable and gives each model an inverted weight, given that smaller AIC values are preferred. This inverted weight approach is one of the many ways an analyst could ensemble different models into a single forecasting value. The reason this approach was selected over other ventures is because in any given period the weights will change with complete disregard of past performance, i.e. every period only the model's performance on the training set is considered.

**Data:** Trained Models Forecasts: $Forecast_{models}$

**Result:** Weighted Combination of Models Forecast

$AIC_{models}, Inv_{models}, Weight_{models}$ = Series of AIC by model is initialized with $\emptyset$

$Sum, Ensemble = 0$

**for** *model* ***in*** *models* **do**

$\quad AIC_{model} = 2k - 2\ln(\hat{\mathcal{L}})$

**end**

$AIC_{models} = AIC_{models} + |\arg\min(AIC_{models})| + 1$

$Sum = \sum AIC_{models}$

$Inv_{models} = \frac{1}{Sum}$

$Sum = \sum Inv_{models}$

$Weight_{models} = \frac{Inv_{models}}{Sum}$

$Ensemble = \sum Weight_{models} * Forecast_{models}$

**Algorithm 6:** Proposed Ensemble Algorithm

The last model developed for this thesis is the Dynamic Best model that selects whichever model received the lowest AIC score for every period. This idea stemmed from dynamic programming as every state/period it calculates each AIC score for every model where its inputs and errors are considered in the next Dynamic Best model step.

# 4 Results and Discussion

In this section the results obtained from applying the combined framework can be seen. The performance of the proposed ensemble forecast algorithm is observed to be better than single forecasting methods. Using different loss functions for forecasting evaluation, including MBE, MSE, and MPE, it can be seen that the proposed ensemble outperforms all of the different single models, linear combinations, and majority-vote algorithms.

## 4.1 Results

The first step was running the program that created the forecasts for all of the models in a period-by-period manner using the dataset that was comprised by weekly sales. The last 30 periods, i.e. weeks, were simulated as if each week you generated a forecast for the upcoming week. This methodology was encouraged by what was found in the previous sections as well as the talks with the company's director. When updating the dataset each period the forecasts became less likely to give smooth curves as the ones when trying to predict the next 30 periods without the dataset update. The forecasts for the company's most popular battery model can be seen in 9 and 10.

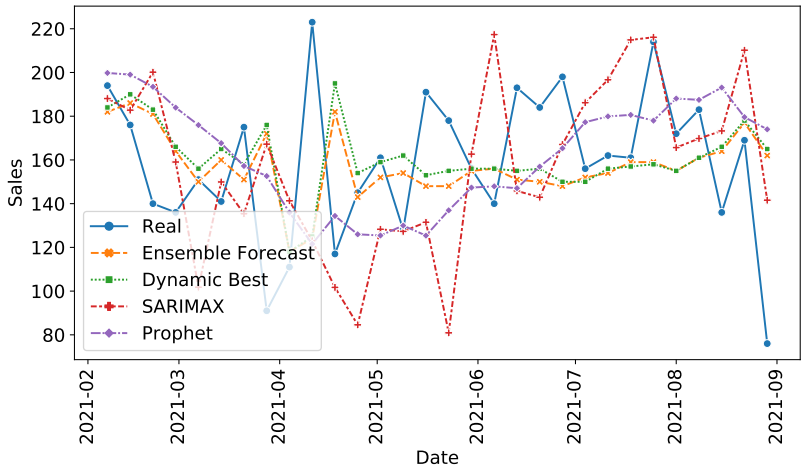


Figure 9: State of the Art Models for the L-65-800

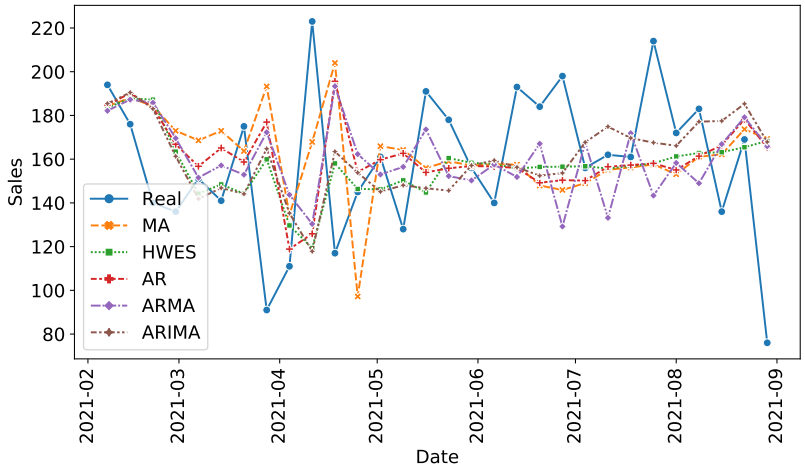


Figure 10: Traditional Models for the L-65-800

## 4.2 Bias Evaluation

The first thing the model needs to be evaluated on is on bias. To evaluate bias a good indicator is the average mean bias error can be seen in the Table: 8. The mean bias allows the analysis of a bias to be found in each of our models. This error, when paired with additional errors that are strong at finding variance, is a good place to start when analyzing forecasting errors. [22] recommends using this type of error to understand bias in models. The expected value when using MBE is that it returns a value close to zero as it would mean that it the model is not producing over or under values in the forecast. If the number deviates more from zero, for larger lead times and forecast estimations, then the model is not accurately reflecting the underlying changes of the time series. As seen in Table: 8 the only model that had a positive bias, that could indicate trend, was the proposed Ensemble Forecast. This means that on average the rest of the models had a negative forecasting error. The Ensemble Forecast was also the model with the lowest Average MBE which can be explained by its ability to use all of the models and select whichever was the better fitting models for each period.

in the Table: 9 the mean bias error for each battery can be found by model. It can also be

