

# Package ‘demic’

January 22, 2024

**Title** Dynamic Estimator of Microbial Communities

**Description** Multi-sample algorithm based on contigs and coverage values, to infer the relative distances of contigs from the replication origin and to accurately compare bacterial growth rates between samples. Yuan Gao and Hongzhe Li (2018) <[doi:10.1038/s41592-018-0182-0](https://doi.org/10.1038/s41592-018-0182-0)>.

**Version** 2.0.0

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Imports** lme4, Matrix (>= 1.6-2), reshape2

**Suggests** covr, lintr, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Depends** R (>= 2.10)

**LazyData** true

**License** GPL (>= 3)

**URL** <https://github.com/Ulthran/DEMIC>

**BugReports** <https://github.com/Ulthran/DEMIC/issues>

**NeedsCompilation** no

**Author** Yuan Gao [aut, cph],  
Charlie Bushman [cre]

**Maintainer** Charlie Bushman <[ctbushman@gmail.com](mailto:ctbushman@gmail.com)>

**Repository** CRAN

**Date/Publication** 2024-01-22 22:22:47 UTC

## R topics documented:

compare_contig_subsets . . . . .	2
compare_sample_subsets . . . . .	3
consist_transfer . . . . .	4
ContigCluster1 . . . . .	5

ContigCluster2 . . . . .	5
contig_pca . . . . .	6
cor_diff . . . . .	6
df_transfer . . . . .	7
est_ptr . . . . .	7
est_ptrs_subset . . . . .	8
est_ptr_on . . . . .	9
est_ptr_on_all . . . . .	10
filter_sample . . . . .	11
get_epr_stats . . . . .	11
iterate_pipelines . . . . .	12
ks . . . . .	13
lme4_model . . . . .	13
lm_column . . . . .	14
max_bin_001 . . . . .	14
max_bin_002 . . . . .	15
max_bin_003 . . . . .	15
pipeline . . . . .	16
reshape_filtered . . . . .	16
select_by_ks_test . . . . .	17
test_reasonable . . . . .	17
verify_input . . . . .	18

<b>Index</b>	<b>19</b>
--------------	-----------

---

compare\_contig\_subsets

*Compares contig subset x against contig subset y*

---

## Description

Compares contig subset x against contig subset y

## Usage

```
compare_contig_subsets(
  est_ptrs_x,
  est_ptrs_y,
  pipeline_x,
  pipeline_y,
  cor_cutoff,
  max_cor
)
```

**Arguments**

est_ptrs_x	PTR estimates from contig subset x
est_ptrs_y	PTR estimates from contig subset y
pipeline_x	pipeline for contig subset x
pipeline_y	pipeline for contig subset y
cor_cutoff	the correlation cutoff
max_cor	the max correlation

**Value**

a named list including the est\_ptr dataframe and a max\_cor value

- sample: sample
- est\_ptr: PTR estimate
- coefficient: coefficient of linear regression
- pValue: p-value of linear regression
- cor: correlation coefficient
- correctY: corrected coverage

max\_cor: the max correlation achieved

---

compare\_sample\_subsets

*Compares sample subset x against sample subset y*

---

**Description**

Compares sample subset x against sample subset y

**Usage**

```
compare_sample_subsets(  
  est_ptrs_x,  
  est_ptrs_y,  
  pipeline_x,  
  pipeline_y,  
  cor_cutoff,  
  max_cor  
)
```

**Arguments**

est_ptr_x	PTR estimates from sample subset x
est_ptr_y	PTR estimates from sample subset y
pipeline_x	pipeline for sample subset x
pipeline_y	pipeline for sample subset y
cor_cutoff	the correlation cutoff
max_cor	the max correlation

**Value**

a named list including the est\_ptr dataframe and a max\_cor value

- sample: sample
- est\_ptr: PTR estimate
- coefficient: coefficient of linear regression
- pValue: p-value of linear regression
- cor: correlation coefficient
- correctY: corrected coverage

---

consist_transfer	<i>A function for data frame integration</i>
------------------	--

