

# Package: ISO6976.2016 (via r-universe)

May 10, 2026

**Version** 0.1-0

**Date** 2026-03-28

**Title** Calorific Values and Properties of Natural Gas per ISO 6976:2016

**Depends** R (>= 3.5.0)

**LinkingTo** Rcpp

**Imports** Rcpp (>= 1.0), R6

**Suggests** knitr, lintr, rmarkdown, testthat (>= 3.0.0)

**Encoding** UTF-8

**LazyData** false

**ByteCompile** true

**Description** Calculates calorific values (gross and net), density, relative density, and Wobbe indices together with their standard uncertainties from natural gas composition, implementing the method of ISO 6976:2016 ``Natural Gas — Calculation of calorific values, density, relative density and Wobbe indices from composition''. Uncertainty propagation follows Annex B of that standard. Reference: International Organization for Standardization (2016)  
<<https://www.iso.org/standard/55842.html>>.

**License** GPL (>= 3)

**URL** [https://github.com/RuedigerForster/ISO\\_6976](https://github.com/RuedigerForster/ISO_6976)

**BugReports** [https://github.com/RuedigerForster/ISO\\_6976/issues](https://github.com/RuedigerForster/ISO_6976/issues)

**RoxygenNote** 7.3.3

**VignetteBuilder** knitr

**Repository** <https://ruedigerforster.r-universe.dev>

**Date/Publication** 2026-04-10 06:35:17 UTC

**RemoteUrl** [https://github.com/ruedigerforster/iso\\_6976](https://github.com/ruedigerforster/iso_6976)

**RemoteRef** HEAD

**RemoteSha** b300c50f2d5fb2b078d3c146a91e527ab3ef940c

## Contents

ISO6976.2016-package . . . . .	2
calculateProperties . . . . .	3
componentIndex . . . . .	4
componentName . . . . .	5
componentNames . . . . .	5
example1 . . . . .	6
example2 . . . . .	6
example3 . . . . .	7
example3_ex . . . . .	7
GasComponents . . . . .	8
iso6976_calc . . . . .	10

<b>Index</b>	<b>12</b>
--------------	-----------

---

ISO6976.2016-package	<i>ISO6976.2016: Calorific Values and Properties of Natural Gas per ISO 6976:2016</i>
----------------------	---

---

## Description

Calculates calorific values (gross and net, molar, mass and volumetric bases), density, relative density, and Wobbe indices together with their standard uncertainties from natural gas composition, following ISO 6976:2016 "Natural Gas — Calculation of calorific values, density, relative density and Wobbe indices from composition".

## Details

Uncertainty propagation is implemented according to Annex B of that standard (variance-covariance method).

\*\*Application restrictions (ISO 6976:2016 §5):\*\* \* Combustion temperature: 0, 15, 15.55 (60 °F), 20, or 25 °C. \* Volume reference temperature: 0, 15, 15.55 (60 °F), or 20 °C. \* Reference pressure: 90–110 kPa. \* Compression factor Z must be > 0.9.

## Author(s)

Rüdiger Forster <meticulous.measurements@gmail.com>

## References

ISO 6976:2016 "Natural Gas — Calculation of calorific values, density, relative density and Wobbe indices from composition".

## See Also

Useful links:

- [https://github.com/RuedigerForster/ISO\\_6976](https://github.com/RuedigerForster/ISO_6976)
- Report bugs at [https://github.com/RuedigerForster/ISO\\_6976/issues](https://github.com/RuedigerForster/ISO_6976/issues)

---

calculateProperties     *Calculate natural gas properties per ISO 6976:2016*

---

## Description

Computes all combustion and volumetric properties of a natural gas mixture together with their standard (or expanded) uncertainties, following the method and tables of ISO 6976:2016 and the uncertainty propagation formulae of its Annex B.

## Usage

```
calculateProperties(  
  compositionArray,  
  uncertaintyArray,  
  correlationMatrix,  
  combustionTemperature = 25,  
  volumeTemperature = 15,  
  pressure = 101.325,  
  coverage = 1  
)
```

## Arguments

compositionArray	Numeric vector of length 60: mole fractions $\backslash[\text{mol/mol}\backslash]$ in the component order of ISO 6976:2016 Table A.2. See <a href="#">componentNames</a> for the ordering. If values are given in mol%, divide by 100 before passing them here.
uncertaintyArray	Numeric vector of length 60: standard uncertainties of the mole fractions (same units as compositionArray).
correlationMatrix	60x60 numeric matrix of correlation coefficients between component mole fractions. Use <code>diag(60)</code> when correlations are unknown or assumed to be zero.
combustionTemperature	Combustion reference temperature in °C. Permitted values: 0, 15, 15.55, 20, 25. Default: 25.
volumeTemperature	Volume reference temperature in °C. Permitted values: 0, 15, 15.55, 20. Default: 15.
pressure	Reference pressure in kPa. Must be in $\backslash[90, 110\backslash]$ . Default: 101.325.
coverage	Coverage factor $k$ ; uncertainties are multiplied by $k$ before being returned. Default: 1 (standard uncertainty).