Table 8: Mean Bias Error by Model

| Forecast Models | MBE |
|---|---|
| Ensemble Forecast | 1.072 |
| HWES | -1.638 |
| ARIMA | -1.756 |
| SARIMAX | -2.098 |
| ARMA | -2.336 |
| Dynamic Best | -2.542 |
| Median | -2.574 |
| Olympics | -2.666 |
| Mean | -2.690 |
| MinMax | -2.772 |
| AR | -2.916 |
| Prophet | -2.990 |
| MA | -3.528 |
| SES | -4.280 |

observed how the Ensemble Forecast ranked in absolute values in high rankings most of the time. This means that the Ensemble Forecast was able to better grasp the trend and that its errors where not as big as other models. The MBE is a very easy to read error as it's units are in the same form as the forecast. Analyzing the Ensemble Forecast we can see how in the first battery type it missed by 4.83 batteries each week while in its best performance, battery type L-34/78-750, it missed by 0 batteries on average. This means that the result at the end of the 30 week period, if there are no missed sales and inventory costs allow for a policy to hold stock over non-sold batteries, would be 0 batteries on inventory.

Table 9: Mean Bias Error by Battery Model

| Forecast Model | L-31T-900 | L-34/78-750 | L-47-550 | L-58-575 | L-65-800 |
|---|---|---|---|---|---|
| Ensemble Forecast | 4.83 | 0.00 | 0.07 | -1.17 | 1.63 |
| ARIMA | -1.50 | -0.94 | -3.58 | -1.14 | -1.62 |
| Prophet | -2.80 | -2.28 | -3.54 | -2.67 | -3.66 |
| ARMA | -0.24 | -1.02 | -5.65 | -2.88 | -1.89 |
| HWES | -1.22 | -1.97 | -3.35 | -3.34 | 1.69 |
| Olympics | -0.89 | -2.91 | -3.83 | -3.78 | -1.92 |
| Median | -0.41 | -2.92 | -3.69 | -3.80 | -2.05 |
| Mean | -0.18 | -2.79 | -4.35 | -4.06 | -2.07 |
| MinMax | 1.94 | -2.45 | -5.91 | -4.93 | -2.51 |
| MA | 1.01 | -4.68 | -4.69 | -5.20 | -4.08 |
| SES | -2.74 | -3.84 | -3.37 | -5.27 | -6.18 |
| Dynamic Best | 2.23 | -3.87 | -3.40 | -5.67 | -2.00 |
| AR | 1.69 | -4.17 | -3.85 | -5.94 | -2.31 |
| SARIMAX | 4.32 | -3.44 | -6.79 | -6.08 | 1.50 |

Below on Figure: 11 it is apparent that most models had values over and under 0 which is a positive thing for forecasting models as it evidences the model's capacity to integrate new data into its forecasting. As a comparison to the Ensemble Forecast, the Prophet model continuously had a very small variance in bias but being that all of the averages of each 30 week where under 0 could tell us that it was not very good at forecasting the changes in trend. SARIMAX had the largest biases of all the models which could indicate that in this scenario it lost some of its forecasting efficiency over shorter time frames and seasonalities. The violinplot also shows the distribution of each of the forecasting models and in it we

can observe that the ARIMA model had a very good performance, in respect to its low bias variability forecasts, which can be used when proposing or designing new ensembling methods.

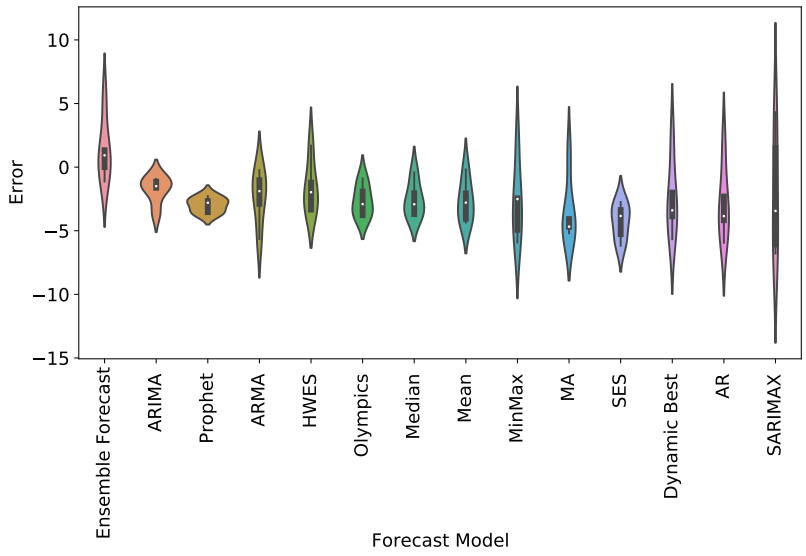


Figure 11: Dispersion of MBE by Model

## 4.3 Variance Evaluation

The second step to evaluate the models' fit is to analyze the variance. This is done by analyzing the MSE of the models. In the Table: 10 the results for each battery dataset is shown. There is no point in aggregating the data here as the magnitudes of each dataset are somewhat different therefore squared errors does not show accurately how each model behaves individually as outliers are heavily penalized. The MSE shows that the overall variance of the models is not big and paired with the bias evaluation the conclusion that the forecast models for this application performed good. The variance is between what can be expected for models in this time frame and show that the predictions are inline with the real results.