---

**Description**

A function for data frame integration

**Usage**

```
consist_transfer(x, y, i)
```

**Arguments**

x	first data frame
y	second data frame
i	'sample' column

**Value**

a data frame with the other column as mean or max of that in the original two

---

ContigCluster1	<i>Contig Cluster 1</i>
----------------	-------------------------

---

**Description**

Data associated with DEMIC paper (on SourceForge)

**Usage**

ContigCluster1

**Format**

ContigCluster1:

A data frame with 120,897 rows and 5 columns:

**log\_cov** Log Coverage for Sliding Windows over Contigs

**GC\_content** GC Content for Sliding Windows over Contigs

**sample** Sample Name

**contig** Contig Name

**length** Length of Contig

**Source**

<https://sourceforge.net/projects/demic/files/>

---

ContigCluster2	<i>Contig Cluster 2</i>
----------------	-------------------------

---

**Description**

Data associated with DEMIC paper (on SourceForge)

**Usage**

ContigCluster2

**Format**

ContigCluster2:

A data frame with 66,735 rows and 5 columns:

**log\_cov** Log Coverage for Sliding Windows over Contigs

**GC\_content** GC Content for Sliding Windows over Contigs

**sample** Sample Name

**contig** Contig Name

**length** Length of Contig

**Source**

<https://sourceforge.net/projects/demic/files/>

---

contig\_pca

*A function to return the first dimension of PCA on an input matrix*

---

**Description**

A function to return the first dimension of PCA on an input matrix

**Usage**

```
contig_pca(X)
```

**Arguments**

X                    a matrix to undergo PCA

**Value**

first dimension of the PCA results

---

cor\_diff

*Determine the majority orientation of the input PTR estimates correlations*

---

**Description**

Determine the majority orientation of the input PTR estimates correlations

**Usage**

```
cor_diff(Z)
```

**Arguments**

Z                    a vector of values

**Value**

a minor subset, where each value has the same orientation

---

df_transfer	<i>A function for data frame transfer</i>
-------------	---

---

**Description**

A function for data frame transfer

**Usage**

```
df_transfer(x, y)
```

**Arguments**

x	first data frame with six columns
y	second data frame with six columns

**Value**

a data frame with the same six columns but integrated info

---

est_ptr	<i>Estimate PTRs using all input data as well as using subsets of contigs and samples</i>
---------	---

---

**Description**

Estimate PTRs using all input data as well as using subsets of contigs and samples

**Usage**

```
est_ptr(X)
```

**Arguments**

X	dataframe with coverage matrix (column names: "log_cov", "GC_content", "sample", "contig", "length")
---	--

**Value**

named list with results from all three methods all\_ptr dataframe with the estimated PTRs on success, null otherwise

- est\_ptr: estimated PTR values
- coefficient: coefficient of linear regression
- pValue: p-value of linear regression

- cor: correlation coefficient
- correctY: corrected coverage

contigs\_ptr dataframe with the estimated PTRs on success, null otherwise

- est\_ptr: estimated PTR values
- coefficient: coefficient of linear regression
- pValue: p-value of linear regression
- cor: correlation coefficient
- correctY: corrected coverage

samples\_ptr dataframe with the estimated PTRs on success, null otherwise

- est\_ptr: estimated PTR values
- coefficient: coefficient of linear regression
- pValue: p-value of linear regression
- cor: correlation coefficient
- correctY: corrected coverage

### Examples

```
est_ptr_subset_001 <- est_ptr(max_bin_003)
est_ptr_subset_001
```

---

est\_ptr\_subset      *Get PTR estimates for output of the core pipeline on a subset of data*

---

### Description

Get PTR estimates for output of the core pipeline on a subset of data

### Usage

```
est_ptr_subset(p)
```

### Arguments

p                      is the pipeline named list



**Value**

a dataframe

- sample: sample
- est\_ptr: PTR estimate
- coefficient: coefficient of linear regression
- pValue: p-value of linear regression
- cor: correlation coefficient
- correctY: corrected coverage

---

est_ptr_on	<i>Tries up to max_attempts times to compare each permutation of removing random subsets of contigs/samples from X, and returns the PTR estimate if a valid one comes back from the comparisons</i>
------------	---

