

Package ‘qvirus’

May 21, 2025

Title Quantum Computing for Analyzing CD4 Lymphocytes and Antiretroviral Therapy

Version 0.0.4

Description Resources, tutorials, and code snippets dedicated to exploring the intersection of quantum computing and artificial intelligence (AI) in the context of analyzing Cluster of Differentiation 4 (CD4) lymphocytes and optimizing antiretroviral therapy (ART) for human immunodeficiency virus (HIV). With the emergence of quantum artificial intelligence and the development of small-scale quantum computers, there's an unprecedented opportunity to revolutionize the understanding of HIV dynamics and treatment strategies. This project leverages the R package 'qsimulatR' (Ostmeyer and Urbach, 2023, <<https://CRAN.R-project.org/package=qsimulatR>>), a quantum computer simulator, to explore these applications in quantum computing techniques, addressing the challenges in studying CD4 lymphocytes and enhancing ART efficacy.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

Depends R (>= 3.5)

LazyData true

Suggests rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

Imports dplyr, magrittr

NeedsCompilation no

Author Juan Pablo Acuña González [aut, cre, cph] (ORCID: <<https://orcid.org/0009-0003-6029-6560>>)

Maintainer Juan Pablo Acuña González <22253567@uagro.mx>

Repository CRAN

Date/Publication 2025-05-20 22:30:07 UTC

Contents

cds_diff 2

cd_3	3
cd_diff	4
estimate_payoffs	5
InteractionClassification	5
mse	6
mse.InteractionClassification	7
mse.payoffs	8
nearest_payoff	8
payoffs_list	9
phen_hiv	10
preds	11
preds2	11
qphen	12
summary	13
summary.InteractionClassification	14
summary.payoffs	15
vlogs_diff	15
vlog_diff	16
vl_3	16
vl_diff	18
Index	19

cds_diff	<i>Create Mean Standardized Differences from Longitudinal CD4 Data</i>
----------	--

Description

This function calculates the mean standardized differences of CD4 counts across time for each individual in the dataset.

Usage

```
cds_diff(cd_data)
```

Arguments

cd_data	A data frame of longitudinal CD4 count values per individual, where rows represent patients and columns represent sequential measurements across time (e.g., years or visits).
---------	--

Value

An object of class "Interaction" with the following components:

cds3_diff Mean standardized differences of raw CD4 count values.

Examples

```
data(cd_3)
cd_data <- cd_3[,-1]
result <- cds_diff(cd_data)
```

cd_3

Longitudinal CD4 Lymphocyte Counts for HIV Patients (2018-2024)

Description

Contains longitudinal measurements of CD4 lymphocyte counts for 176 patients living with HIV, recorded over the period from 2018 to 2024. CD4 counts are a critical indicator of immune function, used to monitor the progression of HIV and the effectiveness of treatments. Measurements were taken at various points throughout the study, with some missing values due to unavailable data for specific patients at certain times.

Usage

```
cd_3
```

Format

A data frame with 176 rows and 18 variables:

ID Unique identifier for each patient.

cd_2018_1 CD4 count for the first measurement in 2018.

cd_2018_2 CD4 count for the second measurement in 2018.

cd_2019_1 CD4 count for the first measurement in 2019.

cd_2019_2 CD4 count for the second measurement in 2019.

cd_2020_1 CD4 count for the first measurement in 2020.

cd_2021_1 CD4 count for the first measurement in 2021.

cd_2021_2 CD4 count for the second measurement in 2021.

cd_2021_3 CD4 count for the third measurement in 2021.

cd_2022_1 CD4 count for the first measurement in 2022.

cd_2022_2 CD4 count for the second measurement in 2022.

cd_2022_3 CD4 count for the third measurement in 2022.

cd_2023_1 CD4 count for the first measurement in 2023.

cd_2023_2 CD4 count for the second measurement in 2023.

cd_2023_3 CD4 count for the third measurement in 2023.

cd_2024_1 CD4 count for the first measurement in 2024.

cd_2024_2 CD4 count for the second measurement in 2024.

cd_2024_3 CD4 count for the third measurement in 2024.