Table 10: Mean Square Error by Battery, Models Better than Ensemble Forecast Highlighted

| Forecast Model | L-31T-900 | L-34/78-750 | L-47-550 | L-58-575 | L-65-800 |
|---|---|---|---|---|---|
| Ensemble Forecast | 1716.17 | 354.27 | 721.80 | 590.17 | 1487.23 |
| Dynamic Best | 1657.17 | 380.80 | 753.80 | 652.27 | 1593.27 |
| Olympics | 1744.62 | 334.04 | 747.78 | 563.62 | 1435.07 |
| MinMax | 2012.00 | 371.59 | 833.64 | 666.97 | 1468.34 |
| Mean | 1772.63 | 330.30 | 754.28 | 577.38 | 1421.27 |
| Median | 1657.48 | 334.05 | 738.89 | 557.44 | 1424.76 |
| SES | 1833.39 | 338.03 | 686.77 | 523.48 | 1323.16 |
| HWES | 1851.48 | 347.00 | 772.28 | 565.91 | 1404.76 |
| MA | 1680.45 | 385.94 | 832.59 | 696.76 | 1699.54 |
| AR | 1658.27 | 374.32 | 762.14 | 675.79 | 1610.27 |
| ARMA | 1707.97 | 363.63 | 897.74 | 664.01 | 1735.81 |
| ARIMA | 2118.81 | 348.35 | 763.29 | 545.23 | 1485.25 |
| Prophet | 2104.37 | 416.85 | 786.31 | 615.57 | 1658.63 |
| SARIMAX | 3509.42 | 739.52 | 1311.43 | 1269.16 | 2215.35 |

As seen in Table 10, The top performers in terms of variance were the ensemble forecast and the linear combinations. This means that ensembles are able to handle variance better and in these cases the results were evident by a lot. Except for the SES and ARIMA, no single model beat the proposed ensemble 2 times.

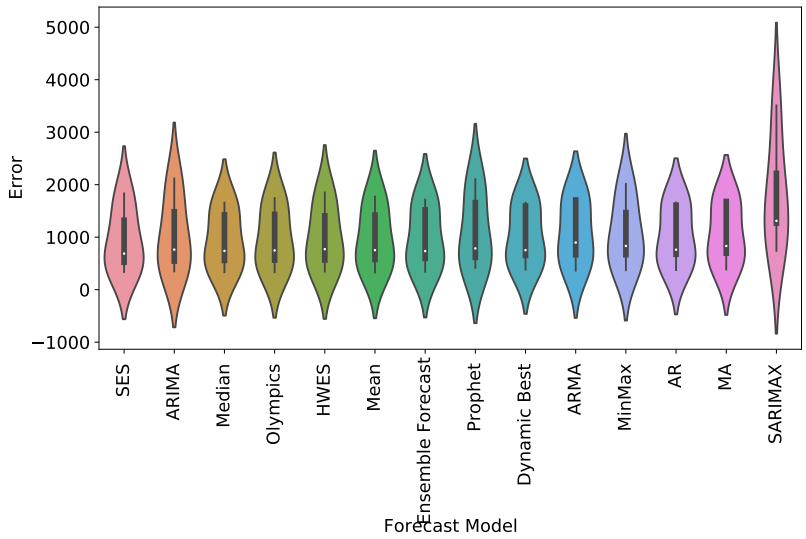


Figure 12: Dispersion of MSE by Model

## 4.4 Relative Forecast Error Evaluation

The third step is to convert the error into a relative one so that the error values are not scale dependent. To analyze relative errors MPE is a great error measurement as it is easily interpretable and the size of the measurement can easily be converted into expected units, i.e. the time series' units. In Table: 11 the difference between the Ensemble Forecast and some of the others linear combinations are insignificant. The Ensemble Forecast differs by a 2.4% to the ARIMA Forecast but by a 4.8% margin to the Simple Exponential Smoothing Forecast. This table shows the average difference amongst all of the batteries, Table: 12 shows the actual model's performance in each of the batteries datasets. The table shows how the Ensemble Forecast is consistently in the top performers for each dataset. In that table it can also be observed that for each dataset different models perform better for example, in the battery model with bigger sales, L-65-800, SARIMAX performed better than even the Ensemble Forecast however it's misses were higher thus when aggregating the performance over different datasets it is heavily penalized. That volatility is one of the reasons why ensembling, throughout its uses, is being studied so much. Ensembling methods allow the

flexibility of choosing different models at different periods and generally resulting in better models. In this application the relative error for ensembling was lower than the rest of the models, including the linear combinations and the Dynamic Best approach.

Table 11: Mean Percentage Error by Model

| Model | MPE |
|---|---|
| Ensemble Forecast | -0.076 |
| ARIMA | -0.098 |
| SARIMAX | -0.100 |
| HWES | -0.106 |
| ARMA | -0.106 |
| Median | -0.108 |
| MinMax | -0.108 |
| Mean | -0.110 |
| Olympics | -0.110 |
| Prophet | -0.114 |
| Dynamic Best | -0.114 |
| AR | -0.118 |
| MA | -0.122 |
| SES | -0.124 |

Table 12: Mean Percentage Error by Battery, Models Better than Ensemble Forecast Highlighted

| Forecast Forecast | L-31T-900 | L-34/78-750 | L-47-550 | L-58-575 | L-65-800 |
|---|---|---|---|---|---|
| Ensemble Forecast | -0.10 | -0.08 | -0.07 | -0.08 | -0.05 |
| Dynamic Best | -0.12 | -0.14 | -0.10 | -0.13 | -0.08 |
| Olympics | -0.15 | -0.12 | -0.10 | -0.11 | -0.07 |
| MinMax | -0.12 | -0.11 | -0.12 | -0.12 | -0.07 |
| Mean | -0.14 | -0.12 | -0.11 | -0.11 | -0.07 |
| Median | -0.14 | -0.12 | -0.10 | -0.11 | -0.07 |
| SES | -0.17 | -0.14 | -0.09 | -0.12 | -0.10 |
| HWES | -0.16 | -0.11 | -0.10 | -0.11 | -0.05 |
| MA | -0.13 | -0.15 | -0.11 | -0.13 | -0.09 |
| AR | -0.13 | -0.14 | -0.10 | -0.14 | -0.08 |
| ARMA | -0.14 | -0.09 | -0.12 | -0.10 | -0.08 |
| ARIMA | -0.15 | -0.09 | -0.10 | -0.08 | -0.07 |
| Prophet | -0.17 | -0.12 | -0.10 | -0.10 | -0.08 |
| SARIMAX | -0.08 | -0.12 | -0.13 | -0.13 | -0.04 |

In the Table 12 the results are even more evident than in the variance and bias measurement. By database there were only 2 instances were a model performed better than the Ensemble Forecast and 2 models that tied it. Across the results it can be seen that these clearly state how the proposed model outperforms every other model and it also is a small percentage error.

