

Package ‘AdsorpR’

September 2, 2026

Type Package

Title Adsorption Isotherm Models

Version 0.1.1

Maintainer Jajati Mandal <J.Mandal2@salford.ac.uk>

Description Model adsorption behavior using classical isotherms, including Langmuir, Freundlich, Brunauer–Emmett–Teller (BET), and Temkin models. The package supports parameter estimation through both linearized and non-linear fitting techniques and generates high-quality plots for model diagnostics. It is intended for environmental scientists, chemists, and researchers working on adsorption phenomena in soils, water treatment, and material sciences. Functions are compatible with base 'R' and 'ggplot2' for visualization.

License GPL

Encoding UTF-8

Imports ggplot2, stats

Suggests testthat, knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Jajati Mandal [cre],
Sandipan Samanta [aut]

Config/roxygen2/version 8.1.0

Repository CRAN

Date/Publication 2026-09-02 11:10:14 UTC

Contents

bet_model	2
freundlich_model	3
langmuir_model	3
nonlinear_bet	4
nonlinear_freundlich	5
nonlinear_langmuir	6
nonlinear_temkin	6
temkin_model	7

Index	9
-------	---

bet_model	<i>BET Isotherm Model</i>
-----------	---------------------------

Description

The BET isotherm extends the Langmuir theory to multilayer adsorption (Brunauer et al., 1938). It is used to estimate surface area and porosity of adsorbents. The model is applicable under specific physical or chemical conditions and is given by: $Q = (Q_m * C_b * P) / ((P_0 - P)(1 + (C_b - 1) * P / P_0))$ <https://doi.org/10.1021/ja01269a023>

Usage

```
bet_model(Ce, Qe, Cs = max(Ce) * 1.1)
```

Arguments

- Ce Numeric vector of equilibrium concentrations.
- Qe Numeric vector of amount adsorbed.
- Cs Saturation concentration.

Value

A named list of BET parameters and model details.

See Also

Other linear: [freundlich_model\(\)](#), [langmuir_model\(\)](#), [temkin_model\(\)](#)

Examples

```
Ce <- c(1, 2, 3, 4, 5)
Qe <- c(0.8, 1.5, 2.1, 2.6, 2.9)
result <- bet_model(Ce, Qe)
print(result[1:2])
print(result$`Model Summary`)
print(result$Plot)
```

freundlich_model	<i>Freundlich Isotherm Model</i>
------------------	----------------------------------

Description

Freundlich isotherm describes adsorption on heterogeneous surfaces and assumes that the stronger binding sites are occupied first (Freundlich, 1907). It is represented by: $Q = K_f * C_e^{(1/n)}$ where K_f is the Freundlich constant related to adsorption capacity and n indicates adsorption intensity. <https://doi.org/10.1002/ange.19070201805>

Usage

```
freundlich_model(Ce, Qe)
```

Arguments

Ce	Numeric vector of equilibrium concentrations.
Qe	Numeric vector of amount adsorbed.

Value

A named list of Freundlich parameters and model details.

See Also

Other linear: [bet_model\(\)](#), [langmuir_model\(\)](#), [temkin_model\(\)](#)

Examples

```
Ce <- c(1, 2, 3, 4, 5)
Qe <- c(0.8, 1.5, 2.1, 2.6, 2.9)
result <- freundlich_model(Ce, Qe)
print(result[1:2])
print(result$`Model Summary`)
print(result$Plot)
```

langmuir_model	<i>Langmuir Isotherm Model</i>
----------------	--------------------------------

Description

Langmuir isotherm assumes monolayer adsorption onto a surface with a finite number of identical sites (Langmuir, 1918). It is characterized by uniform energies of adsorption onto the surface and no transmigration of adsorbate in the plane of the surface. The model is described by the equation: $Q = (Q_{max} * K_L * C_e) / (1 + K_L * C_e)$ where Q is the amount adsorbed, C_e is the equilibrium concentration, Q_{max} is the maximum adsorption capacity, and K_L is the Langmuir constant. <https://doi.org/10.1021/ja02242a004>

Usage

```
langmuir_model(Ce, Qe)
```

Arguments

Ce Numeric vector of equilibrium concentrations.
 Qe Numeric vector of amount adsorbed.

Value

A named list of Langmuir parameters and model details.

See Also

Other linear: [bet_model\(\)](#), [freundlich_model\(\)](#), [temkin_model\(\)](#)

Examples

```
Ce <- c(1, 2, 3, 4, 5)
Qe <- c(0.8, 1.5, 2.1, 2.6, 2.9)
result <- langmuir_model(Ce, Qe)
print(result[1:2])
print(result$`Model Summary`)
print(result$Plot)
```

nonlinear_bet

Non-linear BET Model

Description

Non-linear BET Model

Usage

```
nonlinear_bet(Ce, Qe, Cs = max(Ce) * 1.1)
```

Arguments

Ce Numeric vector of equilibrium concentrations.
 Qe Numeric vector of amount adsorbed.
 Cs Saturation concentration.

Value

A named list of BET parameters and model details.