Details

. CD4 counts are used to monitor immune system health in individuals with HIV. A lower CD4 count often indicates a weakened immune system, whereas higher counts suggest a stronger immune response. Some values are missing, indicating no measurement was taken for a particular patient at that time.

Source

Clinical data from Hospital Vicente Guerrero, IMSS, HIV Clinic.

Examples

```
# Load the dataset
data(cd_3)

# Summarize CD4 counts for the year 2021
summary(cd_3[, c("cd_2021_1", "cd_2021_2", "cd_2021_3")])
```

cd_diff

Create Mean Differences from Longitudinal CD4 Data

Description

This function calculates the mean differences of CD4 counts across time for each individual in the dataset.

Usage

```
cd_diff(cd_data)
```

Arguments

cd_data	A data frame of longitudinal CD4 count values per individual, where rows represent patients and columns represent sequential measurements across time (e.g., years or visits).
---------	--

Value

An object of class "Interaction" with the following components:

cd3_diff Mean differences of raw CD4 count values.

Examples

```
data(cd_3)
cd_data <- cd_3[,-1]
result <- cd_diff(cd_data)
```

estimate_payoffs	<i>Estimate Payoff Parameters for HIV Phenotype Interactions</i>
------------------	--

Description

This function estimates the payoff parameters for HIV phenotype interactions based on the provided classification object and predictions from a viral load model. It calculates the mean differences in viral loads and CD4 counts, as well as the average payoffs for each classification.

Usage

```
estimate_payoffs(object, predictions)
```

Arguments

object	An object of class <code>InteractionClassification</code> containing the data on viral load differences and CD4 counts.
predictions	A <code>data.frame</code> containing predictions of viral loads or CD4 values.

Examples

```
set.seed(42)
data(cd_3)
cd_data <- cd_3[, -1]
cd_result <- cds_diff(cd_data)
data(vl_3)
vl_data <- vl_3[, -1]
vl_result <- vlogs_diff(vl_data)
result <- InteractionClassification(cd_result = cd_result, vl_result = vl_result)
data(preds)
payoffs_results <- estimate_payoffs(result, preds)
```

InteractionClassification	<i>Classify HIV phenotype interactions using k-means clustering</i>
---------------------------	---

Description

This function performs k-means clustering on the differences in viral load and CD4 counts to classify interaction types between HIV phenotypes. It returns an object of class `InteractionClassification`, a `data.frame` with classification labels.

Usage

```
InteractionClassification(cd_result, vl_result, k = 4, ns = 100, seed = 123)
```

Arguments

<code>cd_result</code>	A numeric vector of differences in CD4 T-cell counts.
<code>vl_result</code>	A numeric vector of differences in log viral load.
<code>k</code>	Integer. The number of clusters to use in k-means. Default is 4.
<code>ns</code>	Integer. Number of random initializations for the k-means algorithm. Default is 100.
<code>seed</code>	Integer. Seed for random number generation to ensure reproducibility. Default is 123.

Value

A data.frame of class `InteractionClassification` with three columns:

`cds3_result` The CD4 count difference for each interaction.

`vlogs3_result` The viral load difference (log scale) for each interaction.

`classification` An integer label (1 to k) indicating the interaction cluster.

Examples

```
set.seed(42)
data(cd_3)
cd_data <- cd_3[, -1]
cd_result <- cds_diff(cd_data)
data(vl_3)
vl_data <- vl_3[, -1]
vl_result <- vlogs_diff(vl_data)
result <- InteractionClassification(cd_result = cd_result, vl_result = vl_result)
```

mse

Mean Squared Errors for Interaction Classification

Description

Mean squared errors (MSE) for viral load differences and CD4 count differences by comparing the actual values with the group means from the classification.

Computes the mean squared error (MSE) between observed CD4 and viral load differences and their corresponding predicted payoff values within each interaction classification.

