Comparative Analysis of Time Series Forecasting Methods: A Case Study on Monthly Beer Production

Beatriz Gonçalves
Master in Data Science
DETI
University of Aveiro
Aveiro, Portugal
Mec.N. 115367

Frederico Vieira
Master in Data Science
DETI
University of Aveiro
Aveiro, Portugal
Mec.N. 98518

Tiago Freitas
Master in Data Science
DETI
University of Aveiro
Aveiro, Portugal
Mec.N. 76748

***Abstract*—This study conducts an in-depth comparison of various time series forecasting techniques including SARIMA, Holt-Winters, and Long short-term memory (LSTM), utilizing a case study on monthly beer production in Australia. The parameters for each model are optimized using first difference and seasonal adjustments, producing forecasts for each model. Furthermore, in order to study the effect of eliminating the seasonality component from the series on forecast accuracy, a variation of the LSTM model is studied. These forecasts are evaluated based on Mean Squared Error (MSE) and Mean Absolute Error (MAE) to provide an objective measure of performance.**

***Index Terms*—Time Series Forecasting, SARIMA, Holt-Winters, LSTM, Seasonality, Mean Squared Error (MSE), Mean Absolute Error (MAE), Monthly Beer Production.**

June 2023

1. Introduction

Beer, an ancient and beloved alcoholic beverage, has a rich history that spans centuries. Its diverse flavors, aromas, and brewing techniques continue to fascinate beer enthusiasts and connoisseurs worldwide. [1] However, behind every pint lies a complex production process, and brewers are constantly seeking ways to optimize this process and improve their craft. To this end, innovative approaches such as time series analysis have gained prominence. [2] [3]

Time series analysis is a robust statistical technique that enables the examination of patterns and trends in data over time. While traditionally used in fields ranging from stock market predictions and weather forecasts to sales projections and demand planning, With advancements in machine learning techniques and the growing volume of data, the prediction accuracy of these models has significantly improved, thereby amplifying their importance and multidisciplinarity. However, it remains essential to choose an appropriate model based on the unique characteristics of the data at hand. By applying time series analysis to the brewing process, brewers can gain deeper insights, enhance quality control, and optimize production efficiency. [4] [5]

In this article, we embark on a comprehensive exploration of the potential of time series analysis in beer production. [6] By employing this technique, brewers can analyze vast amounts of data collected throughout the brewing process, unveiling hidden patterns that may otherwise go unnoticed. From the selection of malt and fermentation to packaging and distribution, time series analysis holds promise in revolutionizing every step of the beer production journey. Here, we will focus on its monthly production, as time series analysis offers the ability to forecast demand, enabling brewers to optimize inventory management and production planning.

This project explores and contrasts three popular time series forecasting methods— SARIMA, Holt-Winters, and Long Short-Term Memory (LSTM)—on a real-world dataset of monthly beer production [6].

The aim of this study is to provide a comprehensive comparison of the chosen models, considering the underlying time series components such as trend and

seasonality. Seasonality is a recurring pattern observable in many time series datasets, particularly in the retail and manufacturing sectors, where demand can fluctuate based on the time of the year. For example, sales tend to peak for the Christmas season and then decline after the holidays. So time series of retail sales will typically show increasing sales from September through December and declining sales in January and February. [7] Now, the trend is a long-term movement in time series data. It can be upward, downward, or constant over time. Trend analysis is the practice of collecting information and attempting to spot a pattern, or trend, in the information. In time series analysis, the trend is an important component, since it basically represents the overall direction of the data over time. [4] Furthermore, the study explores the implications of neglecting the seasonality component in LSTM models.

Due to its distinct seasonal patterns and discernible trend, the beer production dataset [6] presents a compelling use-case. The models' performance is quantified by comparing the forecasts they provide to two common metrics: mean squared error (MSE) and mean absolute error (MAE). [8]

By exploring the capabilities and limitations of each model, this study will contribute to the broader understanding and application of time series forecasting techniques. Furthermore, in this specific case, as the beer industry continues to evolve, the integration of time series analysis promises to bring advantages when compared to its competitors in the industry.

2. Data and exploratory analysis /data transformation

The dataset [6] consists of the monthly beer production recorded over several years, more specifically from January 1956 to August 1995. We can see this in figure 1. In black, we represent the actual monthly production over time, and we can see that there is an overall growth until close to 1975. After that year production remained approximately the same. There are also some spikes, which seem to occur at regular intervals, suggesting a strong seasonality component in the data. We can confirm the observed growth when analyzing the average, represented in red, as the average also rises to 1975 and then remains approximately constant. Lastly, in blue, we can see the standard deviation, which is the measure of the amount of variation or dispersion of a set of monthly beer production values. We can actually see that some periods show more variability than others. For example, we see larger fluctuations in standard deviation during the later years (1973-1974) compared to the earlier years (1956-1957). This indicates that the dispersion of data points can change over different periods. This way, we can infer that the values seem to spread more in the 1970s compared to the 1950s and 1960s.

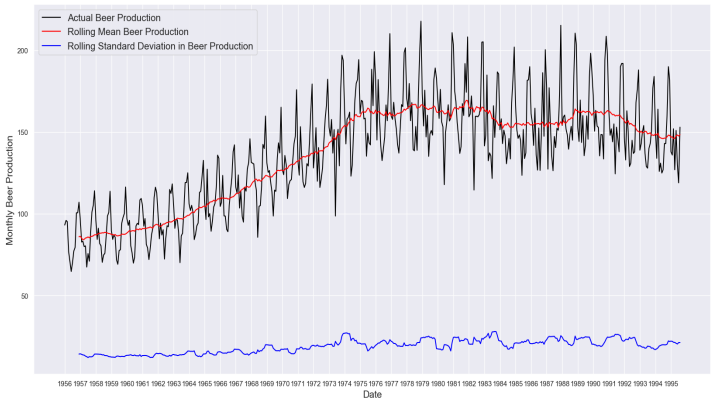

Figure 1. Monthly Production.

After this, we can carry out the decomposition of the time series to verify the evolution of the trend, seasonality, and residuals:

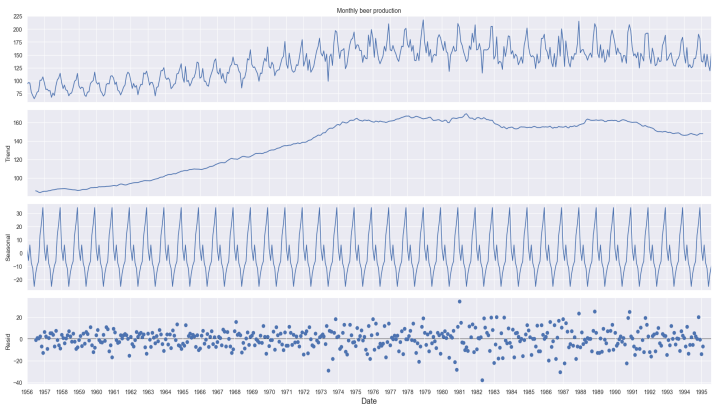

Figure 2. Time Series Decomposition.

In this figure, we can confirm some of the observations we made from figure 1. We can see that there's an overall increasing trend in beer production over time. This suggests that beer production generally

increased from the year 1956 to 1975. After that, it remains approximately constant. From the seasonality plot, we can confirm the previously mentioned strong seasonality component in the data. It appears that the seasonal component has a constant pattern (cyclical) and that these spikes occur towards the end of the year, typically in the last quarter (October to December). This could potentially be attributed to increased demand during festive seasons or holidays during those months. The production then seems to decrease in the first quarter of the year, reaches its lowest in mid-year, before climbing back up again. Lastly, we can see that the residuals are more concentrated around the mean zero. However, we can still dispersion on the residuals.

To better understand the behavior of the residuals, we can look at figure 3, which shows the evolution of the residual's mean. The mean seems to be close to zero in most cases, although we see some unusual spikes as the years grow. In figure 4 we can also see a histogram of the residuals, where we can see that most of the residuals are in the interval [-10,10], however, we consider that there's still a significant portion of them outside this range.

