

6 Steps to Inventory Forecasting with ARIMA Models

Inventory forecasting is an essential aspect of inventory management that helps businesses to predict the demand for their products accurately. One of the most effective ways to perform inventory forecasting is by using ARIMA models. ARIMA stands for Auto Regressive Integrated Moving Average, and it is a time-series forecasting method that considers the past values of the data and uses them to make predictions about future values. In this guide, we will provide a step-by-step guide for inventory forecasting with ARIMA models.

Step 1: Collect and Clean the Data

The first step in inventory forecasting is to collect and clean the data. Data can be collected from various sources, such as point of sale systems, inventory management software, or customer orders. The data should be cleaned to remove any errors or missing values that could affect the accuracy of the forecast. One way to check for missing values is to use the `na.check` function in R, which checks for missing values in a data frame. Another method is to use the `complete.cases` function to remove any rows with missing values.

Step 2: Visualize the Data

After cleaning the data, the next step is to visualize it to get a better understanding of its patterns and trends. One way to do this is by using a time series plot, which plots the data against time. For instance, let us consider the following dataset, which shows the monthly sales of a product over a period of two years.

```
sql Copy code
```

Month	Sales
Jan-2018	100
Feb-2018	120
Mar-2018	150
Apr-2018	130
May-2018	160
Jun-2018	170
Jul-2018	180
Aug-2018	190
Sep-2018	210
Oct-2018	220
Nov-2018	230
Dec-2018	240
Jan-2019	260
Feb-2019	280
Mar-2019	300
Apr-2019	320
May-2019	350
Jun-2019	370
Jul-2019	390
Aug-2019	400
Sep-2019	420
Oct-2019	450
Nov-2019	480
Dec-2019	500

We can create a time series plot in R using the following code:

```
library(ggplot2)
library(scales)

sales_data <- read.csv("sales.csv")

ggplot(sales_data, aes(x = as.Date(Month, "%b-%Y"), y = Sales)) +
  geom_line() +
  labs(x = "Month", y = "Sales") +
  scale_x_date(date_breaks = "3 months", date_labels = "%b-%y") +
  theme_bw()
```

The resulting plot should give us a better understanding of the patterns and the upward trends in the sales data over the two-year period.

Step 3: Check for Stationarity

The next step in inventory forecasting with ARIMA models is to check for stationarity. Stationarity refers to the property of a time series where the statistical properties of the series, such as mean and variance, remain constant over time.

ARIMA models require stationary data, so if the data is not stationary, we need to make it stationary before proceeding with the modeling.

One way to check for stationarity is by using the Augmented Dickey-Fuller (ADF) test, which tests whether a unit root is present in the data. A unit root is a characteristic of non-stationary data where the mean and variance change over time.

We can use the `adf.test` function in R to perform the ADF test. If the p-value is less than 0.05, we can reject the null hypothesis and conclude that the data is stationary.

If the data is not stationary, we can make it stationary by differencing the data. Differencing involves taking the difference between consecutive observations. We can use the `diff` function in R to difference the data.

Step 4: Determine ARIMA Model Parameters

The next step is to determine the parameters of the ARIMA model. An ARIMA model is specified by three parameters: p , d , and q .

p refers to the number of autoregressive terms in the model, which captures the effect of past values on the current value. d refers to the order of differencing required to make the data stationary. q refers to the number of moving average terms in the model, which captures the effect of past forecast errors on the current value.

We can use the autocorrelation function (ACF) and partial autocorrelation function (PACF) plots to determine the values of p and q .

The ACF plot shows the correlation between the current value and lagged values at different time lags. If the ACF plot shows a gradual decline in the correlation, it suggests that we need to difference the data. If there is a sharp drop after a certain lag, it suggests the value of q .

The PACF plot shows the correlation between the current value and lagged values after removing the effect of intermediate lags. If the PACF plot shows a gradual decline in the correlation, it suggests that we need to difference the data. If there is a sharp drop after a certain lag, it suggests the value of p .

We can use the `auto.arima` function in R to automatically select the best ARIMA model based on the AIC value. A lower AIC value indicates a better fit.

Step 5: Fit the ARIMA Model

After determining the parameters of the ARIMA model, we can fit the model to the data using the `ARIMA` function in R. We can use the `forecast` function to make predictions for future periods.

Step 6: Evaluate the Model

The final step is to evaluate the performance of the model. We can use various metrics, such as mean absolute error (MAE), mean squared error (MSE), and root mean squared error (RMSE), to evaluate the accuracy of the model.

We can also plot the actual sales data against the predicted values to visually inspect the performance of the model.

Conclusion

ARIMA models are a powerful tool for inventory forecasting, as they can capture the underlying patterns and trends in the data. By following the steps outlined in this guide, we can perform inventory forecasting with ARIMA models and make accurate predictions about future demand for our products. However, it is important to keep in mind that no model is perfect, and we should always monitor the performance of the model and adjust it as necessary.