See Also

Other nonlinear: [nonlinear_freundlich\(\)](#), [nonlinear_langmuir\(\)](#), [nonlinear_temkin\(\)](#)

Examples

```
Ce <- c(1, 2.5, 4, 5.5, 7)
Qe <- c(0.4, 1.0, 1.7, 2.3, 2.7)
result <- nonlinear_bet(Ce, Qe)
print(result$`BET Qm (mg/g)`)
print(result$`BET Cb`)
print(result$AIC)
print(result$`Pseudo R2`)
print(result$Plot)
```

nonlinear_freundlich *Non-linear Freundlich Model*

Description

Non-linear Freundlich Model

Usage

```
nonlinear_freundlich(Ce, Qe)
```

Arguments

Ce	Numeric vector of equilibrium concentrations.
Qe	Numeric vector of amount adsorbed.

Value

A named list of Freundlich parameters and model details.

See Also

Other nonlinear: [nonlinear_bet\(\)](#), [nonlinear_langmuir\(\)](#), [nonlinear_temkin\(\)](#)

Examples

```
Ce <- c(0.5, 1, 2, 4, 6, 8)
Qe <- c(0.3, 0.8, 1.6, 2.4, 2.9, 3.2)
result <- nonlinear_freundlich(Ce, Qe)
print(result$`Freundlich Kf`)
print(result$`Freundlich n`)
print(result$AIC)
print(result$`Pseudo R2`)
print(result$Plot)
```

nonlinear_langmuir	<i>Non-linear Langmuir Model</i>
--------------------	----------------------------------

Description

Fits the Langmuir isotherm model using non-linear least squares (nls).

Usage

```
nonlinear_langmuir(Ce, Qe)
```

Arguments

Ce	Numeric vector of equilibrium concentrations.
Qe	Numeric vector of amount adsorbed.

Value

A named list of Langmuir parameters and model details.

See Also

Other nonlinear: [nonlinear_bet\(\)](#), [nonlinear_freundlich\(\)](#), [nonlinear_temkin\(\)](#)

Examples

```
Ce <- c(1, 2, 4, 6, 8, 10)
Qe <- c(0.9, 1.6, 2.3, 2.7, 2.9, 3.0)
result <- nonlinear_langmuir(Ce, Qe)
print(result$`Langmuir Qmax (mg/g)`)
print(result$`Langmuir KL (L/mg)`)
print(result$AIC)
print(result$`Pseudo R2`)
print(result$Plot)
```

nonlinear_temkin	<i>Non-linear Temkin Model</i>
------------------	--------------------------------

Description

Non-linear Temkin Model

Usage

```
nonlinear_temkin(Ce, Qe, R = 8.314, T = 298)
```

Arguments

Ce	Numeric vector of equilibrium concentrations.
Qe	Numeric vector of amount adsorbed.
R	Universal gas constant.
T	Temperature in Kelvin.

Value

A named list of Temkin parameters and model details.

See Also

Other nonlinear: [nonlinear_bet\(\)](#), [nonlinear_freundlich\(\)](#), [nonlinear_langmuir\(\)](#)

Examples

```
Ce <- c(0.5, 1.5, 3, 4.5, 6)
Qe <- c(0.7, 1.3, 2.0, 2.4, 2.7)
result <- nonlinear_temkin(Ce, Qe)
print(result$`Temkin A`)
print(result$`Temkin B`)
print(result$AIC)
print(result$`Pseudo R2`)
print(result$Plot)
```

temkin_model

Temkin Isotherm Model

Description

The Temkin isotherm considers the effects of indirect adsorbate/adsorbate interactions. It assumes that the heat of adsorption of all molecules in the layer decreases linearly with coverage (Temkin and Pyzhev 1940). The model is given by: $Q = (RT / bT) * \ln(AT * Ce)$

Usage

```
temkin_model(Ce, Qe, R = 8.314, T = 298)
```

Arguments

Ce	Numeric vector of equilibrium concentrations.
Qe	Numeric vector of amount adsorbed.
R	Universal gas constant.
T	Temperature in Kelvin.

Value

A named list of Temkin parameters and model details.

See Also

Other linear: [bet_model\(\)](#), [freundlich_model\(\)](#), [langmuir_model\(\)](#)

Examples

```
Ce <- c(1, 2, 3, 4, 5)
Qe <- c(0.8, 1.5, 2.1, 2.6, 2.9)
result <- temkin_model(Ce, Qe)
print(result[1:2])
print(result$`Model Summary`)
print(result$Plot)
```

Index

* **linear**

- bet_model, 2
- freundlich_model, 3
- langmuir_model, 3
- temkin_model, 7

* **nonlinear**

- nonlinear_bet, 4
- nonlinear_friendly, 5
- nonlinear_langmuir, 6
- nonlinear_temkin, 6

bet_model, 2

bet_model(), 3, 4, 8

freundlich_model, 3

freundlich_model(), 2, 4, 8

langmuir_model, 3

langmuir_model(), 2, 3, 8

nonlinear_bet, 4

nonlinear_bet(), 5–7

nonlinear_friendly, 5

nonlinear_friendly(), 5–7

nonlinear_langmuir, 6

nonlinear_langmuir(), 5, 7

nonlinear_temkin, 6

nonlinear_temkin(), 5, 6

temkin_model, 7

temkin_model(), 2–4