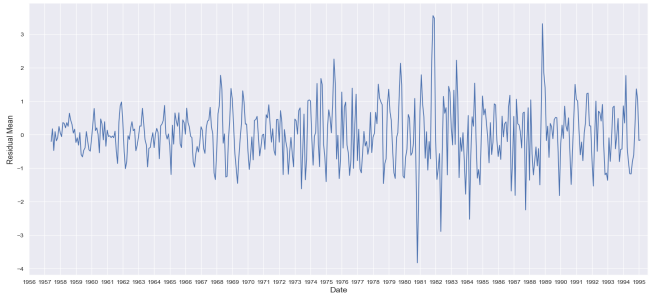

Figure 3. Mean of Residuals Over Time.

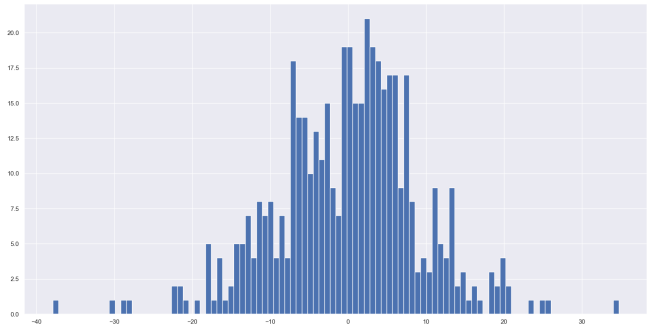

Figure 4. Residuals' Histogram.

Before applying the models for our forecasting, we should start by splitting the data into a training set (for model purposes) and a test set (for forecast purposes). Therefore, we decided to set all but the last 5 years of production as the training dataset. These 5 years will serve as the test set, to test the forecasting. In figure 16, in the Appendix, we can see a representation of the splitting criteria described. In a lighter (gray) color we have the train set and in a darker (black) color we have the test.

3. Model proposals (SARIMA type or/and ETS model)

We can propose SARIMA and Exponential Smoothing State Space Model (ETS) for forecasting. Exponential Smoothing State Space Models (ETS) are a popular choice for time series forecasting, especially for data with seasonality and trend. The ETS model can be implemented with additive or multiplicative error, trend, and seasonality that suit different types of data patterns. There are different types of ETS models such as ETS(A,N,N) for additive errors, no trend, no seasonality; ETS(A,A,N) for additive errors, additive trend, no seasonality; and so on. The ETS model we will apply to our data is the Holt-Winters model. [9]

3.1. SARIMA

A Seasonal Autoregressive Integrated Moving Average (SARIMA) model is a powerful statistical technique used for time series forecasting. It is an extension of the traditional Autoregressive Integrated Moving Average (ARIMA) model that incorporates seasonal patterns in the data.

The SARIMA model takes into account not only the autoregressive (AR), moving average (MA), and differencing components of the data but also the seasonal variations that occur at fixed intervals. It is particularly useful when dealing with data that exhibit predictable patterns that repeat over fixed time intervals, such as monthly, quarterly, or yearly cycles.

[10] The components of a SARIMA model are as follows:

- **Seasonal Component:** The seasonal component accounts for the repeating patterns in the data over fixed intervals. It captures the relationship between the observed values and

the corresponding observations from previous seasons;

- **Autoregressive Component (AR):** The autoregressive component models the dependency between the current observation and a number of lagged observations from the same series. It takes into account the linear relationship between past values and the present value of the series;
- **Integrated Component (I):** The integrated component incorporates differencing into the model. Differencing is performed on the time series data to remove trends or seasonality. It transforms the series into a stationary series, making it easier to model using the AR and MA components;
- **Moving Average Component (MA):** The moving average component models the dependency between the current observation and the residual errors from past observations. It captures the random shocks or noise present in the data.

[10]

To identify, estimate, and check models for time series data, we use the Box-Jenkins methodology. It uses an iterative three-stage modeling approach:

1) **Model identification and model selection:** making sure that the variables are stationary, identifying seasonality in the dependent series (seasonally differencing it if necessary), and using plots of the autocorrelation (ACF) and partial autocorrelation (PACF) functions of the dependent time series to decide which (if any) autoregressive or moving average component should be used in the model.
   The ACF measures the correlation between the time series with a lagged version of itself. In other words, it helps us understand how a given data point is related to its past values. For example, a lag of 1 means the correlation between a time series (let's say, from t1 to t20) and its own past values (t0 to t19). If the data has a seasonal component, the ACF plot will often show a cyclical pattern. [11]
   The PACF, on the other hand, measures the correlation between the time series with a lagged version of itself but after eliminating the variations already explained by the intervening comparisons. Essentially, PACF is a conditional correlation; [11]
2) **Parameter estimation** using computation algorithms to arrive at coefficients that best fit the selected ARIMA model. The most common methods use maximum likelihood estimation or non-linear least-squares estimation;
3) **Statistical model checking** by testing whether the estimated model conforms to the specifications of a stationary univariate process. In particular, the residuals should be independent of each other and constant in mean and variance over time.

[12]

3.2. Holt-Winters

The Holt-Winters model, also known as triple exponential smoothing, is a popular and effective time series forecasting method that incorporates trend, seasonality, and level components. It is an extension of the simple exponential smoothing method that considers not only the current level of the series but also the trend and seasonal components. It can be applied to time series data that exhibit both trend and seasonality, allowing for more accurate and robust forecasts.

The model consists of three components:

- **Level Component:** The level component represents the average value of the series over time. It is updated at each time point and represents the "baseline" or central tendency of the data;
- **Trend Component:** The trend component captures the overall direction and rate of change in the series. It reflects the long-term upward or downward movement of the data;
- **Seasonal Component:** The seasonal component accounts for the repeating patterns or cycles in the data that occur at fixed intervals. It captures the regular, predictable fluctuations that repeat across different time periods, such as daily, weekly, or monthly.

[13]

The Holt-Winters model uses exponential smoothing to update and forecast these three components. Exponential smoothing assigns different weights to recent observations, giving more importance to recent data points. The weights are determined by smoothing

parameters, namely $alpha$, $beta$, and $gamma$, which control the rate at which new observations influence the respective components.

To forecast using the Holt-Winters model, three equations are applied:

- **Level Equation:** This equation provides an estimation of the level at the current time point. In its simplest form, it's calculated as a weighted average of the observed value at the current time point and the estimated level at the previous time point adjusted for the trend. Let's denote:
  - L_t as the level at time t;
  - Y_t as the observed value at time t;
  - T_{t-1} as the trend at time $t-1$;
  - S_{t-m} as the seasonal component at time $t-m$ (m is the seasonal period)
  - α as the level smoothing factor (between 0 and 1)

  Then the level equation can be expressed as:

$$L_t = \alpha(Y_t - S_{t-m}) + (1-\alpha)(L_{t-1} + T_{t-1}) \quad (1)$$

- **Trend Equation** (also known as the smoothing equation for the trend): This equation updates the trend component, which is essentially a weighted average of the estimated trend at the current time point and the trend of the previous time point.
  Let's denote T_t as the trend at time t, and β as the trend smoothing factor. Then the trend equation can be expressed as:

$$T_t = \beta(L_t - L_{t-1}) + (1-\beta)T_{t-1} \quad (2)$$

- **Seasonal Equation**(also known as the smoothing equation for the seasonal component): This equation updates the seasonal component. It's essentially a weighted average of the estimated seasonal component at the current time point and the seasonal component at the same point in the previous season.
  Let's denote S_t as the seasonal component at time t, and γ as the seasonal smoothing factor. Then the seasonal equation can be expressed as:

$$S_t = \gamma(Y_t - L_t) + (1-\gamma)S_{t-m} \quad (3)$$

These equations are recursively applied to update the components at each time point and generate forecasts for future periods. In this case, we presented the equations that make the basis of the additive Holt-Winters model. If the model is multiplicative, the equations are slightly adjusted.

[14]

The Holt-Winters model offers different variations based on the nature of the data. There are three main variations:

- **Additive Model:** This variation assumes that the seasonal fluctuations have a constant amplitude and are added to the level component;
- **Multiplicative Model:** This variation assumes that the seasonal fluctuations are proportional to the level component. In this case, the seasonal component is multiplied by the level component;
- **Damped Trend Model:** This variation introduces a damping factor to gradually reduce the influence of the trend over time, making it more suitable for time series with a diminishing trend.

