NDK_COLNRTY_TEST

Last Modified on 04/20/2016 12:38 pm CDT

- C/C++
- .Net

```
int __stdcall NDK_COLNRTY_TEST(double **          XX,
                               size_t              N,
                               size_t              M,
                               LPBYTE              mask,
                               size_t              nMaskLen,
                               COLNRTY_TEST_TYPE   nMethod,
                               WORD                nColIndex,
                               double *            retVal
                              )
```

Returns the collinearity test statistics for a set of input variables.

Returns

status code of the operation

Return values

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

Parameters

[in] **XX** is the input variables matrix data (two dimensional).

[in] **N** is the number of rows (observations) in XX.

[in] **M** is the number of columns (variables) in XX.

[in] **mask** is the boolean array to select a subset of the input variables in X. If NULL, all variables in X are included.

[in] **nMaskLen** is the number of elements in the mask. Must be zero or equal to M.

[in] **nMethod** is the multi-colinearity measure to compute

Method	Value	Description
COLNRTY_CN	1	Condition Number.
COLNRTY_VIF	2	Variation Inflation Factor (VIF)
COLNRTY_DET	3	Determinant
COLNRTY_EIGEN	4	Eigenvalues

[in] **nColIndex** is a switch to designate the explanatory variable to examine (not required for condition number).

[out] **retVal** is the calculated statistics of collinearity.

Remarks

- The sample data may include missing values.

- Each column in the input matrix corresponds to a separate variable.
- Each row in the input matrix corresponds to an observation.
- Observations (i.e. row) with missing values are removed.
- In the variance inflation factor (VIF) method, a series of regressions models are constructed, where one variable is the dependent variable against the remaining predictors.
- \[\textrm{Tolerance}_i = 1-R_i^2\] \[\textrm{VIF}_i =\frac{1}{\textrm{Tolearance}_i} = \frac{1}{1-R_i^2}\] Where:
  - \(R_i^2\) is the coefficient of determination of a regression of explanator \(i\) on all the other explanators.
- A tolerance of less than 0.20 or 0.10 and/or a VIF of 5 or 10 and above indicates a multicollinearity problem.
- The condition number (\(\kappa\)) test is a standard measure of ill-conditioning in a matrix; It will indicate that the inversion of the matrix is numerically unstable with finite-precision numbers (standard computer floats and doubles).
- \[X = \begin{bmatrix} 1 & X_{11} & \cdots & X_{k1} \\ \vdots & \vdots & & \vdots \\ 1 & X_{1N} & \cdots & X_{kN} \end{bmatrix} \] \[\kappa = \sqrt{\frac{\lambda_{max}}{\lambda_{min}}}\] Where:
  - \(\lambda_{max}\) is the maximum eigenvalue.
  - \(\lambda_{min}\) is the minimum eigenvalue.
- As a rule of thumb, a condition number (κ) greater or equal to 30 indicates a severe multi-collinearity problem.
- The CollinearityTest function is available starting with version 1.60 APACHE.

Requirements

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

Examples

```
int NDK_COLNRTY_TEST(ref UIntPtr        pData,
                         UIntPtr        nSize,
                         UIntPtr        nVars,
```

Namespace: NumXLAPI
Class: SFSDK
Scope: Public

```
                Byte[]              mask,
                UIntPtr             nMaskLen,
                COLNRTY_TEST_TYPE   nMethod,
                short               nColIndex,
                ref double          retVal
              )
```

Lifetime: Static

Returns the collinearity test statistics for a set of input variables.

Returns

status code of the operation

Return values

NDK_SUCCESS Operation successful

NDK_FAILED Operation unsuccessful. See Macros for full list.

Parameters

[in] **XX** is the input variables matrix data (two dimensional).

[in] **N** is the number of rows (observations) in XX.

[in] **M** is the number of columns (variables) in XX.

[in] **mask** is the boolean array to select a subset of the input variables in X. If NULL, all variables in X are included.

[in] **nMaskLen** is the number of elements in the mask. Must be zero or equal to M.

[in] **nMethod** is the multi-colinearity measure to compute

Method	Value	Description
COLNRTY_CN	1	Condition Number.
COLNRTY_VIF	2	Variation Inflation Factor (VIF)
COLNRTY_DET	3	Determinant
COLNRTY_EIGEN	4	Eigenvalues

[in] **nColIndex** is a switch to designate the explanatory variable to examine (not required for condition number).

[out] **retVal** is the calculated statistics of collinearity.

Remarks

- The sample data may include missing values.
- Each column in the input matrix corresponds to a separate variable.
- Each row in the input matrix corresponds to an observation.
- Observations (i.e. row) with missing values are removed.
- In the variance inflation factor (VIF) method, a series of regressions models are constructed, where one variable is the dependent variable against the remaining predictors.
- \[\textrm{Tolerance}_i = 1-R_i^2\] \[\textrm{VIF}_i =\frac{1}{\textrm{Tolearance}_i} = \frac{1}{1-R_i^2}\] Where:
    - \(R_i^2\) is the coefficient of determination of a regression of explanator \(i\) on all the other explanators.

- A tolerance of less than 0.20 or 0.10 and/or a VIF of 5 or 10 and above indicates a multicollinearity problem.
- The condition number (\(\kappa\)) test is a standard measure of ill-conditioning in a matrix; It will indicate that the inversion of the matrix is numerically unstable with finite-precision numbers (standard computer floats and doubles).
- \[X = \begin{bmatrix} 1 & X_{11} & \cdots & X_{k1} \\ \vdots & \vdots & & \vdots \\ 1 & X_{1N} & \cdots & X_{kN} \end{bmatrix} \] \[\kappa = \sqrt{\frac{\lambda_{max}}{\lambda_{min}}}\] Where:
  - \(\lambda_{max}\) is the maximum eigenvalue.
  - \(\lambda_{min}\) is the minimum eigenvalue.
- As a rule of thumb, a condition number (κ) greater or equal to 30 indicates a severe multi-collinearity problem.
- The CollinearityTest function is available starting with version 1.60 APACHE.

Exceptions

Exception Type	Condition
None	N/A

Requirements

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

Examples

References

Hamilton, J .D.; Time Series Analysis , Princeton University Press (1994), ISBN 0-691-04289-6

Tsay, Ruey S.; Analysis of Financial Time Series John Wiley & SONS. (2005), ISBN 0-471-690740

See Also

[template("related")]