**Value**

A named list with the following elements (all numeric scalars):

Name	Description	Unit
M	Molar mass	kg/kmol
Z	Compression factor	—
G_o	Ideal-gas relative density	—
D_o	Ideal-gas density	kg/m <sup>3</sup>
G, u_G	Real-gas relative density and uncertainty	—
D, u_D	Real-gas density and uncertainty	kg/m <sup>3</sup>
Hcg, u_Hcg	Molar gross calorific value and uncertainty	kJ/mol
Hcn, u_Hcn	Molar net calorific value and uncertainty	kJ/mol
Hmg, u_Hmg	Mass-basis gross calorific value and uncertainty	MJ/kg
Hmn, u_Hmn	Mass-basis net calorific value and uncertainty	MJ/kg
Hvg_o, u_Hvg_o	Ideal-gas vol. gross CV and uncertainty	MJ/m <sup>3</sup>
Hvn_o, u_Hvn_o	Ideal-gas vol. net CV and uncertainty	MJ/m <sup>3</sup>
Hvg, u_Hvg	Real-gas vol. gross CV and uncertainty	MJ/m <sup>3</sup>
Hvn, u_Hvn	Real-gas vol. net CV and uncertainty	MJ/m <sup>3</sup>
Wg_o	Ideal-gas gross Wobbe index	MJ/m <sup>3</sup>
Wn_o	Ideal-gas net Wobbe index	MJ/m <sup>3</sup>
Wg, u_Wg	Real-gas gross Wobbe index and uncertainty	MJ/m <sup>3</sup>
Wn, u_Wn	Real-gas net Wobbe index and uncertainty	MJ/m <sup>3</sup>

**References**

ISO 6976:2016 "Natural Gas — Calculation of calorific values, density, relative density and Wobbe indices from composition"

**See Also**

[GasComponents](#), [componentNames](#)

**Examples**

```
data("example1")
res <- calculateProperties(example1$fractionArray, example1$uncertaintyArray,
                          example1$correlationMatrix,
                          combustionTemperature = 15, volumeTemperature = 15)
res$M      # molar mass [kg/kmol]
res$Hvg    # real-gas vol. gross CV [MJ/m^3]
res$u_Wg   # standard uncertainty of gross Wobbe index
```

---

componentIndex

*Return the index of a gas component by name*

---

**Description**

Return the index of a gas component by name

**Usage**

```
componentIndex(name)
```

**Arguments**

name                    Character string: component name (English). See [componentNames](#) for the full list.

**Value**

Integer index (1–60) into the composition vector.

---

componentName                    *Return the name of a gas component by index*

---

**Description**

Return the name of a gas component by index

**Usage**

```
componentName(index)
```

**Arguments**

index                    Integer 1–60.

**Value**

Character string with the English component name.

---

componentNames                    *Return the names of all 60 gas components*

---

**Description**

Returns the ordered character vector of all 60 natural gas components recognised by ISO 6976:2016 (Table A.2). The position of each name corresponds to the index used in `compositionArray`, `uncertaintyArray`, and `correlationMatrix`.

**Usage**

```
componentNames()
```

**Value**

Character vector of length 60.

**See Also**

[componentIndex](#), [componentName](#)

---

example1	<i>ISO 6976:2016 Annex D.2 — Example 1 (5-component mixture, 15/15 °C)</i>
----------	--

---

**Description**

A five-component natural gas mixture from Annex D.2 of ISO 6976:2016. Non-zero components: methane, ethane, propane, nitrogen, carbon dioxide.

**Format**

A named list with three elements:

**fractionArray** Numeric vector (length 60) of mole fractions [mol/mol].

**uncertaintyArray** Numeric vector (length 60) of standard uncertainties [mol/mol].

**correlationMatrix** 60×60 identity correlation matrix.

**Source**

ISO 6976:2016 Annex D.2, Table D.2.

---

example2	<i>ISO 6976:2016 Annex D.3 — Example 2 (mixture with water vapour, 15.55/15.55 °C)</i>
----------	--

---

**Description**

An eleven-component natural gas mixture including water vapour from Annex D.3 of ISO 6976:2016.

**Format**

A named list with three elements (same structure as [example1](#)).

**Source**

ISO 6976:2016 Annex D.3, Table D.3.

---

example3	<i>ISO 6976:2016 Annex D — Example 3 (11-component mixture, identity matrix)</i>
----------	--

---

**Description**

An eleven-component natural gas mixture from Annex D of ISO 6976:2016 with an identity correlation matrix (no inter-component correlations). Used to verify calculations at 15/15 °C and 25/0 °C.

**Format**

A named list with three elements (same structure as [example1](#)).

**Source**

ISO 6976:2016 Annex D.

---

example3_ex	<i>ISO 6976:2016 Annex D — Example 3, full correlation matrix</i>
-------------	---

---

**Description**

The same eleven-component mixture as [example3](#) but with a full inter-component correlation matrix from a GC calibration covariance analysis, as given in Annex D of ISO 6976:2016.

**Format**

A named list with three elements (same structure as [example1](#)).

**Source**

ISO 6976:2016 Annex D (extended correlation case).

