Package ‘cvq2’

March 13, 2013

Type Package

Title Calculate the predictive squared correlation coefficient

Version 1.1.0

Date 2013-03-13

Author Torsten Thalheim

Maintainer Torsten Thalheim <torstenthalheim@gmx.de>

Description The external prediction capability of quantitative structure-activity relationship (QSAR) models is often quantified using the predictive squared correlation coefficient. This value can be calculated with an external data set or by cross validation.

Depends stats, methods

License GPL-3

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2013-03-13 07:44:44

R topics documented:

- cvq2-package ........................................ 2
- cvq2-class .......................................... 4
- cvq2.setA .......................................... 5
- cvq2.setA_pred .................................... 6
- cvq2.setB .......................................... 7
- q2 .................................................... 8
- q2-class ........................................... 10

Index 12
\textbf{cvq2-package} \hspace{1cm} \textit{Calculate the predictive squared correlation coefficient.}

\section*{Description}
This package calculates the predictive squared correlation coefficient, $q^2$, in comparison to the well known conventional squared correlation coefficient, $r^2$. For a given model $M$, $q^2$ indicates the prediction performance of $M$, whereas $r^2$ is a measure for its calibration performance.

\section*{Details}

Package: cvq2  
Type: Package  
Version: 1.1.0  
Date: 2013-03-13  
Depends: stats  
License: GPL v3  
LazyLoad: yes

The calculation procedure is as follows:  
The model $M$ is described as a data set, where the parameters $x_1 \ldots x_n$ describe an observation $y$.  
First, a general linear regression is applied to $M$. Therewith, the conventional squared correlation coefficient, $r^2$, can be calculated:

$$
r^2 = 1 - \frac{\sum_{i=1}^{N} \left( y_i^{\text{fit}} - y_i \right)^2}{\sum_{i=1}^{N} \left( y_i - y_{\text{mean}} \right)^2} \equiv 1 - \frac{\text{RSS}}{\text{SS}}
$$

The denominator complies with the Residual Sum of Squares RSS, the difference between the fitted values $y_i^{\text{fit}}$ and the observed values $y_i$. The numerator is the Sum of Squares, SS, and refers to the difference between the observed values $y_i$ and their mean $y_{\text{mean}}$.

To compare the calibration of $M$ with its prediction power, $M$ is applied to an external data set. The comparison of the predicted values $y_i^{\text{pred}}$ with the observed values $y_i$ leads to the predictive squared correlation coefficient, $q^2$:

$$
q^2 = 1 - \frac{\sum_{i=1}^{N} \left( y_i^{\text{pred}} - y_i \right)^2}{\sum_{i=1}^{N} \left( y_i - y_{\text{mean}} \right)^2} \equiv 1 - \frac{\text{PRESS}}{\text{SS}}
$$

The Predictive residual Sum of Squares PRESS is the difference between the predicted values $y_i^{\text{pred}}$ and the observed values $y_i$. The Sum of Squares SS refers to the difference between the observed values $y_i$ and their mean $y_{\text{mean}}$. 
To avoid any bias, $y_{mean}$ is the arithmetic mean of the $y_i$ from the external data set. Hence the clarifying $q_{tr}^2$ equation is slightly different to the previous $q^2$ equation:

$$q_{tr}^2 = 1 - \frac{\sum_{i=1}^{N} (y_{i}^{pred} - y_i)^2}{\sum_{i=1}^{N} (y_i - y_{mean})^2}$$

The arithmetic mean of the observed values in the external data set, $y_{training}^{mean}$, is used to determine the prediction performance, $q_{tr}^2$, of $M$.

In case, that no external data set is available, one can perform a cross-validation to evaluate the prediction performance. The cross-validation splits the model data set ($N$ elements) into a training set ($N - k$ elements) and a test set ($k$ elements). Each training set yields to an individual model $M'$, which is used to predict the missing $k$ value(s). Each model $M'$ is slightly different to $M$. At least, any observed value is predicted once and the comparison between the observation and the prediction yields to $q_{cv}^2$:

$$q_{cv}^2 = 1 - \frac{\sum_{i=1}^{N} (y_{i}^{pred(N-k)} - y_i)^2}{\sum_{i=1}^{N} (y_i - y_{N-k,i}^{mean})^2}$$

The arithmetic mean used in this equation, $y_{N-k,i}^{mean}$, is individually for any test set and calculated for the observed values comprised in the training set.

