

# Package ‘CCWeights’

May 7, 2026

**Type** Package

**Title** Perform Weighted Linear Regression for Calibration Curve

**Version** 0.1.6

**Author** Yonghui Dong

**Maintainer** Yonghui Dong <yonghui.dong@gmail.com>

**Description** Automated assessment and selection of weighting factors for accurate quantification using linear calibration curve.

In addition, a 'shiny' App is provided, allowing users to analyze their data using an interactive graphical user interface, without any programming requirements.

**Depends** R (>= 3.5.0)

**Imports** plotly, dplyr, stats, magrittr, shiny, bs4Dash, fresh, DT, tools, readxl, rmarkdown, readr

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.2

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2021-11-09 21:40:02 UTC

## Contents

doCalibration . . . . .	2
doEvaluation . . . . .	2
doFtest . . . . .	3
doWlm . . . . .	4
expData . . . . .	5
runGui . . . . .	5
<b>Index</b>	<b>6</b>

doCalibration                      *Perform Calibration*

---

**Description**

Perform calibration

**Usage**

```
doCalibration(DF, weights = NULL)
```

**Arguments**

DF	data frame, it must contain a column named 'Concentration' and a column named 'Response'
weights	default is NULL

**Value**

dataframe, the quantification result

**Author(s)**

Yonghui Dong

**Examples**

```
Concentration <- rep(c(10, 50, 100, "unknown"), each = 3)
Response <- c(133, 156, 177, 6650, 7800, 8850, 13300, 15600, 17700, 156, 1450, 1400)
DF <- cbind.data.frame(Concentration = Concentration, Response = Response)
result <- doCalibration(DF)
```

---

doEvaluation                      *Evaluate Different Weighting Factors*

---

**Description**

Evaluate different weighting factors.

**Usage**

```
doEvaluation(DF, p = 0.05, userWeights = NULL)
```

**Arguments**

DF                    data frame, it must contain a column named 'Concentration' and a column named 'Response'

p                     p-value, default is 0.05

userWeights        user defined weights in linear regression, default is NULL. User can easily define weights, e.g., "1/x", "1/x^2", "1/y"

**Value**

dataframe, weighting factor evaluation result

**Author(s)**

Yonghui Dong

**Examples**

```
Concentration <- rep(c(10, 50, 100, 500), each = 3)
Response <- c(133, 156, 177, 1300, 1450, 1600, 4000, 3881, 3700, 140000, 139000, 140000)
DF <- cbind.data.frame(Concentration = Concentration, Response = Response)
result <- doEvaluation(DF)
```

---

doFtest

*Perform F Test*


---

**Description**

perform F test to evaluate homoscedasticity.

**Usage**

```
doFtest(DF, p = 0.01, lower.tail = FALSE)
```

**Arguments**

DF                    data frame, it must contain a column named 'Concentration' and a column named 'Response'

p                     p-value

lower.tail         default is FALSE

**Value**

dataframe, F test result

**Author(s)**

Yonghui Dong

## Examples

```
Concentration <- rep(c(10, 50, 100, 500), each = 3)
Response <- c(133, 156, 177, 1300, 1450, 1600, 4000, 3881, 3700, 140000, 139000, 140000)
DF <- cbind.data.frame(Concentration, Response)
result <- doFtest(DF, p = 0.01)
```

---

doWlm

*Perform Weighted Linear Regression*

---

## Description

Perform weighted linear regression and evaluate by using summed residual.

## Usage

```
doWlm(DF, weights = NULL)
```

## Arguments

DF	data frame, it must contain a column named 'Concentration' and a column named 'Response'
weights	the weights used in linear regression, default is NULL. User can easily define weights, e.g., "1/x", "1/x^2", "1/y"

## Value

list, weighted linear regression result

## Author(s)

Yonghui Dong

## Examples

```
Concentration <- rep(c(10, 50, 100, 500), each = 3)
Response <- c(133, 156, 177, 1300, 1450, 1600, 4000, 3881, 3700, 140000, 139000, 140000)
DF <- cbind.data.frame(Concentration = Concentration, Response = Response)
result <- doWlm(DF, weights = "1/x^2")
```

---

expData	<i>expData</i>
---------	----------------

---

**Description**

Two example data set: one with internal standards (IS), and one without IS

**Usage**

```
expData
```

**Format**

A list with 2 data frames:

**noSTD** the example data without IS

**STD** the example data with IS

---

runGui	<i>Run CCWeights Gui</i>
--------	--------------------------

---

**Description**

Run CCWeights Gui.

**Usage**

```
runGui()
```

**Value**

Gui

**Author(s)**

Yonghui Dong

**Examples**

```
if(interactive()){}
```

# Index

\* **datasets**

expData, [5](#)

doCalibration, [2](#)

doEvaluation, [2](#)

doFtest, [3](#)

doWlm, [4](#)

expData, [5](#)

runGui, [5](#)