[14]

This model is particularly effective when dealing with time series that exhibit both long-term trends and seasonal variations. However, it is important to note that the model assumes stationarity in the data and may not perform well for highly irregular or unpredictable series. [15]

3.3. Additional Model Proposal: Long Short-Term Memory (LSTM)

This model is not classified as an Exponential Smoothing State Space Model (ETS) or ARIMA model. These latter models are traditional statistical methods, while LSTMs are a type of machine learning model. The methods used to fit these models and make predictions are fundamentally different. [16]

A Long Short-Term Memory (LSTM) model is a type of recurrent neural network (RNN) architecture that is designed to handle sequence data and capture long-term dependencies in the data.

LSTMs are particularly effective in tasks that involve sequential data, such as natural language processing (NLP), speech recognition, time series analysis, and machine translation. They are known for their

ability to overcome the vanishing gradient problem, which is a common issue in training deep neural networks, especially RNNs. [17]

The key feature of an LSTM model is its memory cell, which allows it to selectively remember or forget information over long periods of time. The memory cell consists of three main components: an input gate, a forget gate, and an output gate.

- **Input gate**: The input gate determines which information from the current input and the previous hidden state should be stored in the memory cell. It applies a *sigmoid* activation function to the input and hidden state, producing a value between 0 and 1 for each element of the memory cell;
- **Forget gate**: The forget gate decides which information to discard from the memory cell. It takes the input and the previous hidden state as input and applies a *sigmoid* activation function to determine the amount of information to be forgotten from each element of the memory cell;
- **Output gate**: The output gate controls the flow of information from the memory cell to the next hidden state and the output of the LSTM. It applies a sigmoid activation function to the input and the previous hidden state, combined with the updated memory cell values. It produces a filtered output that captures the relevant information from the memory cell.

These gates, through their activations, enable LSTMs to regulate the flow of information, retain important information over long sequences, and mitigate the impact of irrelevant or noisy inputs. This makes LSTMs well-suited for capturing dependencies and patterns in sequential data that may occur over extended time intervals.

[18]

During training, the parameters of an LSTM model, including the weights and biases of the gates, are optimized using backpropagation through time (BPTT). This process involves updating the weights based on the error calculated at each time step, allowing the LSTM to learn to predict or generate sequences based on the input data. [19]

[16]

4. Future observations forecast

4.1. SARIMA

As we explained when describing the SARIMA model, we perform the Box-Jenkins methodology. This way, we start by testing for stationarity in the time series data.

Stationarity is an important characteristic of time series data that most forecasting models require. A stationary time series has a constant mean and variance over time. There are different types of stationarity. For example, a series is said to be strictly stationary if its joint distribution remains the same at all time periods, and weakly stationary if only mean and variance are constant over time. [20]

The Augmented Dickey-Fuller (ADF) test is one way to test for stationarity. The null hypothesis of the ADF test is that the time series is non-stationary. So if the p-value of the test is less than the significance level (usually 0.05), we can reject the null hypothesis and say that the series is stationary. [21]

The first ADF test was performed on the original time series ('train' data), and the p-value was found to be 0.377238, which is greater than 0.05, indicating that the series is non-stationary:

TABLE 1. ADF TEST RESULTS FOR ORIGINAL DATA.

Test Statistics	-1.806502
p-value	0.377238
No. of lags used	17.000000
Number of observations used	398.000000
Critical value (1%)	-3.446888
Critical value (5%)	-2.868829
Critical value (10%)	-2.570653

To make the series stationary, first differencing was applied to the data, which is the transformation that replaces each data point with the difference between it and the preceding point. This is often effective at removing trends in the data. The ADF test was then re-run on the differenced data, and the p-value was found to be 0.000039, which is less than 0.05, indicating that the differenced series is stationary:

We can verify this in figure 18 in the Appendix, where we plot the differenced data and its mean, for both training and test sets. If a series is stationary, the rolling mean would be roughly a flat, horizontal line.

Seasonal decomposition was then performed on the data, which decomposes the time series into its

TABLE 2. ADF TEST RESULTS ON THE DIFFERENCED DATA.

Test Statistics	-4.875081
p-value	0.000039
No. of lags used	18.000000
Number of observations used	396.000000
Critical value (1%)	-3.446972
Critical value (5%)	-2.868866
Critical value (10%)	-2.570673

trend, seasonal, and residual components. The seasonal component was then extracted and the ADF test was applied to it. The resulting p-value was 0, indicating that the seasonal component is stationary:

TABLE 3. ADF TEST RESULTS FOR THE SEASONAL COMPONENT.

Test Statistics	-3.230271×10^{14}
p-value	0.000000×10^{0}
No. of lags used	12
Number of observations used	403
Critical value (1%)	-3.446681
Critical value (5%)	-2.868739
Critical value (10%)	-2.570605

Again, we can verify this by plotting the seasonal component and the correspondent rolling mean, for both training and test sets, as we show in figure **??** in the Appendix. The rolling mean is the mean of the data within a certain 'window' that rolls along the time axis. If a series is stationary, the rolling mean would be roughly a flat, horizontal line, which is what we see in this case.

We can see that the original beer production data was found to be non-stationary, but after applying first differencing and isolating the seasonal component, stationarity was achieved. This is an important preprocessing step for our forecasting models.

After this, we perform the Autocorrelation Function (ACF) and the Partial Autocorrelation Function (PACF). From the ACF and PACF plots, we can infer the order of the AR (autoregressive) or MA (moving average) terms for the SARIMA model. If the ACF of the differenced series displays a sharp cutoff and/or the lag-1 autocorrelation is negative–i.e., the series may have had a strong negative autocorrelation–we would use an AR model for forecasting. In this case, we can infer the order of the AR term from the point where the PACF plot crosses the upper confidence interval for the first time. If the ACF of the differenced series decays more slowly–i.e., the series may have positive autocorrelation–we would use an MA model. Here, we can infer the order of the MA term from the point where the ACF plot crosses the upper confidence interval for the first time.

Looking at the ACF and PACF of the seasonal component can help us infer the seasonal part of the ARIMA model, noted as (P,D,Q)s.

For this purpose, we have plotted the ACF and PACF functions for both differenced data, in figure 5, and the seasonal component, in figure 6.

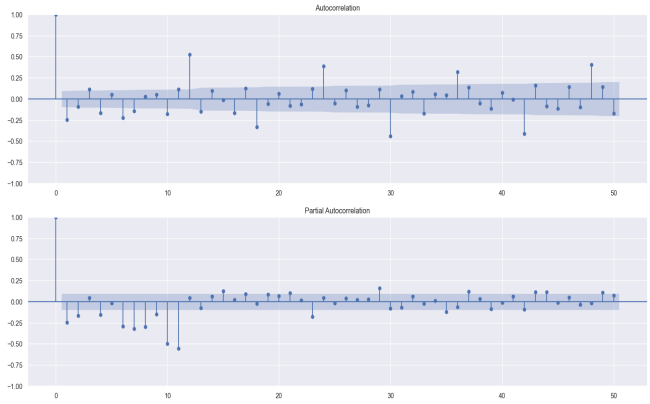

Figure 5. ACF and PACF Functions of the Differenced Data.

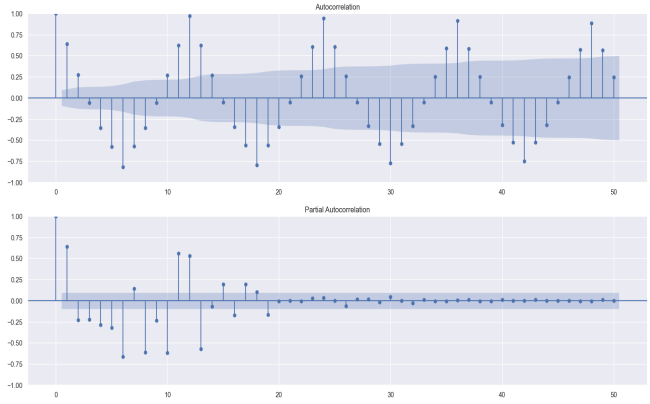

Figure 6. Seasonal Component ACF and PACF functions.