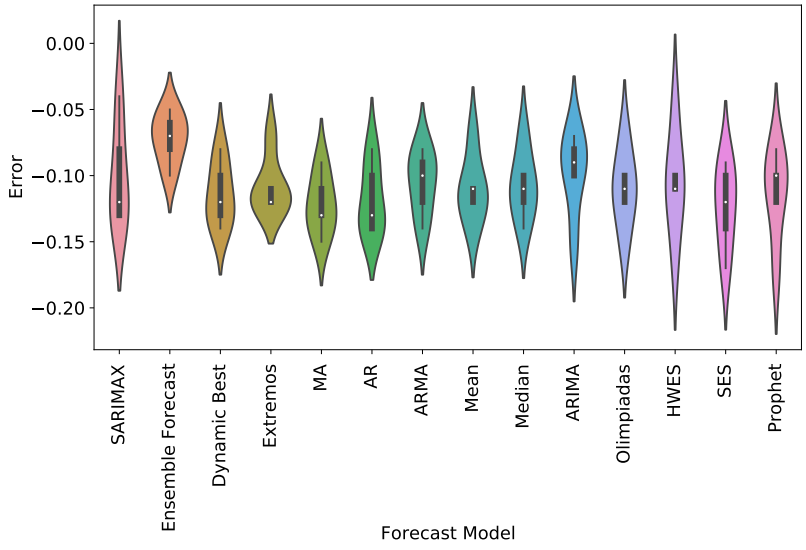


Figure 13: Dispersion of MPE by Model

## 4.5 AIC Against Other Criterions

In this subsection the working differences between criterions are explored. The search for other criterions stems from the possibility that other criterions might work better as the proposed ensemble is not infallible. The criterions selected for further study are the following:

- Akaike Information Criterion: Measures the overall model quality based on relative information loss.

$$AIC = 2k - 2ln(\hat{\mathcal{L}}) \tag{28}$$

- Bayesian Information Criterion: Even though they might look similar, once again, the BIC's objective is to find the model with the highest probability of having generated the data.

$$BIC = ln(n)k - 2ln(\hat{\mathcal{L}}) \tag{29}$$

- Logarithmic AIC: This measurement was intended to reduce the differences between

weights in each categories by diminishing the total model sum. This part was implemented in the code as it also included the addition of a constant (2). This change can be seen in Algorithm: 7.

**Data:** Trained Models Forecasts: $Forecast_{models}$

**Result:** Weighted Combination of Models Forecast

$AIC_{models}, Inv_{models}, Weight_{models}$ = Series of AIC by model is initialized with $\emptyset$

$Sum, Ensemble = 0$

**for** *model* ***in*** *models* **do**

$AIC_{model} = 2k - 2\ln(\hat{\mathcal{L}})$

**end**

$AIC_{models} = ln(AIC_{models} + |\arg\min(AIC_{models})| + 2)$

$Sum = \sum AIC_{models}$

$Inv_{models} = \frac{1}{Sum}$

$Sum = \sum Inv_{models}$

$Weight_{models} = \frac{Inv_{models}}{Sum}$

$Ensemble = \sum Weight_{models} * Forecast_{models}$

**Algorithm 7:** Proposed Ensemble Algorithm with Log AIC

The difference between this algorithm and Algorithm: 6 are that this algorithm uses a constant of 2 instead of 1 and the logarithm of the AIC value is used to calculate the total sum of the models. This subtle change did have an effect on the forecasting values predicted. The differences amongst these criterions will be analyzed the same way the difference between models were analyzed.

### 4.5.1 Bias Evaluation

Following the same analysis as in the model comparison, the first evaluation represents bias. In Table: 13 it can be seen that the smallest bias is the original AIC formulation while BIC

obtained the second spot and finally the AIC Log. The values do not differ much amongst themselves thus, for the purpose of ensembling it is clear that the criterion does not have a huge impact on the bias performance of the ensemble. All of the methods however were an improvement over the single models. The errors shown in the table can also be translated as the prediction error of 1 battery per day.

Table 13: Mean Bias Error by Criterion

| Model | MBE |
|---|---|
| Ensemble Forecast AIC | 1.072 |
| Ensemble Forecast BIC | 1.158 |
| Ensemble Forecast AIC Log | 1.160 |

### 4.5.2 Variance Evaluation

The variance evaluation showed a similar result to the bias evaluation. All 3 of the criterions behaved in a similar fashion. At least every criterion obtained the best result in a battery model which means that all criterions have applicability in these scenarios. The AIC Log did present a better ranking as it won in 3 of the 5 categories although the variation between each was not considerable.

Table 14: Mean Square Error by Criterion, Best Criterion Highlighted

| Model | L-31T-900 | L-34/78-750 | L-47-550 | L-58-575 | L-65-800 |
|---|---|---|---|---|---|
| Ensemble Forecast AIC | 1716.17 | 354.27 | 721.80 | 590.17 | 1487.23 |
| Ensemble Forecast AIC Log | 1798.27 | 329.80 | 734.20 | 563.23 | 1431.30 |
| Ensemble Forecast BIC | 1724.60 | 351.23 | 717.33 | 595.50 | 1483.47 |

### 4.5.3 Relative Forecast Error Evaluation

The relative forecast error showed that the criterions slightly varied performance. This small error (0.4%) could be attributed to a smaller volatility of the AIC Log ensemble. The forecast error seen by each ensemble criterion was better than any of the single models and other combination ensembles and this goes to show that having a mathematically logical ensemble can improve forecast accuracy.

Table 15: Mean Percentage Error by Criterion

| Model | MPE |
|---|---|
| Ensemble Forecast AIC Log | -0.072 |
| Ensemble Forecast AIC | -0.076 |
| Ensemble Forecast BIC | -0.076 |

### 4.5.4 Criterion Selection Conclussion

The overall good performance of all criterions in the ensembles was expected as they are mathematically similar despite the fact that AIC and BIC have different objectives. The criterion selected in the thesis, AIC, performs as good as BIC and AIC Log and, because of the objectives of each criterion and the objective of obtaining good forecasts that maintains some volatility, the original ensemble model is accepted. BIC could be used for the Dynamic Best model because its objective is more in line with the use inside this thesis although ensemble worked better than the Dynamic Best and did not increase computational costs by a perceptible amount.

