NDK_MAPE

Last Modified on 04/15/2016 11:07 am CDT

- C/C++
- .Net

```
int __stdcall NDK_MAPE(double * X,
                       double * Y,
                       size_t   N,
                       BOOL     SMAPE,
                       double * retVal
                      )
```

Calculates the mean absolute percentage error (deviation) function for the forecast and the eventual outcomes.

Returns

status code of the operation

Return values

NDK_SUCCESS Operation successful
NDK_FAILED Operation unsuccessful. See Macros for full list.

Parameters

[in] **X** is the original (eventual outcomes) time series sample data (a one dimensional array).
[in] **Y** is the forecast time series data (a one dimensional array).
[in] **N** is the number of observations in X.
[in] **SMAPE** is a switch to select the return output (FALSE=MAPE (default), TRUE=Symmetric MAPE (SMAPI)).
[out] **retVal** is the calculated value of this function.

Remarks

1. MAPE is also referred to as MAPD.
2. The time series is homogeneous or equally spaced.
3. For a plain MAPE calculation, in the event that an observation value (i.e. \(x_k\)) is equal to zero, the MAPE function skips that data point.
4. The mean absolute percentage error (MAPE), also known as mean absolute percentage deviation (MAPD), measures the accuracy of a method for constructing fitted time series values in statistics.
5. The two time series must be identical in size.
6. The mean absolute percentage error (MAPE) is defined as follows:
\[\mathrm{MAPE}=\frac{100}{N}\times \sum_{i=1}^N \left | \frac{x_i - \hat x_i}{x_i} \right |\],

>where:

- $\{x_i\}$ is the actual observations time series.
- $\{\hat x_i\}$ is the estimated or forecasted time series.
- N is the number of non-missing data points.

7. When calculating the average MAPE for a number of time series, you may encounter a problem: a few of the series that have a very high MAPE might distort a comparison between the average MAPE of a time series fitted with one method compared to the average MAPE when using another method.
8. In order to avoid this problem, other measures have been defined, for example the SMAPE (symmetrical MAPE), weighted absolute percentage error (WAPE), real aggregated percentage error and relative measure of accuracy (ROMA).
9. The symmetrical mean absolute percentage error (SMAPE) is defined as follows:
$$\mathrm{SMAPE}=\frac{200}{N}\times \sum_{i=1}^N \left | \frac{x_i - \hat x_i}{x_i+\hat x_i} \right |$$
10. The SMAPE is easier to work with than MAPE, as it has a lower bound of 0% and an upper bound of 200%.
11. The SMAPE does not treat over-forecast and under-forecast equally.
12. For a SMAPE calculation, in the event the sum of the observation and forecast values (i.e. $x_k + \hat x_k$) equals zero, the MAPE function skips that data point.

Requirements

Header	SFSDK.H
Library	SFSDK.LIB
DLL	SFSDK.DLL

Examples

```
int NDK_MAPE(double[]  pData1,
             double[]  pData2,
             UIntPtr   nSize,
             short     retType,
             ref double retVal
            )
```

Namespace: NumXLAPI
Class: SFSDK
Scope: Public
Lifetime: Static

Calculates the mean absolute percentage error (deviation) function for the forecast and the eventual outcomes.

Return Value

a value from **NDK_RETCODE** enumeration for the status of the call.

NDK_SUCCESS operation successful
Error Error Code

Parameters

[in] **pData1** is the original (eventual outcomes) time series sample data (a one dimensional array).
[in] **pData2** is the forecast time series data (a one dimensional array).
[in] **nSize** is the number of observations in pData1.
[in] **retType** is a switch to select the return output (FALSE=MAPE (default), TRUE=Symmetric MAPE (SMAPI)).
[out] **retVal** is the calculated value of this function.

Remarks

1. MAPE is also referred to as MAPD.
2. The time series is homogeneous or equally spaced.
3. For a plain MAPE calculation, in the event that an observation value (i.e. \(x_k\)) is equal to zero, the MAPE function skips that data point.
4. The mean absolute percentage error (MAPE), also known as mean absolute percentage deviation (MAPD), measures the accuracy of a method for constructing fitted time series values in statistics.
5. The two time series must be identical in size.
6. The mean absolute percentage error (MAPE) is defined as follows:
\[\mathrm{MAPE}=\frac{100}{N}\times \sum_{i=1}^N \left | \frac{x_i - \hat x_i}{x_i} \right |\],
>where:

- \(\{x_i\}\) is the actual observations time series.
- \(\{\hat x_i\}\) is the estimated or forecasted time series.
- \(N\) is the number of non-missing data points.

7. When calculating the average MAPE for a number of time series, you may encounter a problem: a few of the series that have a very high MAPE might distort a comparison between the average MAPE of a time series fitted with one method compared to the average MAPE when using another method.
8. In order to avoid this problem, other measures have been defined, for example the SMAPE (symmetrical MAPE), weighted absolute percentage error (WAPE), real aggregated percentage error and relative measure of accuracy (ROMA).
9. The symmetrical mean absolute percentage error (SMAPE) is defined as follows:

\[\mathrm{SMAPE}=\frac{200}{N}\times \sum_{i=1}^N \left | \frac{x_i - \hat x_i}{x_i+\hat x_i} \right |\]

10. The SMAPE is easier to work with than MAPE, as it has a lower bound of 0% and an upper bound of 200%.
11. The SMAPE does not treat over-forecast and under-forecast equally.
12. For a SMAPE calculation, in the event the sum of the observation and forecast values (i.e. \(x_k + \hat x_k\)) equals zero, the MAPE function skips that data point.

Exceptions

Exception Type	Condition
None	N/A

Requirements

Namespace	NumXLAPI
Class	SFSDK
Scope	Public
Lifetime	Static
Package	NumXLAPI.DLL

Examples

References

Hull, John C.; Options, Futures and Other DerivativesFinancial Times/ Prentice Hall (2011), ISBN 978-0132777421

See Also

[template("related")]