From figure 5, in ACF values, we see a significant positive autocorrelation at lag 12, and again at 24, 36, and 48. This clearly points to a seasonal pattern with seasonality of 12 months (likely due to the nature of beer production). We also see several other lags with significant autocorrelation values (both positive and

negative), which might indicate complex patterns in the data beyond simple seasonality. In the PACF values, we also see significant correlations at various lags. The highest correlation is at lag 12, again indicating strong seasonality.

From the ACF and PACF, we can derive parameters for the SARIMA model. As a starting point, we might consider the following parameters:

- For the AR term (p), we could consider the lag at which the PACF cuts off. In this case, it doesn't seem to have a clear cutoff, suggesting a more complex model might be necessary. However, we see a significant PACF value at lag 1 which could suggest starting with p=1;
- For the MA term (q), we could consider the lag at which the ACF cuts off. Similar to the AR term, there isn't a clear cutoff. We may start with q=1 as we see a significant ACF at lag 1;
- For the seasonal AR term (P), we could consider the lag at which the PACF shows a significant spike in the first seasonal period, which is at lag 12;
- For the seasonal MA term (Q), you could consider the lag at which the ACF shows a significant spike in the first seasonal period, which is at lag 12.

For the seasonal component, in figure 6, the ACF and PACF can be interpreted similarly as they were for the original time series, but we would focus more on the longer lags, corresponding to the seasonal periods.

The ACF values show significant positive autocorrelation at lag 12, lag 24, and so on. This is a strong indicator of seasonality with a period of 12 months.

The PACF values show a significant correlation at lag 12, but afterwards, the values fluctuate significantly with very large absolute values, indicating a complex relationship.

This way, the parameters derived would be similar to the ones we outlined after analyzing the ACF and PACF for the differenced data.

It's important to note that these should be taken as starting points, and further tuning might be necessary. For that, we used $auto\_arima$ function. This function works by running a grid search over multiple combinations of p, d, q (for the ARIMA part) and P, D, Q (for the seasonal part) within given ranges, and selects the combination that minimizes the AIC (Akaike Information Criterion). Then, the selected SARIMA model was fitted to the training data. Using this fitted model, a prediction for the test period is made, and the Mean Squared Error (MSE) and Mean Absolute Error (MAE) between the forecast and the actual test data are computed to measure the accuracy of the forecast. In this case, we got an MSE of, approximately 243.83 and an MAE of, approximately, 13.62. In the end, we plot the train data, test data and forecast data in order to visualize how well the model is predicting:

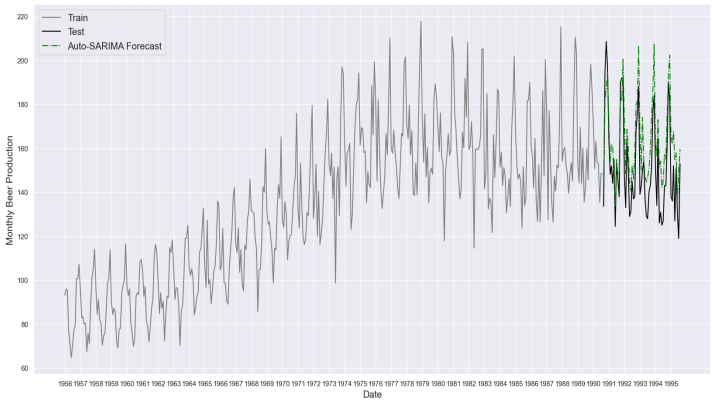

Figure 7. Auto Sarima Forecast.

The original seasonal order suggested by auto-ARIMA was (2,0,1,12), which suggests a second-order seasonal autoregression (AR), no seasonal differencing, a first-order seasonal moving average (MA), and a seasonality of 12. However, when analyzing the summary statistics for the second component of the seasonal AR (ar.S.L24), the following was noted:

- The coefficient (coef) was 0.078. This is the value of the AR parameter itself;
- The standard error (std err), which measures the accuracy of this coefficient, was 0.055;
- The z-score (z) was 0.142, which is the coefficient divided by its standard error. It's a measure of how many standard deviations the coefficient is from 0;
- The p-value ($P > |z|$) was 0.887, which is a measure of the probability that the coefficient equals 0. If the p-value is small (typically less or equal to 0.05), it indicates strong evidence that the coefficient is different from 0;
- The 95% confidence interval for the coefficient was [-0.101, 0.116]. This means we are 95%

confident that the actual value of the coefficient is within this range.

In this case, the p-value was very large (0.887), much larger than the typical threshold of 0.05. This means there is insufficient evidence to conclude that the coefficient is significantly different from 0. In other words, it might as well be 0 for our purposes. This is further reinforced by the fact that the 95% confidence interval for the coefficient includes 0.

As a result, the conclusion was made to disregard the second order of seasonal AR in the model. Therefore, instead of the original order of (2,0,1,12), a new SARIMA model was manually fitted with a modified order of (1,0,1,12). This new order suggests a first-order seasonal autoregression (AR), no seasonal differencing, a first-order seasonal moving average (MA), and a seasonality of 12.

The same steps are repeated for this new SARIMA model: fitting the model, predicting the test data, and computing the MSE and MAE. We got an MSE of, approximately, 243.83 and an MAE of 13.59.

The most important results we got from this model are as follows:

TABLE 4. SELECTED SARIMAX RESULTS

Term	Coefficient	Std. Err.	P-value	[0.025, 0.975]
ar.L1	-1.1535	0.004	0.000	-1.162, -1.145
ar.L2	-0.9961	0.005	0.000	-1.006, -0.987
ma.L1	0.2312	0.025	0.000	0.182, 0.281
ma.L2	-0.0180	0.025	0.474	-0.067, 0.031
ma.L3	-0.8650	0.023	0.000	-0.910, -0.820
ar.S.L12	0.9964	0.004	0.000	0.990, 1.003
ma.S.L12	-0.8330	0.035	0.000	-0.901, -0.765
$sigma^2$	81.8828	4.683	0.000	72.705, 91.061

TABLE 5. TEST STATISTICS AND RESIDUAL METRICS

Test/Metric	Value
Ljung-Box (L1) (Q)	0.11
Prob(Q)	0.74
Jarque-Bera (JB)	44.31
Prob(JB)	0.00
Heteroskedasticity (H)	5.22
Prob(H) (two-sided)	0.00
Skew	-0.24
Kurtosis	4.53

- **Autoregressive terms** (ar.L1, ar.L2): Both the first and second autoregressive terms are statistically significant given their p-values are less than 0.05 (0.000). The coefficients for these terms are -1.1535 and -0.9961 respectively;
- **Moving average terms** (ma.L1, ma.L2, ma.L3): The first and third moving average terms are significant (p-value < 0.05), with coefficients 0.2312 and -0.8650 respectively. The second moving average term, ma.L2, with a coefficient of -0.0180 is not statistically significant given its p-value of 0.474, suggesting this term does not significantly impact the model;
- **Seasonal components** (ar.S.L12, ma.S.L12): The seasonal autoregressive and moving average components are statistically significant (p-value < 0.05). The coefficients for these terms are 0.9964 and -0.8330 respectively;
- **Sigma Squared**: This is the estimate of the variance of the residuals (or error terms). Its value is 81.8828, and this estimate is statistically significant given its p-value is less than 0.05.
- **Ljung-Box test** and **Jarque-Bera test**: The Ljung-Box test checks for autocorrelation in the residuals, with a null hypothesis of no autocorrelation. The high p-value (0.74) suggests that we fail to reject the null hypothesis, i.e., there's no significant autocorrelation in the residuals. The Jarque-Bera test checks the null hypothesis that the residuals are normally distributed. The very low p-value (0.00) indicates that we reject the null hypothesis, i.e., the residuals are not normally distributed;
- **Heteroskedasticity**: This test checks the null hypothesis of constant variance in the residuals (homoscedasticity). A low p-value (0.00) leads to rejecting the null hypothesis, suggesting the presence of heteroskedasticity, i.e., the variance of the residuals changes over time;
- **Skew and Kurtosis**: These are measures of the shape of the distribution of the residuals. Skewness is a measure of symmetry, or more precisely, the lack of symmetry. A distribution, or data set, is symmetric if it looks the same to the left and right of the center point. The skewness for the residuals is -0.24. Kurtosis is a measure of whether the data are heavy-tailed or light-tailed relative to a normal distribution. The kurtosis of the residuals is 4.53.