# 5 Conclusion

Forecasting is an important part not only in business settings, but in many sciences and applications where having a structured and modern forecasting methodology can result in better overall performance. Ensemble forecasting has its many advantages over the selection of single models not only in the forecasting accuracy department but in the flexibility of types of data that a user can utilize. Accurate forecasting can lead to better key performance indicators and additional resources for the decision makers to use. The methodology followed and the algorithm selected for ensemble forecasting aids in the regular challenges faced when venturing into forecasting. Because of its nature, forecasting is an expensive tool that in many cases, especially for PYMEs, can be unsustainable. Having accurate forecasts allows users to control, in business settings, their inventory levels and thus have control over cashflow and inventorying costs. PYMEs are heavily reliant on product/services experts that have little to no forecasting training and because of it, withdraw from many quantitative forecasting methods.

Because ensembling methods have provided strong results in machine learning applications, the search for a similar method within a time series context seemed to provide a good place to start when tackling better forecasts. Ensembling methods are also one of the most straightforward ways to combine different models to preserve their strengths and reducing their, usually big, misses. Given that linear combinations performed well in monthly aggregated data, the search for additional ensembling methods was justified and the idea of using dynamic weights allowed the forecasts to maintain their flexibility. Using different loss functions to ensemble and verify forecast accuracy this thesis proved that dynamic combinations worked best over static weights and rules of thumb.

Linear combinations provided a good starting point for forecasts as they sometimes improved single models performance. The ways to linearly combine these models are limited to the

analysts creativity however, only the ones specified in this thesis have been proven to work in CAQ's context. These combinations maintain some of the volatility of single models but their errors are mitigated by being central tendency inclined. This mixture has created good forecasts and can be the best model in some of the instances.

One of the difficulties or issues presented by using the algorithm developed in this thesis is that the user assumes the correct hyperparameters in the models. This presents the challenge of working behind a black box algorithm, unless the user has knowledge on time-series analysis and forecasting, and the utter trust on the final forecast. Unlike individual models the break down of the components in this model requires the further exploration of the individual models to understand how and why a certain point is forecasted. Another challenge presented when working with this model is that it does require more computational power and statistical knowledge from the user. This ultimately means that fine-tuning of models or an exchange of individual models can only be done by someone knowledgable and with more time than, for example, someone that will use exclusively moving average modeling.

State-of-the-art models such as the prophet or the SARIMAX have proven to be quite powerful in many applications however their volatility opens the confidence intervals and in some cases one-time mistakes are enough to cost them a good performance. These models are in some cases the most precise however they are not the most accurate which come with its own strings of problems.

The AIC works nicely as a weight estimator for the Ensemble Forecast. This statistic proved to successfully assign weights throughout the iterations as it was the most accurate and in the top 2 precise forecasts. The results obtained with this Ensemble Forecast model make it clear that powerful tools can be developed for forecasting and a machine learning crossover with time series produces good models and consequently good results.

# 6 Future Work

Further work could be done in many different areas as this thesis opens the door to many possible branches. Comparing an exclusively univariate ensemble with a multivariate ensemble would provide deep insights into seeing if the added computational cost benefits the forecasting errors. There are many models being developed that can handle such endeavours, e.g. the Vector Auto-regression, and with it the possibility of also combining both kinds of models into a mixed ensemble to compare the results amongst them. The analysis of this ensemble outside supply chain applications could also be explored for instance the financial markets present a unique opportunity as it is sometimes considered a benchmark when deploying new time series forecasting models. Another possibility would be creating the interface of the framework explained throughout this thesis for which any user could simply load their time series and obtain an accurate forecast. This tool could aid PYMEs and big businesses alike, as most forecasting experts use different coding languages to deploy forecasting models.

Another possible exploration of the performance can be the analysis and optimization of additional criterions inside the ensembling algorithm as well as combining the single models using a different mathematical logic. These analysis could aid in the creation of an even more powerful, in terms of accuracy and loss function results, forecasting method. Criterions and loss functions could also be different filters to the models that will be considered each round to work around some of the outliers found in throughout CAQ's databases. The ensemble methodology created for this thesis must be studied against other ensembles proposed by the analyzed literature to see if the improvement over single models can be transposed outside these databases.

Using CAQ's context, a transfer of information gained in this thesis would be the next step for them to analyze and understand that individual battery models can not be handled

by individual treatments and that frameworks, such as the proposed one, are better at forecasting than the usual "One model fits all"single model technique. This means that single model strategies should not be used as well as establishing individual inventory policies. Even with huge amounts of products the framework defined in this thesis should aid in the selection of these individual policies as it was found that even if battery models have the same magnitude or the same shape the treatment of these models should be unique to its specifications to obtain the most accurate results. The company is to benefit if it implements the methodology described in the thesis and the forecast ensemble and from what it is seen in the literature it will benefit even if using faster/easier single models as the combinations tried in the literature do improve forecasting accuracy.

# Appendix A

These dendrograms represent the distinct clusters that were generated when processing the weekly sales database generated by CAQ. The dendrograms appear in the same order as in 3.3.1 but are enlarged as to include the batteries that were not significant in the truncated version.