If $k > 1$, the compilation of training and test set may have impact on the calculation of the predictive squared correlation coefficient. To overcome biasing, one can repeat this calculation with various compilations of training and test set. Thus, any observed value is predicted several times, according to the number of runs performed.

Remark, if the prediction performance is evaluated with cross-validation, the calculation of the predictive squared correlation coefficient, $q^2$, is more accurate than the calculation of the conventional squared correlation coefficient, $r^2$.

**Note**

The package development started few years ago in the Ecological Chemistry Department during my time at the Helmholtz Centre for Environmental Research in Leipzig. Thereby it is based on Schüürmann et al. 2008, External validation and prediction employing the predictive squared correlation coefficient - test set activity mean vs training set activity mean.

**Author(s)**

Torsten Thalheim <torstenthalheim@gmx.de>

**References**


Examples

```r
library(cvq2)
data(cvq2.setA)
result <- cvq2( cvq2.setA, y ~ x1 + x2 )
result

data(cvq2.setB)
result <- cvq2( cvq2.setB, y ~ x, nFold = 3 )
result

data(cvq2.setB)
result <- cvq2( cvq2.setB, y ~ x, nFold = 3, nRun = 5 )
result

data(cvq2.setA)
data(cvq2.setA_pred)
result <- q2( cvq2.setA, cvq2.setA_pred, y ~ x1 + x2 )
result
```

---

cqv2-class

Class "cvq2"

Description

The class "cvq2" extends class "q2" and is used to store information about the model calibration and its prediction performance. The prediction performance is calculated with a cross-validation.

Objects from the Class

Objects can be created by calls of the form `new("cvq2", ...)`.

Slots

- **result**: Contains three lists (fit, pred, cv) regarding the results from linear regression (model calibration, fit) and cross-validation (prediction power, pred and cv) for the given model.
- **output**: A list of parameters like number formats, output restrictions or output targets.
**Linear regression and prediction result list:**
These lists are inherited from the parent class `q2`. Differences caused by cross-validation appear in the prediction result list for:

- `datatable` Additionaly, for each observed value the model parameters used for prediction are stored, as well as the arithmetic mean of the training set

- `nTrainingSet` The number of elements in one training set \((N - k)\) plus an eventually variation.

- `nTestSet` The number of elements in one test set \((k)\) minus an eventually variation.

**Cross-validation result list:**

- `decimalSplit True`, if some test sets consist of \(k - 1\) elements.

- `nFold` The model data set (modelData) is randomly partitioned into \(n\) equal sized (according to `decimalSplit`) test sets for each individual run.

- `nRun` The number of runs each value is predicted.

**Extends**

Class "`q2`", directly.

**Methods**

- `show` Returns a comprehensive overview about the model calibration and the prediction performance.

**Author(s)**

Torsten Thalheim <torstenthalheim@gmx.de>

**Examples**

`showClass("cvq2")`

---

```r
data(cvq2.setA)
```

---

**Description**

Contains a small data set with four observations, the observed value \(y\) depends on two parameters \((x_1, x_2)\).

**Usage**

`data(cvq2.setA)`
**Format**

A data frame with four observations. Each row contains two parameters and the observed value.

- `x1` parameter 1
- `x2` parameter 2
- `y` observed value

**Details**

This data set can be used to demonstrate the differences between the model calibration and the prediction power. The prediction power can be determined either with cross-validation or the application of the model to the data set `cvq2.setA_pred`.

**Note**

This data set contains one outlier (row 2). If the prediction power is determined with cross-validation, this outlier leads to a considerably decreased prediction power, $q_c^2$, compared to the model calibration, $r^2$. For this data set, one can perform a Leave-One-Out cross-validation only.

**Source**

Generic data set, created for this purpose only.

**Examples**

```r
data(cvq2.setA)
```

<table>
<thead>
<tr>
<th>cvq2.setA_pred</th>
<th>Prediction set for model set <code>cvq2.setA</code>.</th>
</tr>
</thead>
</table>

**Description**

This data set can be used to determine the prediction power of the model `cvq2.setA`.

**Usage**

```r
data(cvq2.setA_pred)
```

**Format**

A data frame with four observations. Each row contains two parameters and the observed value.