Then, we calculate the forecasted values along with

the 95% confidence interval. The actual test data, the forecasted data, and the confidence interval are plotted for visualization:

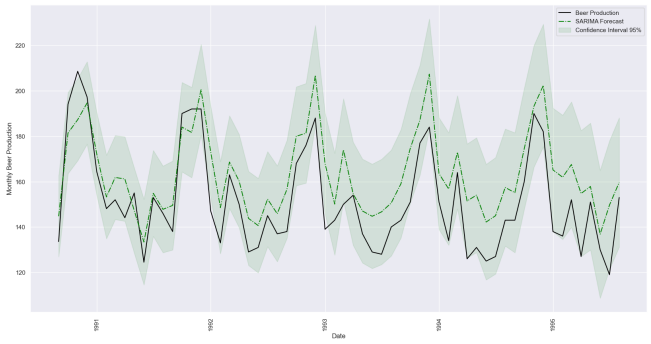

Figure 8. SARIMA Forecast.

To provide common statistical checks on the residuals of the fitted model, we do a diagnostic of the model. This plot includes the standardized residual plots, histogram plus estimated density of standardized residuals along with a Normal Q-Q plot, and a Correlogram. These checks are helpful to ensure that the residuals of the model are uncorrelated, normally distributed, and have zero mean.

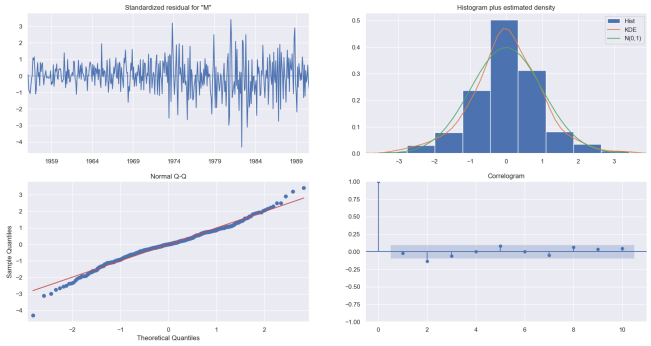

Figure 9. SARIMA Diagnostics.

From the first plot, it seems that the mean of the standardized residuals is around zero value, with the histogram showing that their distribution seems to be very similar to normal distribution centered around 0 (when comparing the residual distribution in orange with the N(0,1) in green). From the Q-Q plot, even the tails are close to the line meaning that the sample quantiles are very close to the theoretical quantities, which confirms that the residuals are normally distributed. Looking at the residuals: the ACF values are inside the confidence interval, showing that they are approximately 0, indicating non-correlation between them and similar behavior to white noise. This is a good sign, indicating that our model has appropriately captured the underlying temporal dependencies in the time series data, and the residuals are just random fluctuations around zero that can't be predicted.

It seems that the residuals of our SARIMA model don't have a perfect normal distribution according to the Jarque-Bera test and kurtosis value. However, the diagnostic plots suggest that they are approximately normally distributed and the residuals do not exhibit significant autocorrelation, which are positive signs for our model. This divergence may occur due to the limitations of the Jarque-Bera (JB) test, as well as the inherent differences between visual and statistical examinations of normality.

The JB test is asymptotically valid, meaning it's more accurate for large sample sizes. However, the power of the JB test can be low in small samples for certain types of alternative hypotheses, especially for distributions with short tails or a bimodal shape. Thus, the test can falsely indicate a deviation from normality. Additionally, the JB test assumes that the observations are independently and identically distributed (i.i.d.). However, the residuals from time series models often have temporal dependencies (autocorrelation) or changes in variance over time (heteroskedasticity). If these issues are present, they could impact the JB test result.

Lastly, when we plot the diagnostics, we're visually inspecting the distribution of the residuals. This can be subjective and depends on the scale and binning of the histogram. In contrast, the JB test is a formal statistical test with a strict null hypothesis (normality) and can be sensitive to small deviations.

[22]

4.2. Holt-Winters

As mentioned in the Introduction, we also performed the Holt-Winters Exponential Smoothing (ETS) method. We used additive seasonality, but we cannot be sure about the trend. We tried additive, multiplicative, or without specifying the trend. The best results (in terms of MSE and MAE) came from not specifying the trend. T

The Mean Squared Error (MSE) and Mean Absolute Error (MAE) between the predicted and actual

values were calculated. We got an MSE of 233.32 and an MAE of 13.03, approximately.

Just like we did when exploring the SARIMA Results, we generate a summary that provides further statistical tests and measures about the model's residuals, such as the Ljung-Box test for autocorrelation and the Jarque-Bera test for normality:

TABLE 6. HOLT-WINTERS PERFORMANCE SUMMARY.

	Value		Value
Ljung-Box (Q)	209.62	Jarque-Bera (JB)	31.39
Prob(Q)	0.00	Prob(JB)	0.00
Heteroskedasticity (H)	3.98	Skew	-0.27
Prob(H) (two-sided)	0.00	Kurtosis	4.23

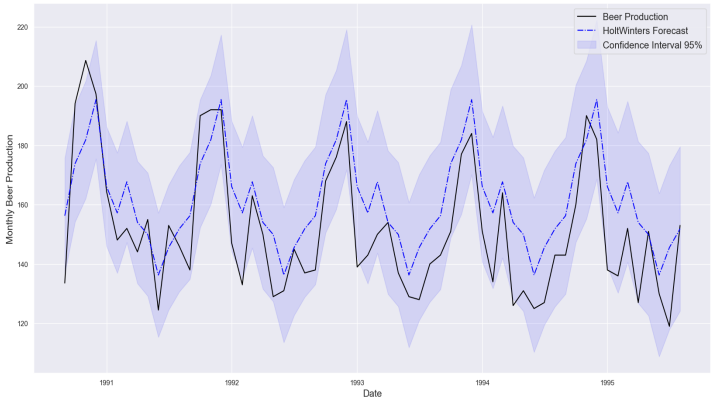

Figure 10. Holt-Winters Forecast.

- **Ljung-Box (Q) Test**: The Ljung-Box test suggests that there is significant autocorrelation in the residuals, given the very low p-value (0.00);
- **Jarque-Bera (JB) Test**: The Jarque-Bera test indicates that the residuals are not normally distributed, as the p-value is very low (0.00);
- **Heteroskedasticity (H) Test**: The heteroskedasticity test reveals that the variance of the residuals is not constant over time (heteroskedastic), as the p-value is very low (0.00);
- **Skew and Kurtosis:** The skewness of -0.27 indicates a slight left skew in the residuals. The kurtosis value of 4.23 suggests that the tails of the distribution of residuals are heavier than those of a normal distribution, indicating a larger number of outliers.

However, we need to have in mind the limitations we outlined when discussing the Sarima Results, and remember we are dealing with a relatively small dataset.

Let's have a look at the plot of the actual values ('Beer Production'), the forecasts ('HoltWinters Forecast'), and the 95% confidence intervals. This helps visualize the model's performance and its forecasted values compared to actual values.

As we can see, this prediction is quite good, as it is very close to the original result and also manages to predict well the oscillations in the movement, being able to reach quite successfully its maximums and minimums. It is always important not only to capture the overall trend of the data, but also to accurately predict the oscillations or fluctuations, including the peaks and troughs. If the model does this effectively, it suggests that it is appropriately capturing the underlying patterns in the data.

Similarly to the SARIMA results, we can see in Figure 11 that the mean of our residuals is centered around 0, despite the values being fairly larger when compared to the SARIMA residuals. Again, we also see that the residuals appear to be normally distributed (histogram and QQ plot) despite the result of the JB test. This seems to validate our hypothesis that the JB test might not be very suited to our data (likely too small for the test). We can also see that there seems to be some very small correlation between the residuals as indicated by the results of the Ljung-Box test. Some of these results appear to indicate that the Holt-Winters might not be the most adequate model for our dataset, but it still seems to have produced very good predictions, as mentioned above.