Usage

```
mse(object, ...)
```

```
mse(object, ...)
```

Arguments

object An object of class `payoffs`.
... Additional arguments passed to other methods (currently not used).

Value

A data.frame containing the MSE for CD4 count differences (`mse_cds_diff`) and (`mse_vlogs_diff`) for viral load differences.

Examples

```
set.seed(42)
data(cd_3)
cd_data <- cd_3[,-1]
cd_result <- cds_diff(cd_data)
data(vl_3)
vl_data <- vl_3[,-1]
vl_result <- vlogs_diff(vl_data)
result <- InteractionClassification(cd_result = cd_result, vl_result = vl_result)
mse(result)

set.seed(42)
data(cd_3)
cd_data <- cd_3[,-1]
cd_result <- cds_diff(cd_data)
data(vl_3)
vl_data <- vl_3[,-1]
vl_result <- vlogs_diff(vl_data)
result <- InteractionClassification(cd_result = cd_result, vl_result = vl_result)
data(preds)
payoffs_results <- estimate_payoffs(result, preds)
mse(payoffs_results)
```

`mse.InteractionClassification`*Mean Squared Errors for Interaction Classification*

Description

Mean squared errors (MSE) for viral load differences and CD4 count differences by comparing the actual values with the group means from the classification.

Usage

```
## S3 method for class 'InteractionClassification'
mse(object, ...)
```

Arguments

object	An object of class InteractionClassification containing the classified data and clustering results.
...	Additional arguments passed to other methods (currently not used).

mse.payoffs	<i>Mean Squared Errors for Payoff Predictions</i>
-------------	---

Description

Computes the mean squared error (MSE) between observed CD4 and viral load differences and their corresponding predicted payoff values within each interaction classification.

Usage

```
## S3 method for class 'payoffs'
mse(object, ...)
```

Arguments

object	An object of class payoffs.
...	Additional arguments passed to other methods (currently not used).

nearest_payoff	<i>Find Nearest Payoff</i>
----------------	----------------------------

Description

This function computes the nearest simulated payoff from a given list of payoffs based on a viral load difference (vl_diff). It returns both the nearest payoff value and its corresponding payoff name.

Usage

```
nearest_payoff(vl_diff, payoffs_list)
```

Arguments

vl_diff	Numeric value representing the viral load difference for which the nearest payoff will be found.
payoffs_list	A named list of payoff values, where the names correspond to specific payoffs and the values are the associated payoff values.

Examples

```
I <- diag(2)
H <- 1 / sqrt(2) * matrix(c(1, 1, 1, -1), 2, 2)
Z <- diag(c(1, -1))
gates <- list(I = I, H = H, Z = Z)
alpha <- 0.3; beta <- 0.1; gamma <- 0.5; theta <- 0.2
alpha2 <- 0.35; beta2 <- 0.15; gamma2 <- 0.6; theta2 <- 0.25
pays <- payoffs_list(gates, alpha, beta, gamma, theta, alpha2, beta2, gamma2, theta2)
nearest_payoff(-0.2, pays)
```

payoffs_list

Compute Payoff Values for Quantum HIV Phenotype Interactions

Description

Computes payoff values for all pairwise combinations of quantum gate strategies provided in a named list. For each pair, the function calculates the payoffs for both phenotypes v and V using two different sets of payoff parameters.

Usage

```
payoffs_list(gates, alpha, beta, gamma, theta, alpha2, beta2, gamma2, theta2)
```

Arguments

gates	A named list of 2x2 unitary matrices representing quantum strategies (e.g., I, H, Z).
alpha	Numeric scalar, payoff coefficient for phenotype v when both play 0.
beta	Numeric scalar, payoff coefficient for phenotype v when v plays 0, V plays 1.
gamma	Numeric scalar, payoff coefficient for phenotype v when v plays 1, V plays 0.
theta	Numeric scalar, payoff coefficient for phenotype v and V when both play 1.
alpha2	Numeric scalar, alternate value of alpha for phenotype v in a second scenario.
beta2	Numeric scalar, alternate value of beta for phenotype v in a second scenario.
gamma2	Numeric scalar, alternate value of gamma for phenotype v in a second scenario.
theta2	Numeric scalar, alternate value of theta for phenotype v in a second scenario.