- `x1` parameter 1
- `x2` parameter 2
- `y` observed value
The prediction set fits very good to the model set \( \text{cvq2.setA} \), \( q^2 \) is as high as \( r^2 \).

**Source**

Generic data set, created for this purpose only.

**Examples**

```r
data(cvq2.setA_pred)
```

**Details**

Small data set to demonstrate the difference between the conventional and the predictive squared correlation coefficient while performing a cross-validation.

**Description**

Contains a small data set with six observations, the observed value \( y \) depends on the parameter \( x \).

**Usage**

```r
data(cvq2.setB)
```

**Format**

A data frame with 6 observations and one parameter per observation.

- \( x \) parameter
- \( y \) observation

**Details**

The prediction power can be determined with cross-validation. The cross-validation can be performed as Leave-One-Out (\( n\text{fold} = N = 6 \)) or as k-fold (\( n\text{fold} = 2, 3 \)). If \( n\text{fold} = 2, 3 \), modelData is randomly split into \( n\text{fold} \) disjunct and equal sized (test) sets.

Furthermore, in that case one has the opportunity to repeat the cross-validation, while each run (\( n\text{Run} = 2, \ldots, x \)) has an individual test set compilation.

The prediction power, \( q^2_{cv} \), calculated for this data set is considerably smaller than the model calibration, \( r^2 \), promises.

**Source**

Generic data set, created for this purpose only.

**Examples**

```r
data(cvq2.setB)
```
Model prediction power calculation.

Description

Determines the prediction power of a model. Therefore the model is applied to an external data set, and its observations are compared to the model predictions. If an external data set is not available, the prediction power is calculated while performing a cross-validation to the model data set.

Usage

\[
\begin{align*}
\text{loq2( modelData, formula = NULL, round = 4, extOut = FALSE, extOutFile = NULL )} \\
\text{cvq2( modelData, formula = NULL, nFold = N, nRun = 1, round = 4, extOut = FALSE, extOutFile = NULL )} \\
\text{q2( modelData, predictData, formula = NULL, round = 4, extOut = FALSE, extOutFile = NULL )}
\end{align*}
\]

Arguments

- **modelData**: The model data set consists of parameter \(x_1, x_2, \ldots, x_n\) and an observation \(y\)
- **predictData**: The prediction data set consists of parameter \(x_1, x_2, \ldots, x_n\) and an observation \(y\)
- **formula**: The formula used to predict the observed value, like \(y \sim x_1 + x_2 + \ldots + x_n\)
  - DEFAULT: NULL
  - If NULL, a generic formula is derived from the data set, assuming that the last column contains the observed value
- **nFold**: The model data set `modelData` is randomly partitioned into \(n\) equal sized subsets (test sets) during each run of cross-validation, DEFAULT: \(N\)
  - \(2 \leq nFold \leq N\)
- **nRun**: Number of iterations, the cross-validation is applied to the data set. This corresponds to the number of individual predictions per observed value, DEFAULT: 1
  - 1 \(\leq nRun\)
- **round**: The rounding value used in the output, DEFAULT: 4
- **extOut**: Extended output, DEFAULT: FALSE
  - If `extOutFile` is not specified, write to stdout()
- **extOutFile**: Write extended output into file (implies `extOut` = TRUE), DEFAULT: NULL
Details

The calibration of `modelData`, including the conventional squared correlation coefficient, \( r^2 \), is calculated with a linear regression.

**q2()-method:**

*Alias: qsq(), qsquare()*

The model described by `modelData` is used to predict the observations of `predictData`. These predictions are used in the \( q^2 \) equation to calculate the predictive squared correlation coefficient.

**cvq2()-method:**

*Alias: cvqsq(), cvqsquare()*

A cross-validation is performed for `modelData`, whereas `modelData` (\( N \) elements) is split into \( nFold \) disjunct and equal sized test sets (subsets).

Each test set consists of \( k \) elements:

\[
 k = \left\lfloor \frac{N}{nFold} \right\rfloor
\]

In case, \( \frac{N}{nFold} \) is a decimal number, some test sets consist of \( k - 1 \) elements. The remaining \( N - k \) elements are merged together as training set for this test set and describe the model \( M' \). This model is used to predict the observations in the test set. Note, that \( M' \) is slightly different compared to the model \( M \) for the \( r^2 \)-calculation, which is a result of the missing \( k \) values.