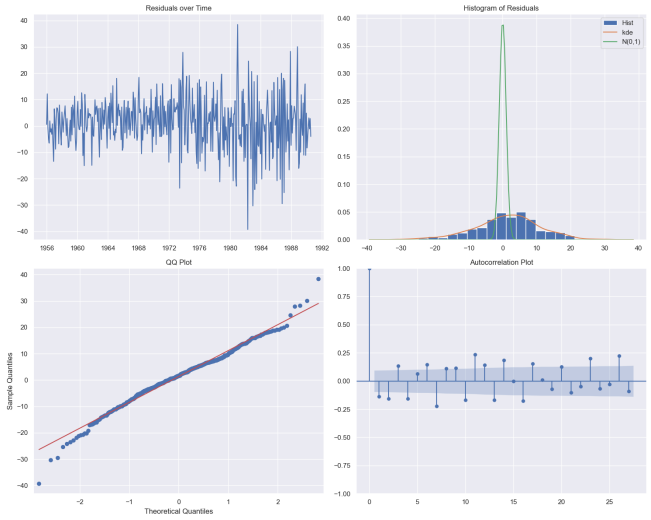

Figure 11. Holt Winters Diagnostics.

4.3. Long Short-Term Memory (LSTM)

Before we fit the LSTM model on the training data, we start by scaling our training and test datasets. This is a common preprocessing step for neural networks. When a network is fit on unscaled data that has a range of values, it is possible for large inputs to slow down the learning and convergence of our network and in some cases prevent the network from effectively learning the problem. [23]

A sequential model is built using Keras [24] where two LSTM layers are followed by a Dense layer. The LSTM layers use the ReLU (Rectified Linear Unit) activation function, and the Dense layer uses a linear activation function [25]:

TABLE 7. MODEL ARCHITECTURE SUMMARY

Layer (type)	Output Shape	Param #
LSTM	(None, 12, 16)	1152
LSTM_1	(None, 4)	336
Dense	(None, 1)	5

Total params	1,493
Trainable params	1,493
Non-trainable params	0

The model uses the Adam optimization algorithm [26] for training, and the mean squared error (MSE) as the loss function (also for training). After compiling our model, we compute the Mean Squared Error (MSE) and Mean Absolute Error (MAE) between the predicted and actual beer production values (performance on the test set).

We got an MSE of, approximately, 210.13, and an MAE of, approximately, 12.39. Unlike MSE, the MAE is on the same scale as the original target variable, making it somewhat easier to interpret. In this case, we can say that, on average, the predictions made by our model are about 12.39 units away from the actual values.

Plotting our forecast from the LSTM model applied, we got the following result:

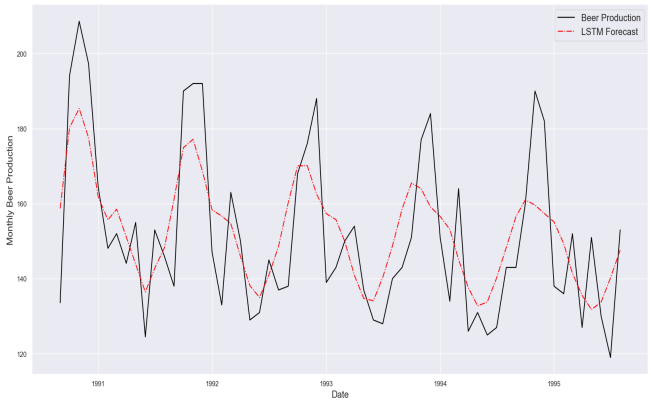

Figure 12. LSTM Forecasting.

From the MSE and MAE values together with the forecast visualization from figure 12 we felt that we could improve our model even more.

As we already know, seasonality is a repeating pattern within each time period - in this case, it's the monthly cycle in beer production. Many time-series forecasting models, like SARIMA, rely heavily on identifying and modeling seasonality.

However, when it comes to LSTM models, the presence of strong seasonality can sometimes interfere with the model's ability to learn underlying patterns in the data. This is particularly true if the patterns that emerge from the seasonality are strong but do not necessarily help with the actual forecasting task at hand.

Removing the seasonality component might help the LSTM model focus on other, potentially more meaningful trends or patterns in the data. These could be, for example, long-term upward or downward trends, or perhaps more complex patterns that are not immediately obvious.

Also, by comparing the results of an LSTM model with and without seasonality, we can gain insights into how much of the predictability in the data is coming from the seasonality itself, and how much is coming from other factors. If the model performs better without the seasonality, this could indicate that the seasonal patterns were perhaps "distracting" the model from learning other, more important trends. [27]

4.4. LSTM (Without Seasonality)

Looking at the forecast of the previous LSTM it seems that one of the reasons for the poor prediction is that the its losing its seasonal component over time. We could retrain the model after every 12 months and make new predictions but, because the seasonal component of our data seems to remain constant, we decided to try to train a LSTM to predict the trend of the time series. In this case, the seasonality component is removed before fitting a new LSTM model. This is done by subtracting the seasonal value of the monthly average beer production from the original values. The steps here are very similar to the previously applied LSTM, with the difference being that the LSTM model is trained and tested on the deseasonalized data. After the predictions are made, the seasonal component is added back to the predicted values to get them back to their original scale. Just like in the first part, the MSE and MAE are calculated. We got an MSE of, approximately, 207.80, and an MAE of, approximately, 12.10. After plotting the forecast it seem clear that it is performing better than the original LSTM we applied before:

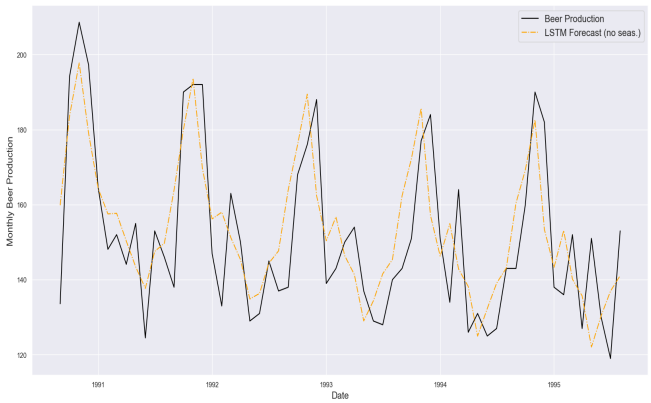

Figure 13. LSTM Without Seasonality Forecasting.

A possible explanation might be that our LSTM model without seasonality made larger mistakes on a few specific points, which would have significantly increased the MAE and MSE. However, for the majority of the predictions, it might have actually done a better job than the LSTM with seasonality, providing a better fit for most of the points. This is why it's important to not solely rely on aggregate error metrics when evaluating your models. Visualizing the actual vs. predicted plots is a great way to gain additional insights into our model's performance. If the plot of the LSTM without seasonality looks closer to the actual data than the plot of the LSTM with seasonality, despite the higher MSE and MAE, it could be that the model without seasonality is indeed a better model, for this specific prediction.

5. Conclusions

A summary of MSE and MAE for all the models can be seen in Table 8, as well a comparison of all forecast with the actual production in Figure 19. We can see the performance of SARIMA and HoltWinters are very similar, which makes sense given the simplicity of the data. The LSTM without seasonality seemed to perform slightly better than the other models, with LSTM on the normal data having a good result in terms of MAE and MSE despite the actual forecast looking to be much worse. When comparing the LSTM (no seas.) forecasts with the other models it appears that it was able to predict slightly better some of the latter peaks and valleys in production (perhaps because its predicting only the trend), as both SARIMA and HoltWinters seem to make slight over-predictions.

TABLE 8. MSE AND MAE FOR OUR MODELS.