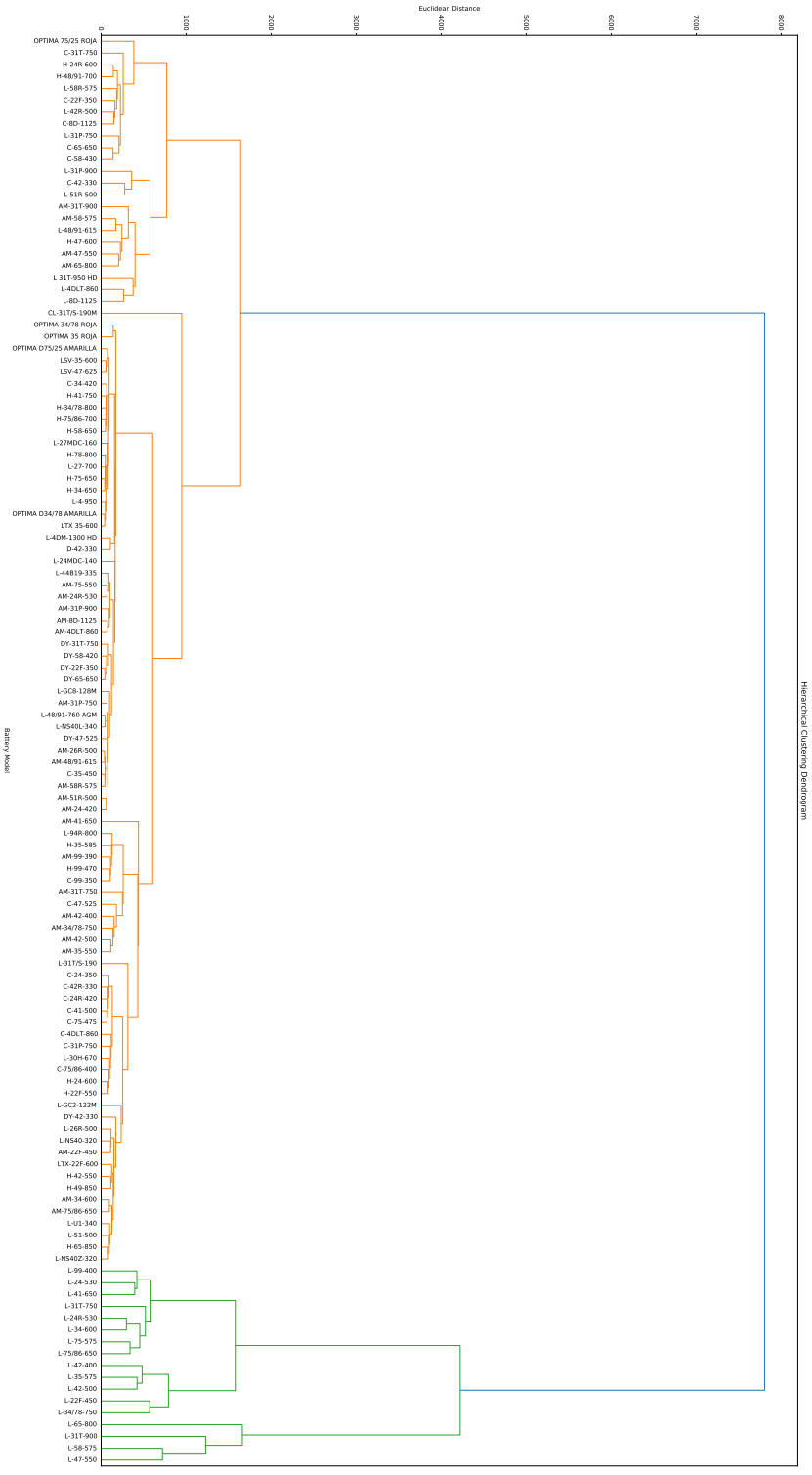


Figure 14: Clustering by Sales

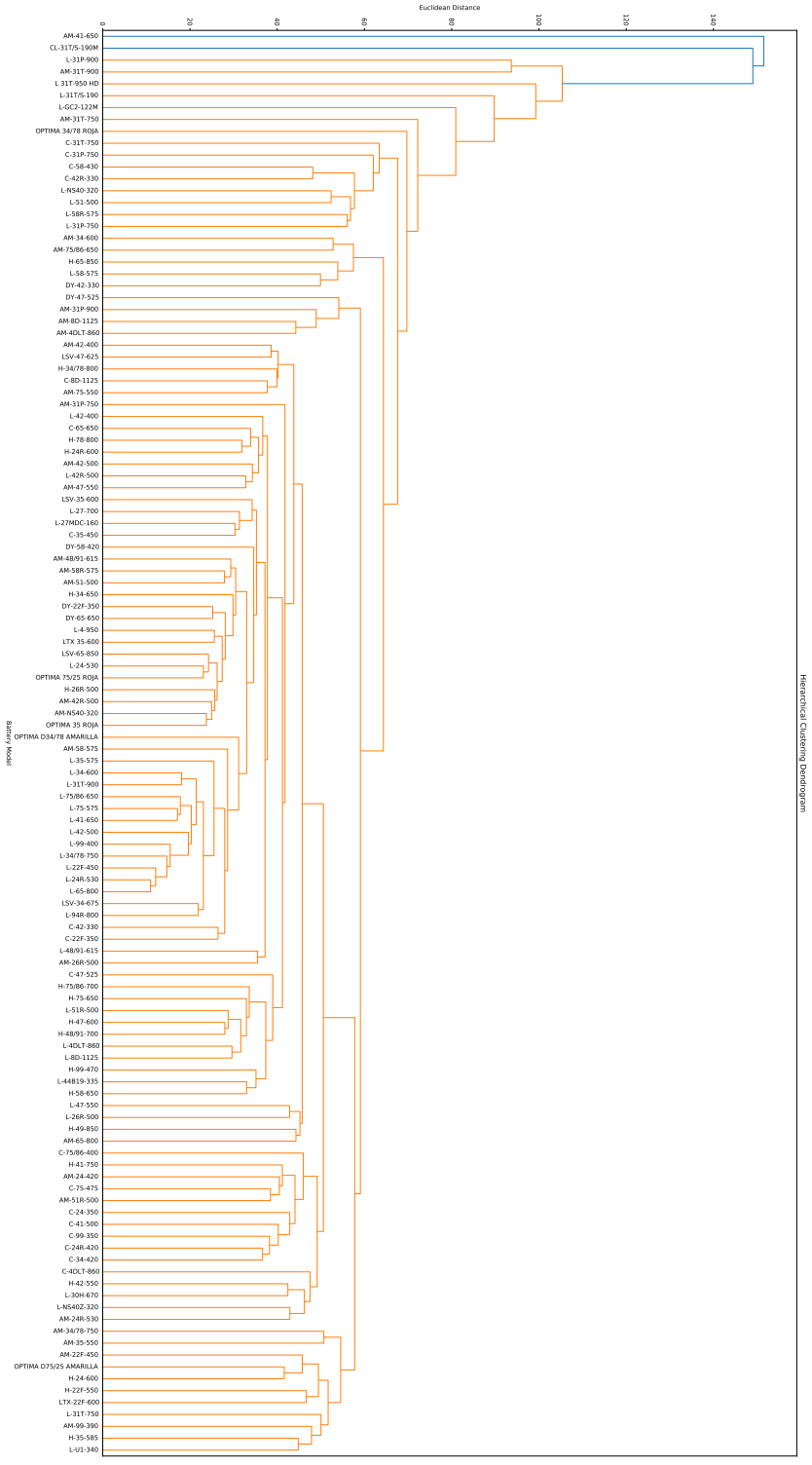


Figure 15: Clustering by Weekly Changes Percentages

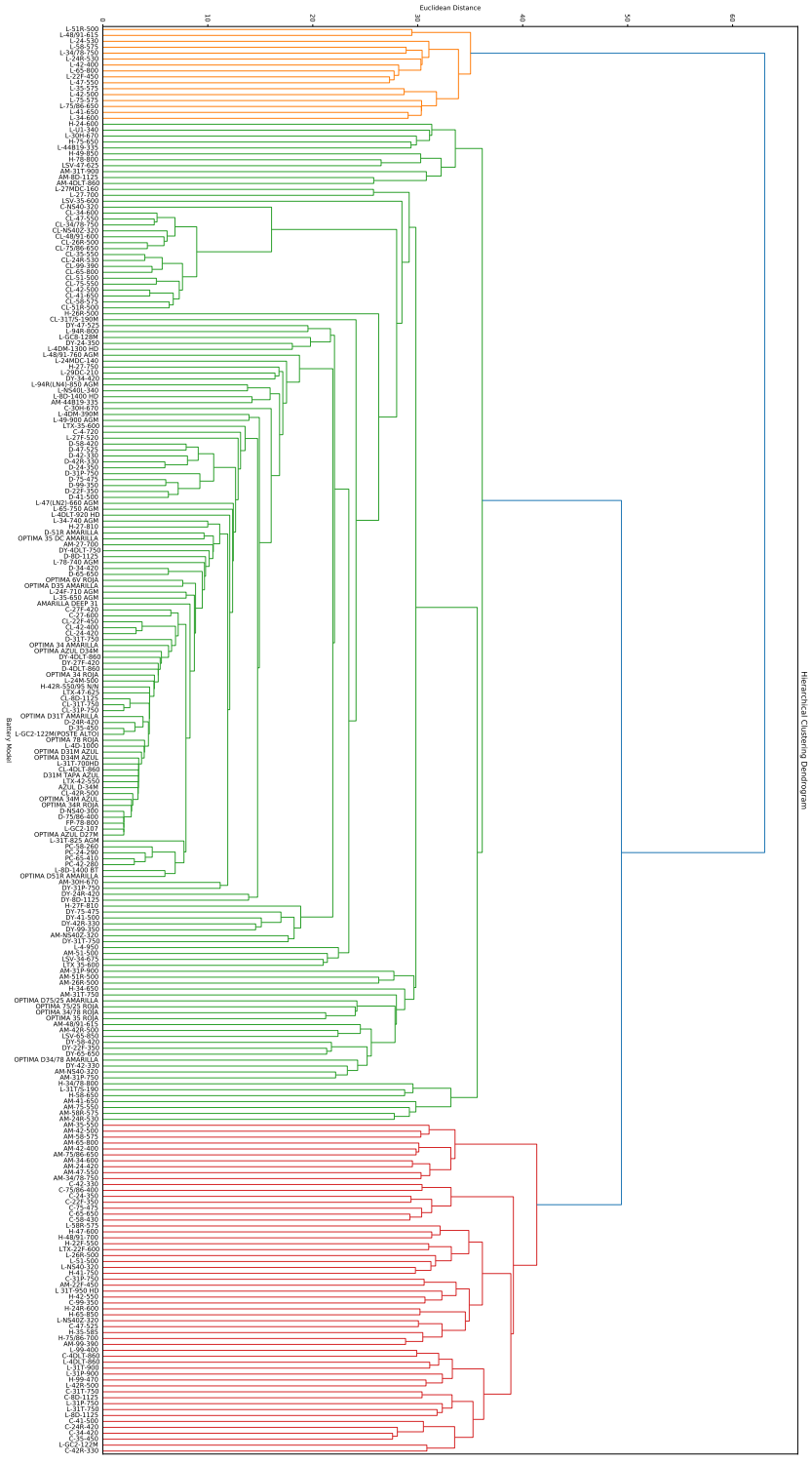


Figure 16: Clustering by Weekly Increase/Decrease

# References


[1] Nimai Chand Das Adhikari et al. "Ensemble methodology for demand forecasting". In: *2017 International Conference on Intelligent Sustainable Systems (ICISS)*. IEEE. 2017, pp. 846–851.

[2] Ratnadip Adhikari and RK Agrawal. "Performance evaluation of weights selection schemes for linear combination of multiple forecasts". In: *Artificial Intelligence Review* 42.4 (2014), pp. 529–548.

[3] Ratnadip Adhikari and Ghanshyam Verma. "Time series forecasting through a dynamic weighted ensemble approach". In: *Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics*. Springer. 2016, pp. 455–465.

[4] Hirotugu Akaike. "Factor analysis and AIC". In: *Selected papers of hirotugu akaike*. Springer, 1987, pp. 371–386.