Each observation from `modelData` is predicted once. The difference between the prediction and the observation within the test sets is used to calculate the PREdictive residual Sum of Squares (PRESS). Furthermore for any training set, the mean of the observed values, \( y^{N-k,i}_{mean} \), is calculated. With PRESS and \( y^{N-k,i}_{mean} \), the modified \( q^2_{cv} \) equation is used to calculate the predictive squared correlation coefficient.

In case \( k > 1 \) one can repeat the cross-validation to overcome biasing. Therefore, in each iteration (\( nRun = 1 \ldots x \)), the test sets are compiled individually by random. Within one iteration, each observation is predicted once. If \( nFold = N \), one need one iteration only.

**looq2()-method:**

Same procedure as cvq2()-method (see above), but implicit \( nFold = N \) to perform a Leave-One-Out cross-validation. For Leave-One-Out cross-validation one need one iteration (\( nRun = 1 \)) only.

**Value**

**q2()-method:**

The method `q2` returns an object of class "`q2`". It contains information about the model calibration and its prediction performance on the external data set.

**cvq2() - method, looq2() - method:**

The methods `cvq2` and `looq2` return an object of class "`cvq2`". It contains information about the model calibration and its prediction performance described by the model data set. Furthermore this object contains data about the cross-validation applied to the model data set.
Author(s)
Torsten Thalheim <torstenthalheim@gmx.de>

Examples

```r
library(cvq2)
data(cvq2.setA)
result <- cvq2(cvq2.setA, y ~ x1 + x2)
result

data(cvq2.setB)
result <- cvq2(cvq2.setB, y ~ x, nFold = 3)
result

data(cvq2.setB)
result <- cvq2(cvq2.setB, y ~ x, nFold = 3, nRun = 5)
result

data(cvq2.setA)
result <- looq2(cvq2.setA, y~x1+x2)
result

data(cvq2.setA)
data(cvq2.setA_pred)
result <- q2(cvq2.setA, cvq2.setA_pred, y~x1+x2)
result
```

---

q2-class

Class "q2"

Description

The class "q2" is used to store information about the model calibration and its prediction performance demonstrated by an external data set.

Objects from the Class

Objects can be created by calls of the form `new("q2", ...)`. 

Slots

- `result` Contains two lists (fit, pred) regarding the results from linear regression (model calibration, fit) and the application of the model to a validation set (prediction power, pred)
- `output` A list of parameters like number formats, output restrictions or output targets

Linear regression result list:

- `datatable` The observed and predicted values
**datatable_columns** The explanation of the datatable’s column names

**model** The linear regression model

**n** The number of elements in the data set

**observed_mean** The arithmetic mean of the observed values

**r2** The conventional squared correlation coefficient

**rmse** The root mean square error

**Prediction result list:**

**datatable** Contains the observed value and its prediction by the model

**datatable_columns** The explanation of the datatable’s column names

**nTrainingSet** The number of elements in the model set \((N - k)\)

**nTestSet** The number of elements in the prediction set \((k)\)

**q2** The predictive squared correlation coefficient

**rmse** The root mean square error is calculated with Bessel’s sample covariance correction, using \(N - 1\) in the denominator instead \(N\)

**Methods**

**show** Returns a comprehensive overview about the model calibration and the prediction performance.

**Author(s)**

Torsten Thalheim <torstenthalheim@gmx.de>

**Examples**

```
showClass("q2")
```
Index

*Topic **classes**
  cvq2-class, 4
  q2-class, 10
*Topic **cross validation**
  q2, 8
*Topic **cross-validation**
  q2, 8
*Topic **prediction power**
  q2, 8
*Topic **predictive squared correlation coefficient**
  cvq2-package, 2
  q2, 8
*Topic **q square**
  cvq2-package, 2
  q2, 8
*Topic **q^2**
  cvq2-package, 2
  q2, 8
*Topic **cvq2**
  cvq2, 9
  cvq2 (q2), 8
  cvq2-class, 4
  cvq2-package, 2
  cvq2.setA, 5, 6, 7
  cvq2.setA_pred, 6, 6
  cvq2.setB, 7
  cvqsq (q2), 8
  cvqsquare (q2), 8

looq2, 9
looq2 (q2), 8

q2, 4, 5, 8, 9
q2-class, 10
qsq (q2), 8
qsquare (q2), 8

show, cvq2-method (cvq2-class), 4

show, q2-method (q2-class), 10