Model	MSE	MAE
SARIMA	243.83	13.59
HoltWinters	233.32	13.03
LSTM	210.13	12.39
LSTM (no seas.)	207.80	12.10

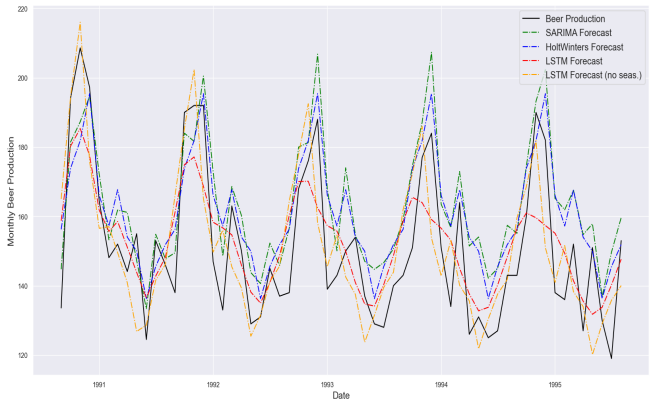

Figure 14. Forecast summary plots (this figure is repeated in a bigger size in the appendix for better visualization).

However, these differences seem to be very small (especially between SARIMA and HoltWinters). This is mostly because our data is a fairly simple time series and, as such, can be well studied using a simple SARIMA model. It would be interesting to see how these models would perform on more complex datasets (for example, with exponential time complexity). It is also important to note that LSTM's are much more expensive (both computationally and time-wise) to train and may have more variability (as the result may depend on the random initialization of the parameters in the model), especially if the amount of data is limited. In our case, while most forecasts were fairly good, for some initialization seeds, the forecasts would sometimes seem to almost "explode" (especially for longer forecasts), as the predicted values were much bigger than the actual values, as seen in Figure 15.

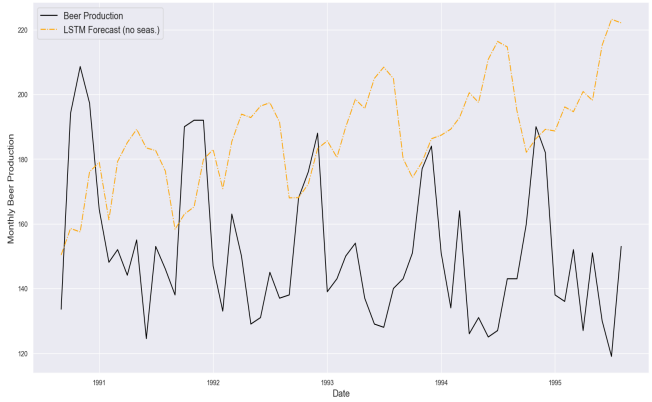

Figure 15. Poor forecast with LSTM (no seas.), with $\mathsf{MSE} = 2374.30$ and $\mathsf{MAE} = 41.90$

References

[1] L. Raihofer, M. Zarnow, M. Gastl, and M. Hutzler, "A short history of beer brewing," *EMBO Reports*, vol. 23, p. e56355, 2022.

[2] B. Danijel and A. Faganel, "Forecasting the primary demand for a beer brand using time series analysis," *Organizacija*, vol. 41, no. 3, pp. 116–124, 2008.

[3] D. Petrides, "Brewery (beer production) process modeling and cycle time analysis with superpro designer," 2020.

[4] "Time series analysis: Definition, types, techniques, and when it's used," https://www.tableau.com/learn/articles/time-series-analysis, accessed: 2023-06.

[5] "What is a time series and how is it used to analyze data?" https://www.investopedia.com/terms/t/timeseries.asp, accessed: 2023-06.

[6] "Australian monthly beer production," https://www.kaggle.com/code/mpwolke/australian-monthly-beer-production, accessed: 2023-06.

[7] "Seasonality," https://itl.nist.gov/div898/handbook/pmc/section4/pmc443.htm, accessed: 2023-06.

[8] "Mse vs mae, which is the better regression metric?" https://stephenallwright.com/mse-vs-mae/, accessed: 2023-06.

[9] "Ets models," https://www.statsmodels.org/dev/examples/notebooks/generated/ets.html, accessed: 2023-06.

[10] "A gentle introduction to sarima for time series forecasting in python," https://machinelearningmastery.com/sarima-for-time-series-forecasting-in-python/, accessed: 2023-06.

[11] "Time series: Interpreting acf and pacf," https://www.kaggle.com/code/iamleonie/time-series-interpreting-acf-and-pacf, accessed: 2023-06.

[12] T. Cipra, "Box–jenkins methodology," 2020.

[13] "A thorough introduction to holtwinters forecasting," https://medium.com/analytics-vidhya/a-thorough-introduction-to-holt-winters-forecasting-c21810b8c0e6, accessed: 2023-06.

[14] . A. G. Hyndman, R.J., *Forecasting: principles and practice, 2nd edition*. OTexts: Melbourne, Australia, 2018.

[15] C. Wongoutong, "The effect on forecasting accuracy of the holt-winters method when using the incorrect model on a non-stationary time series," 2021.

[16] "LSTM Recurrent Neural Networks: How to Teach a Network to Remember the Past," https://towardsdatascience.com/lstm-recurrent-neural-networks-how-to-teach-a-network-to-remember-the-past-55e54c2ff22e, accessed: 2023-06.

[17] "Visualizing the vanishing gradient problem," https://machinelearningmastery.com/visualizing-the-vanishing-gradient-problem/, accessed: 2023-06.

[18] "Lstms explained: A complete, technically accurate, conceptual guide with keras," https://medium.com/analytics-vidhya/lstms-explained-a-complete-technically-accurate-conceptual-guide-with-keras-2a650327e8f2, accessed: 2023-06.

[19] "A gentle introduction to backpropagation through time," https://machinelearningmastery.com/gentle-introduction-backpropagation-time/, accessed: 2023-06.

[20] "Stationarity," https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc442.htm, accessed: 2023-06.

[21] "Augmented dickey fuller test (adf test) – must read guide," https://www.machinelearningplus.com/time-series/augmented-dickey-fuller-test/?utm_content=cmp-true, accessed: 2023-06.

[22] "More on the limitations of the jarque-bera test," https://davegiles.blogspot.com/2014/04/more-on-limitations-of-jarque-bera-test.html, accessed: 2023-06.

[23] "How to scale data for long short-term memory networks in python," https://machinelearningmastery.com/how-to-scale-data-for-long-short-term-memory-networks-in-python/, accessed: 2023-06.

[24] "Keras," https://www.tensorflow.org/guide/keras?hl=es-419, accessed: 2023-06.

[25] J. Lederer, "Activation functions in artificial neural networks: A systematic overview," 2021.

[26] "Gentle introduction to the adam optimization algorithm for deep learning," https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/, accessed: 2023-06.

[27] K. Bandara, C. Bergmeir, and S. Smy, "Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach," 2020.

Appendix

Train-Test-Split Criteria

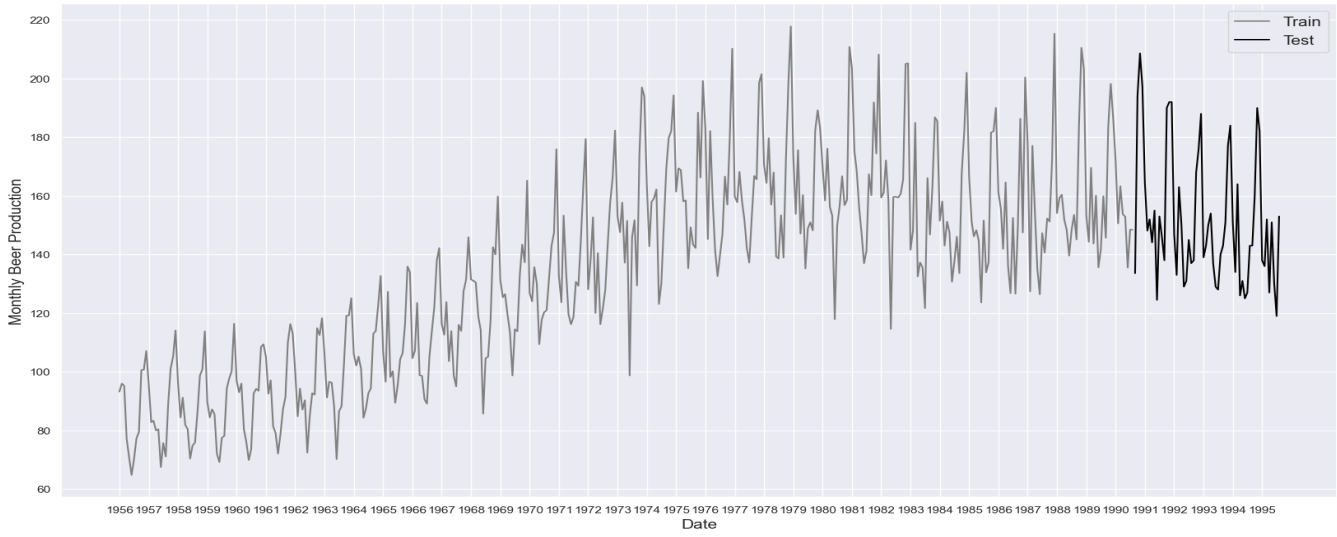

Figure 16. Train-Test-Split.