---

GasComponents

*R6 class for natural gas composition input*

---

## Description

R6 class for natural gas composition input

R6 class for natural gas composition input

## Details

GasComponents holds the mole fractions, their standard uncertainties, and the inter-component correlation matrix for a natural gas mixture. All three quantities are required as input to [calculateProperties](#).

Components are identified either by their integer index (1–60, matching ISO 6976:2016 Table A.2) or by English name (see [componentNames](#)).

## Public fields

`fractions` Numeric vector (length 60) of mole fractions [mol/mol].

`uncertainties` Numeric vector (length 60) of standard uncertainties.

`correlations` 60x60 correlation matrix (default: identity matrix).

## Methods

### Public methods:

- [GasComponents\\$new\(\)](#)
- [GasComponents\\$getFraction\(\)](#)
- [GasComponents\\$getUncertainty\(\)](#)
- [GasComponents\\$getCorrelation\(\)](#)
- [GasComponents\\$setFraction\(\)](#)
- [GasComponents\\$setUncertainty\(\)](#)
- [GasComponents\\$setCorrelation\(\)](#)
- [GasComponents\\$setFractionArray\(\)](#)
- [GasComponents\\$setUncertaintyArray\(\)](#)
- [GasComponents\\$setCorrelationMatrix\(\)](#)
- [GasComponents\\$clone\(\)](#)

**Method** `new()`: Create a new GasComponents object. Initialises fractions and uncertainties to zero and correlations to the 60x60 identity matrix.

*Usage:*

```
GasComponents$new()
```

**Method** `getFraction()`: Get the mole fraction of a single component.

*Usage:*

GasComponents\$getFraction(name)

*Arguments:*

name Component name (English) or integer index 1–60.

*Returns:* Numeric scalar.

**Method** getUncertainty(): Get the standard uncertainty of a single component.

*Usage:*

GasComponents\$getUncertainty(name)

*Arguments:*

name Component name (English) or integer index 1–60.

*Returns:* Numeric scalar.

**Method** getCorrelation(): Get the correlation between two components.

*Usage:*

GasComponents\$getCorrelation(name1, name2)

*Arguments:*

name1 First component name or index.

name2 Second component name or index.

*Returns:* Numeric scalar in  $[-1, 1]$ .

**Method** setFraction(): Set the mole fraction of a single component.

*Usage:*

GasComponents\$setFraction(name, value)

*Arguments:*

name Component name or index.

value Mole fraction [mol/mol].

**Method** setUncertainty(): Set the standard uncertainty of a single component.

*Usage:*

GasComponents\$setUncertainty(name, value)

*Arguments:*

name Component name or index.

value Standard uncertainty.

**Method** setCorrelation(): Set the correlation between two components.

*Usage:*

GasComponents\$setCorrelation(name1, name2, value)

*Arguments:*

name1 First component name or index.

name2 Second component name or index.

value Correlation coefficient in  $[-1, 1]$ .

**Method** setFractionArray(): Set all mole fractions at once.

*Usage:*

GasComponents\$setFractionArray(x)

*Arguments:*

x Numeric vector of length 60.

**Method** setUncertaintyArray(): Set all uncertainties at once.

*Usage:*

GasComponents\$setUncertaintyArray(u)

*Arguments:*

u Numeric vector of length 60.

**Method** setCorrelationMatrix(): Set the full correlation matrix.

*Usage:*

GasComponents\$setCorrelationMatrix(r)

*Arguments:*

r 60x60 numeric matrix; values must lie in  $[-1, 1]$ .

**Method** clone(): The objects of this class are cloneable with this method.

*Usage:*

GasComponents\$clone(deep = FALSE)

*Arguments:*

deep Whether to make a deep clone.

---

iso6976\_calc

*Calculate all ISO 6976:2016 properties and their uncertainties*

---

## Description

Calculate all ISO 6976:2016 properties and their uncertainties

## Usage

```
iso6976_calc(x, u_x, r_x, t1, t2, p2, k = 1)
```

## Arguments

x	Numeric vector of length 60: mole fractions [mol/mol]
u_x	Numeric vector of length 60: standard uncertainties
r_x	Numeric matrix 60x60: correlation matrix
t1	Combustion temperature [°C]: 0, 15, 15.55, 20, or 25
t2	Volume reference temperature [°C]: 0, 15, 15.55, or 20
p2	Reference pressure [kPa]: 90–110
k	Coverage factor (default 1)

**Value**

Named list of properties and uncertainties

# Index

`calculateProperties`, [3](#), [8](#)  
`componentIndex`, [4](#), [5](#)  
`componentName`, [5](#), [5](#)  
`componentNames`, [3](#), [4](#), [5](#), [5](#), [8](#)

`example1`, [6](#), [6](#), [7](#)  
`example2`, [6](#)  
`example3`, [7](#), [7](#)  
`example3_ex`, [7](#)

`GasComponents`, [4](#), [8](#)

`ISO6976.2016 (ISO6976.2016-package)`, [2](#)  
`ISO6976.2016-package`, [2](#)  
`iso6976_calc`, [10](#)