Examples

```
I <- diag(2)
H <- 1 / sqrt(2) * matrix(c(1, 1, 1, -1), 2, 2)
Z <- diag(c(1, -1))
gates <- list(I = I, H = H, Z = Z)
payoffs <- payoffs_list(gates, 1, 0.5, 0.3, 0.2, 1.5, 0.6, 0.7, 0.8)
```

phen_hiv

Calculate Final State and Payoffs in Quantum Game

Description

This function calculates the final quantum state and expected payoffs for two players in a quantum game based on their strategies. The function uses quantum gates and unitary transformations to simulate the game dynamics.

Usage

```
phen_hiv(strategy1, strategy2, alpha, beta, gamma, theta)
```

Arguments

strategy1	A 2x2 matrix representing the strategy of player 1.
strategy2	A 2x2 matrix representing the strategy of player 2.
alpha	A numeric value representing the payoff for outcome 00>.
beta	A numeric value representing the payoff for outcome 01>.
gamma	A numeric value representing the payoff for outcome 10>.
theta	A numeric value representing the payoff for outcome 11>.

References

Özlür Başıer, B. (2022). "Analyzing the competition of HIV-1 phenotypes with quantum game theory". Gazi University Journal of Science, 35(3), 1190–1198. [doi:10.35378/gujs.772616](https://doi.org/10.35378/gujs.772616)

Examples

```
strategy1 <- diag(2) # Identity matrix for strategy 1
strategy2 <- diag(2) # Identity matrix for strategy 2
alpha <- 1
beta <- 0.5
gamma <- 2
theta <- 0.1
result <- phen_hiv(strategy1, strategy2, alpha, beta, gamma, theta)
```

preds	<i>Predictions for Longitudinal Viral Load Values for HIV Patients (2018-2024)</i>
-------	--

Description

Contains predictions of longitudinal viral load values for 176 patients from 2018 to 2024.

Usage

preds

Format

An object of class spec_tbl_df (inherits from tbl_df, tbl, data.frame) with 176 rows and 1 columns.

Source

Clinical data from Hospital Vicente Guerrero, IMSS, HIV Clinic.

Examples

data(preds)
head(preds)

preds2	<i>Batched Predictions for Longitudinal Viral Load Values for HIV Patients (2018-2024)</i>
--------	--

Description

Contains batched predictions of longitudinal viral load values for 176 patients from 2018 to 2024.

Usage

preds2

Format

An object of class spec_tbl_df (inherits from tbl_df, tbl, data.frame) with 176 rows and 1 columns.

Source

Clinical data from Hospital Vicente Guerrero, IMSS, HIV Clinic.

Examples

```
data(preds2)
head(preds2)
```

qphen

Quantum Phenotype Interactions in HIV Model

Description

The qphen dataset contains 176 observations and 24 variables, representing classified phenotype interactions in a quantum game-theoretic model of HIV phenotypes. The data includes CD4 and viral load differences, quantum game strategies, classification clusters, and tuberculosis/genoresistance indicators.

Usage

```
data(qphen)
```

Format

A data frame with 176 rows and 24 variables:

id (double) Unique identifier for each observation.

vl_diff (double) Difference in viral load (log scale) between time points.

cd_diff (double) Difference in CD4 count between time points.

vlogs_diff_mean (double) Mean difference of viral loads across the dataset.

cds_diff_mean (double) Mean difference of CD4 counts across the dataset.

n (double) Number of cases in each interaction cluster.

payoffs (double) Computed payoff for the phenotype interaction. #'

payoffs_b (double) Alternative computed payoff.

nearest_payoff (double) Closest estimated payoff value.

classification_2 (double) Cluster assignment for phenotype interactions (second clustering method).

classification_3 (double) Cluster assignment for phenotype interactions (third clustering method).

classification_4 (double) Cluster assignment for phenotype interactions (fourth clustering method).