Box-Jenkins Methodology

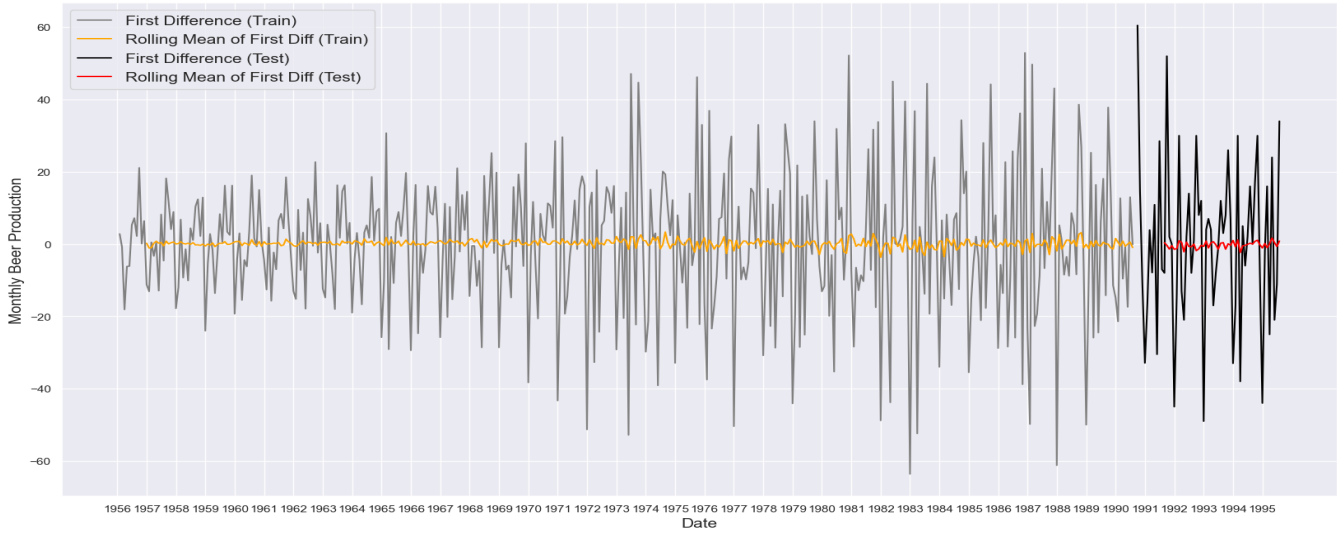

Figure 17. Differenced Training and Test Sets.

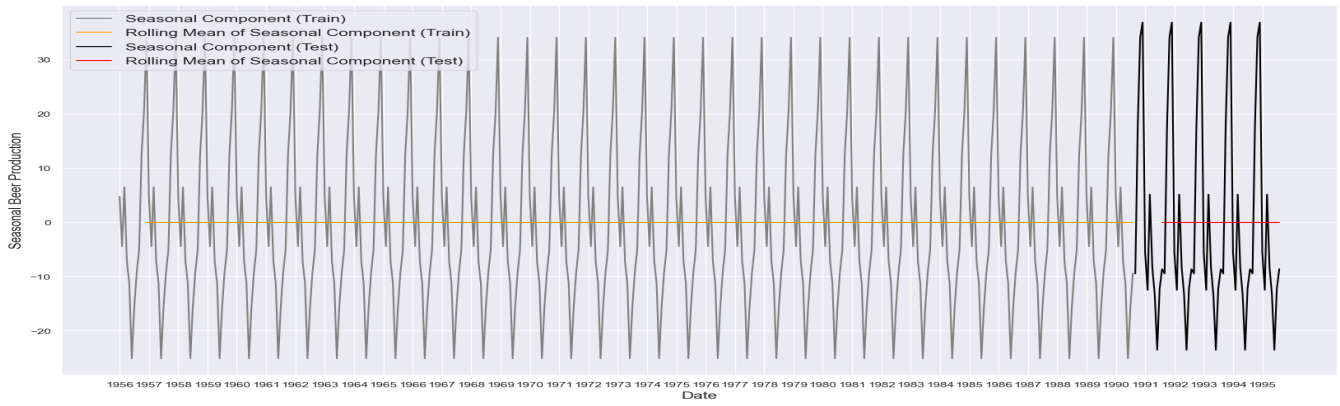

Figure 18. Seasonal Component of Training and Test Sets.

Summary of forecasts

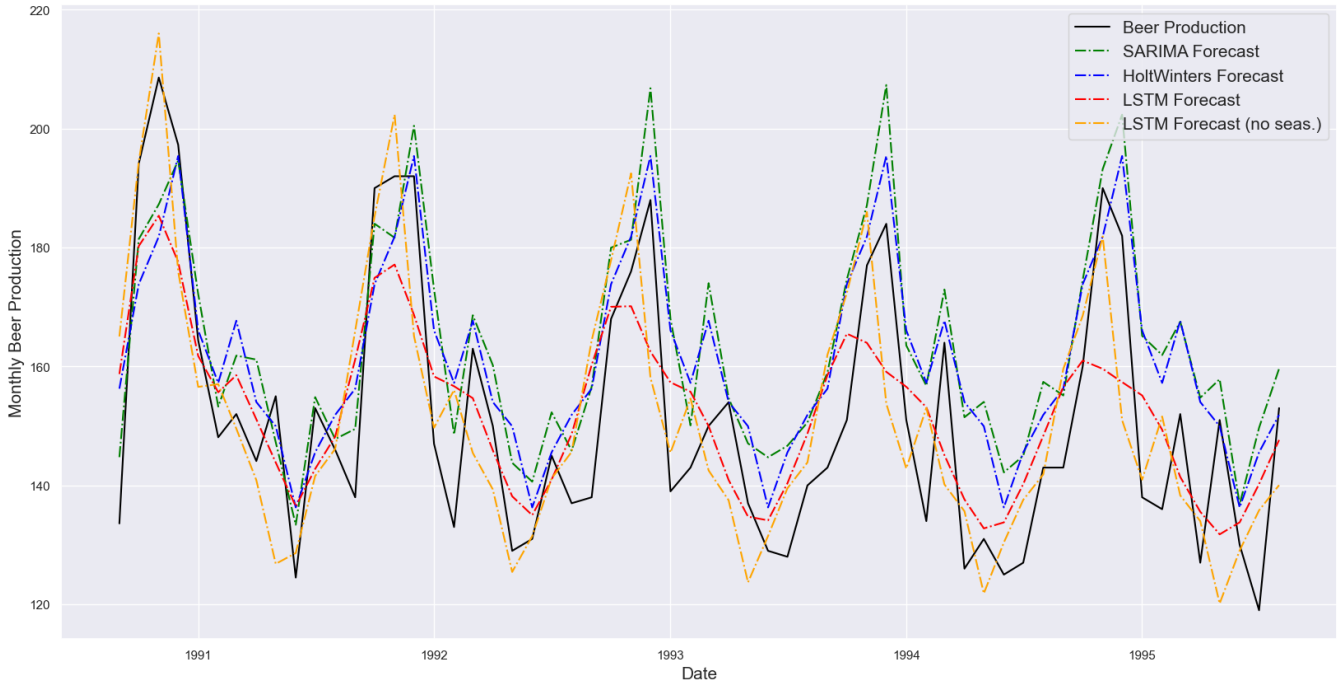

Figure 19. Forecast summary plots.