[5] Jon Scott Armstrong. *Principles of forecasting: a handbook for researchers and practitioners*. Vol. 30. Springer, 2001.

[6] Tanvi Arora et al. "Demand Forecasting In Wholesale Alcohol Distribution: An Ensemble Approach". In: *SMU Data Science Review* 3.1 (2020), p. 7.

[7] George EP Box et al. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[8] Leo Breiman. "Bagging predictors". In: *Machine learning* 24.2 (1996), pp. 123–140.

[9] Kenneth P Burnham and David R Anderson. "Multimodel inference: understanding AIC and BIC in model selection". In: *Sociological methods & research* 33.2 (2004), pp. 261–304.

[10] Chirag Chadha. *Bagging, Boosting, and Gradient Boosting*. Jan. 2020. URL: `https://towardsdatascience.com/bagging-boosting-and-gradient-boosting-1a8f135a5f4e`.

[11] Jui-Sheng Chou and Duc-Son Tran. "Forecasting energy consumption time series using machine learning techniques based on usage patterns of residential householders". In: *Energy* 165 (2018), pp. 709–726.

[12] CONAGUA and Servicio Meteorológico Nacional. URL: https://smn.conagua.gob.mx/es/?option=com_content&view=article&id=177:queretaro&catid=14.

[13] Marquis JA Condorcet and Marquis de Caritat. "Sur les elections par scrutiny". In: *Histoire de l'Academie Royale des Sciences* (1781), pp. 31–34.

[14] James Douglas Hamilton. *Time series analysis*. Princeton university press, 2020.

[15] Charles C Holt. "Forecasting seasonals and trends by exponentially weighted moving averages". In: *International journal of forecasting* 20.1 (2004), pp. 5–10.

[16] Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

[17] Rob J Hyndman and Anne B Koehler. "Another look at measures of forecast accuracy". In: *International journal of forecasting* 22.4 (2006), pp. 679–688.

[18] Nikolaos Kourentzes, Juan R Trapero, and Devon K Barrow. "Optimising forecasting models for inventory planning". In: *International Journal of Production Economics* 225 (2020), p. 107597.

[19] Anders Krogh, Jesper Vedelsby, et al. "Neural network ensembles, cross validation, and active learning". In: *Advances in neural information processing systems* 7 (1995), pp. 231–238.

[20] Tae-Hwy Lee. "Loss functions in time series forecasting". In: *International encyclopedia of the social sciences* (2008), pp. 495–502.

[21] Fernando López. *Ensemble Learning: Bagging & Boosting*. Jan. 2021. URL: https://towardsdatascience.com/ensemble-learning-bagging-boosting-3098079e5422.

[22] Douglas C Montgomery, Cheryl L Jennings, and Murat Kulahci. *Introduction to time series analysis and forecasting*. John Wiley & Sons, 2015.

[23] David Opitz and Richard Maclin. "Popular ensemble methods: An empirical study". In: *Journal of artificial intelligence research* 11 (1999), pp. 169–198.

[24] Juan M Ortiz de Zarate. 2021. URL: https://www.toptal.com/machine-learning/ensemble-methods-kaggle-machine-learn.

[25] Foster Provost and Tom Fawcett. *Data Science for Business: What you need to know about data mining and data-analytic thinking.* " O'Reilly Media, Inc.", 2013.

[26] Deepa Puthran et al. "Comparing SARIMA and Holt-Winters' forecasting accuracy with respect to Indian motorcycle industry". In: *Transactions on Engineering and Sciences* 2.5 (2014), pp. 25–28.

[27] Erhard Reschenhofer. "Prediction with vague prior knowledge". In: *Communications in Statistics-Theory and Methods* 25.3 (1996), pp. 601–608.

[28] K Krishna Rani Samal et al. "Time series based air pollution forecasting using SARIMA and prophet model". In: *proceedings of the 2019 international conference on information technology and computer communications*. 2019, pp. 80–85.

[29] Robert E Schapire. "The boosting approach to machine learning: An overview". In: *Nonlinear estimation and classification* (2003), pp. 149–171.

[30] Skipper Seabold and Josef Perktold. "statsmodels: Econometric and statistical modeling with python". In: *9th Python in Science Conference*. 2010.

[31] Sean J Taylor and Benjamin Letham. "Forecasting at scale". In: *The American Statistician* 72.1 (2018), pp. 37–45.

[32] Khuyen Tran. *Kats: a Generalizable Framework to Analyze Time Series Data in Python.* July 2021. URL: https://towardsdatascience.com/kats-a-generalizable-framework-to-analyze-time-series-data-in-python-3c8d21efe057.

[33] Eric-Jan Wagenmakers and Simon Farrell. "AIC model selection using Akaike weights". In: *Psychonomic bulletin & review* 11.1 (2004), pp. 192–196.

[34] Eric Wilson and CPF. *How Much Does Forecasting Software Cost? — Demand-Planning.com*. July 2018. URL: https://demand-planning.com/2018/07/12/how-much-does-forecasting-software-cost/.

[35] Fred Woudenberg. "An evaluation of Delphi". In: *Technological forecasting and social change* 40.2 (1991), pp. 131–150.

[36] Hao Wu and David Levinson. "The ensemble approach to forecasting: A review and synthesis". In: *Transportation Research Part C: Emerging Technologies* 132 (2021), p. 103357.

# Curriculum Vitae

Alejandro Saldaña is a Master of Science in Engineering candidate, having studied his bachelor's degree in Industrial and Systems Engineering in one of Mexico's most prestigious schools: Tecnologico de Monterrey. Alejandro Saldaña has 4 years of work experience in the industrial and chemical industries, a six-sigma green belt title by the IIE and several industry related research projects with local companies. He has worked as Juan Carlos' aid since he was still a student obtaining his bachelor's degree. His research is mainly in operations research, data analytics, and industry related applications. He was coauthor of a chapter about risk mitigation in a collection of works and was a speaker in POMS 2021 hosted in Lima in the Marketing and Operations Management track.