phen_1 (double) Phenotype type (v or V).

str1_2 (double) Strategy of the first phenotype using X, T, or H gate (binary encoding).

str1_3 (double) Alternative strategy of the first phenotype.

str2_2 (double) Strategy of the second phenotype using H, Id, S, T, X, Y, or Z gate (binary encoding).

str2_3 (double) Alternative strategy of the second phenotype.

str2_4 (double) Alternative strategy of the second phenotype.

str2_5 (double) Alternative strategy of the second phenotype.

str2_6 (double) Alternative strategy of the second phenotype.
str2_7 (double) Alternative strategy of the second phenotype.
batch_1 (double) Indicates whether predictions were made on full data or batch data.
TB_1 (double) Indicator for tuberculosis presence (1 = TB, 0 = no TB).
GR_1 (double) Indicator for genoresistance presence (1 = resistant, 0 = non-resistant).

Examples

```
data(qphen)
head(qphen)
```

summary	<i>Summarize an InteractionClassification object</i>
---------	--

Description

Computes summary statistics by classification group from an object of class `InteractionClassification`, including mean differences in viral load and CD4 counts, and the number of observations per cluster.

This function summarizes the payoffs object by classification.

Usage

```
summary(object, ...)

summary(object, ...)
```

Arguments

<code>object</code>	A payoffs object.
<code>...</code>	Additional arguments (not used).

Value

A data.frame with one row per interaction cluster and the following columns:

classification Cluster label (as factor).
cds_diff_mean Mean of CD4 differences in the group.
vlogs_diff_mean Mean of viral load differences in the group.
n Number of observations in the group.

Examples

```

set.seed(42)
data(cd_3)
cd_data <- cd_3[,-1]
cd_result <- cds_diff(cd_data)
data(vl_3)
vl_data <- vl_3[,-1]
vl_result <- vlogs_diff(vl_data)
result <- InteractionClassification(cd_result = cd_result, vl_result = vl_result)
summary(result)
set.seed(42)
data(cd_3)
cd_data <- cd_3[,-1]
cd_result <- cds_diff(cd_data)
data(vl_3)
vl_data <- vl_3[,-1]
vl_result <- vlogs_diff(vl_data)
result <- InteractionClassification(cd_result = cd_result, vl_result = vl_result)
data(preds)
payoffs_results <- estimate_payoffs(result, preds)
summary(payoffs_results)

```

summary.InteractionClassification

Summarize an InteractionClassification object

Description

Computes summary statistics by classification group from an object of class `InteractionClassification`, including mean differences in viral load and CD4 counts, and the number of observations per cluster.

Usage

```

## S3 method for class 'InteractionClassification'
summary(object, ...)

```

Arguments

object	An object of class <code>InteractionClassification</code> returned by the InteractionClassification() function.
...	Additional arguments passed to other methods (currently not used).

summary.payoffs	<i>Summarize Payoffs</i>
-----------------	--------------------------

Description

This function summarizes the `payoffs` object by classification.

Usage

```
## S3 method for class 'payoffs'  
summary(object, ...)
```

Arguments

<code>object</code>	A <code>payoffs</code> object.
<code>...</code>	Additional arguments (not used).

<code>vlogs_diff</code>	<i>Create Mean Standardized Differences from Logarithmic Viral Load Data</i>
-------------------------	--

Description

This function calculates the mean standardized differences of logarithmic viral loads across time for each individual in the dataset.

Usage

```
vlogs_diff(vl_data)
```

Arguments

<code>vl_data</code>	A data frame of longitudinal viral load values per individual, where rows represent patients and columns represent sequential measurements across time (e.g., years or visits).
----------------------	---

Value

An object of class "Interaction" with the following components:

vlogs3_diff Mean standardized differences of logarithmic viral load values.

Examples

```
data(vl_3)  
vl_data <- vl_3[,-1]  
result <- vlogs_diff(vl_data)
```

vlog_diff	Create Mean Differences from Logarithmic Viral Load Data
-----------	--

Description

This function calculates the mean differences of lograithmic viral loads across time for each individual in the dataset.

