

How can naive forecasting be performed in R, and could you provide some examples?

Authored by
stats writer

April 28, 2024

RECOMMENDED CITATION

stats writer (2024). *How can naive forecasting be performed in R, and could you provide some examples?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=140573>

Naive forecasting in R is a simple and intuitive method of predicting future values based solely on the most recent observed value. This approach assumes that historical patterns will continue into the future, without taking into account any other factors or variables. To perform naive forecasting in R, one can use the "naive" function from the "forecast" package, which automatically calculates the forecast based on the most recent data point. For example, if we have a time series dataset of monthly sales data, we can use naive forecasting to predict the sales for the next month based on the sales from the previous month. Another example could be using naive forecasting to predict stock prices by using the most recent closing price as the forecast for the next day. Naive forecasting in R is a quick and easy way to make predictions, but it should be used with caution as it does not account for any external factors that may impact the forecasted values.

Perform Naive Forecasting in R (With Examples)

A naive forecast is one in which the forecast for a given period is simply equal to the value observed in the previous period.

For example, suppose we have the following sales of a given product during the first three months of the year:

Month	Actual Sales
January	34
February	37
March	44

The forecast for sales in April would simply be equal to the actual sales from the previous month of March:

Month	Actual Sales	Forecasted Sales
January	34	
February	37	
March	44	
April	?	44

Although this method is simple, it tends to work surprisingly well in practice.

This tutorial provides a step-by-step example of how to perform naive forecasting in R.

Step 1: Enter the Data

First, we'll enter the sales data for a 12-month period at some imaginary company:

#create vector to hold actual sales data

```
actual <- c(34, 37, 44, 47, 48, 48, 46, 43, 32, 27, 26, 24)
```

Step 2: Generate the Naive Forecasts

Next, we'll use the following formulas to create naive forecasts for each month:

#generate naive forecasts

```
forecast <- c(NA, actual)
```

```
#view naive forecasts
```

```
forecast
```

```
NA 34 37 44 47 48 48 46 43 32 27 26
```

Note that we simply used NA for the first forecasted value.

Step 3: Measure the Accuracy of the Forecasts

Lastly, we need to measure the accuracy of the forecasts. Two common metrics used to measure accuracy include:

Mean absolute percentage error (MAPE) Mean absolute error (MAE)

```
#calculate MAPE
```

```
mean(abs((actual-forecast)/actual), na.rm=T) * 100
```

```
9.898281
```

```
#calculate MAE
```

```
mean(abs(actual-forecast), na.rm=T)
```

3.454545

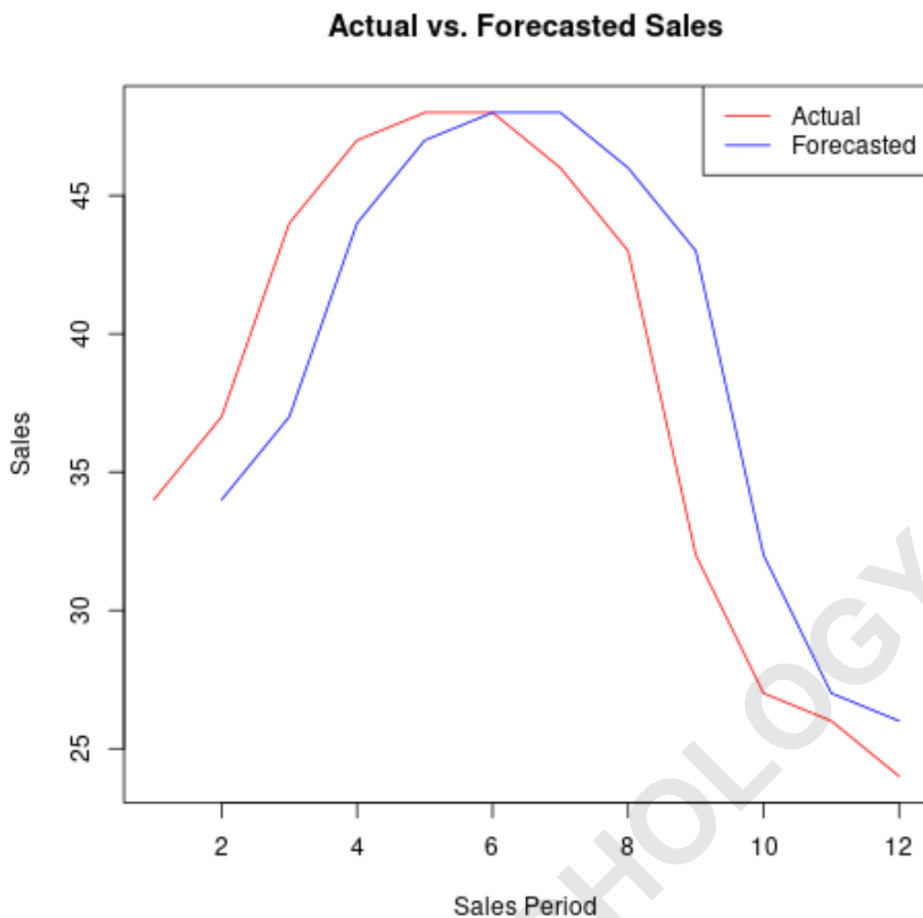
The mean absolute percentage error is 9.898% and the mean absolute error is 3.45

To know if this forecast is useful, we can compare it to other forecasting models and see if the accuracy measurements are better or worse.

Step 4: Visualize the Forecasts

Lastly, we can create a simple line plot to visualize the differences between the actual sales and the naive forecasts for the sales during each period:

```
#plot actual sales  
plot(actual, type='l', col = 'red', main='Actual vs.  
Forecasted Sales',  
xlab='Sales Period', ylab='Sales')  
  
#add line for forecasted sales  
lines(forecast, type='l', col = 'blue')  
  
#add legend  
legend('topright', legend=c('Actual', 'Forecasted'),  
col=c('red', 'blue'), lty=1)
```



Notice that the forecasted sales line is basically a lagged version of the actual sales line.

This is exactly what we would expect the plot to look like since the naive forecast simply forecasts the sales in the current period to be equal to the sales in the previous period.