Usage

```
vlog_diff(vl_data)
```

Arguments

vl_data A data frame of longitudinal viral load values per individual, where rows represent patients and columns represent sequential measurements across time (e.g., years or visits).

Value

An object of class "Interaction" with the following components:

vlog3_diff Mean differences of logarithmic viral load values.

Examples

```
data(vl_3)
vl_data <- vl_3[,-1]
result <- vlog_diff(vl_data)
```

vl_3	Longitudinal Viral Load Values for HIV Patients (2018-2024)
------	---

Description

Contains longitudinal measurements of viral load for 176 patients from 2018 to 2024. Viral load is a critical marker used to monitor the effectiveness of HIV treatment by measuring the amount of HIV RNA in the blood.

Usage

```
vl_3
```


Format

A data frame with 176 rows and 18 variables:

ID Unique identifier for each patient.

vl_2018_1 Viral load for the first measurement in 2018.

vl_2018_2 Viral load for the second measurement in 2018.

vl_2019_1 Viral load for the first measurement in 2019.

vl_2019_2 Viral load for the second measurement in 2019.

vl_2020_1 Viral load for the first measurement in 2020.

vl_2021_1 Viral load for the first measurement in 2021.

vl_2021_2 Viral load for the second measurement in 2021.

vl_2021_3 Viral load for the third measurement in 2021.

vl_2022_1 Viral load for the first measurement in 2022.

vl_2022_2 Viral load for the second measurement in 2022.

vl_2022_3 Viral load for the third measurement in 2022.

vl_2023_1 Viral load for the first measurement in 2023.

vl_2023_2 Viral load for the second measurement in 2023.

vl_2023_3 Viral load for the third measurement in 2023.

vl_2024_1 Viral load for the first measurement in 2024.

vl_2024_2 Viral load for the second measurement in 2024.

vl_2024_3 Viral load for the third measurement in 2024.

Details

The viral load measurements provide insight into the patient's response to antiretroviral therapy (ART). Lower viral load values, especially undetectable levels, indicate better control of the infection. Missing values indicate that no viral load measurement was available for that patient at that specific time.

Source

Clinical data from Hospital Vicente Guerrero, IMSS, HIV Clinic.

Examples

```
## Not run:
# Load the dataset
data(vl_3)

# Summarize viral loads for the year 2021
summary(vl_3[, c("cd_2021_1", "cd_2021_2", "cd_2021_3")])

## End(Not run)
```

`vl_diff`*Create Mean Differences from Longitudinal Viral Load Data*

Description

This function calculates the mean differences of viral loads across time for each individual in the dataset.

Usage

```
vl_diff(vl_data)
```

Arguments

<code>vl_data</code>	A data frame of longitudinal viral load values per individual, where rows represent patients and columns represent sequential measurements across time (e.g., years or visits).
----------------------	---

Value

An object of class "Interaction" with the following components:

vl3_diff Mean differences of raw viral load values.

Examples

```
data(vl_3)
vl_data <- vl_3[, -1]
result <- vl_diff(vl_data)
```

Index

* datasets

- cd_3, [3](#)
- preds, [11](#)
- preds2, [11](#)
- qphen, [12](#)
- v1_3, [16](#)

- cd_3, [3](#)
- cd_diff, [4](#)
- cds_diff, [2](#)

- estimate_payoffs, [5](#)

- InteractionClassification, [5](#)
- InteractionClassification(), [14](#)

- mse, [6](#)
- mse.InteractionClassification, [7](#)
- mse.payoffs, [8](#)

- nearest_payoff, [8](#)

- payoffs_list, [9](#)
- phen_hiv, [10](#)
- preds, [11](#)
- preds2, [11](#)

- qphen, [12](#)

- summary, [13](#)
- summary.InteractionClassification, [14](#)
- summary.payoffs, [15](#)

- v1_3, [16](#)
- v1_diff, [18](#)
- vlog_diff, [16](#)
- vlogs_diff, [15](#)