---

**Description**

Requires a minimum of  $2 * \text{num\_subsets}$  contigs/samples

**Usage**

```
est_ptr_on(X, subset_on, max_attempts = 10, num_subsets = 3, cor_cutoff = 0.98)
```

**Arguments**

X	cov3 dataframe
subset_on	either "contig" or "sample"
max_attempts	max number of attempts to find a valid ptr estimate
num_subsets	number of subsets to split contigs/samples into
cor_cutoff	minimum correlation coefficient to accept PTR estimate

**Value**

est\_ptrs dataframe on success, null otherwise

- est\_ptr: estimated PTR values
- coefficient: coefficient of linear regression
- pValue: p-value of linear regression
- cor: correlation coefficient
- correctY: corrected coverage

**Examples**

```
est_ptrs_001_on_contigs <- est_ptr_on(max_bin_003, "contig", num_subsets = 5)
est_ptrs_001_on_contigs

est_ptrs_001_on_samples <- est_ptr_on(max_bin_003, "sample")
is.null(est_ptrs_001_on_samples)
```

---

est_ptr_on_all	<i>Estimates PTRs based on the whole input dataset</i>
----------------	--

---

**Description**

Estimates PTRs based on the whole input dataset

**Usage**

```
est_ptr_on_all(X)
```

**Arguments**

X cov3 dataframe

**Value**

est\_ptrs dataframe on success, null otherwise

- est\_ptr: estimated PTR values
- coefficient: coefficient of linear regression
- pValue: p-value of linear regression
- cor: correlation coefficient
- correctY: corrected coverage

**Examples**

```
est_ptrs_001 <- est_ptr_on_all(max_bin_003)
est_ptrs_001
```

---

filter_sample	<i>A function for sample filtration Input requirements: 1. have values in more than half of the contigs 2. average <math>\log_2(\text{cov}) &gt; 0</math> in all these contigs</i>
---------------	--

---

**Description**

A function for sample filtration Input requirements: 1. have values in more than half of the contigs  
2. average  $\log_2(\text{cov}) > 0$  in all these contigs

**Usage**

```
filter_sample(Z, avg_cutoff, cutoff_ratio)
```

**Arguments**

Z	a matrix
avg_cutoff	threshold of average
cutoff_ratio	threshold of ratio

**Value**

the coefficient and p value of linear regression

---

get_eptr_stats	<i>Generate a variety of stats on PTR estimates for a given dataset</i>
----------------	---

---

**Description**

Generate a variety of stats on PTR estimates for a given dataset

**Usage**

```
get_eptr_stats(X, iterations = 30)
```

**Arguments**

X	cov3 dataframe
iterations	number of iterations to run

**Value**

named list of stats on PTR estimates

- all\_sd: standard deviation of PTR estimates from all method
- all\_mean: mean of PTR estimates from all method
- contigs\_sd: standard deviation of PTR estimates from contigs method
- contigs\_mean: mean of PTR estimates from contigs method
- samples\_sd: standard deviation of PTR estimates from samples method
- samples\_mean: mean of PTR estimates from samples method

**Examples**

```
stats <- get_eptr_stats(max_bin_001[max_bin_001$sample %in% c('Akk0_001', 'Akk1_001')], [], 2)
stats
```

---

iterate\_pipelines      *A function for iteration of pipeline until convergence*

---

**Description**

A function for iteration of pipeline until convergence

**Usage**

```
iterate_pipelines(Z)
```

**Arguments**

Z                      a matrix of coverages

**Value**

a named list

- samples: vector of final filtered samples
- correct\_ys: matrix of sample, contig and corrected coverages
- pc1: matrix of contig and PC1 values
- pc1\_range: vector of PC1 range
- samples\_y: samples filtered for reliable coverage

---

ks	<i>A convenient function for KS test of uniform distribution</i>
----	--

---

**Description**

A convenient function for KS test of uniform distribution

**Usage**

ks(x)

**Arguments**

x                    a vector without NA

**Value**

the p value of KS test

---

lme4_model	<i>Run mixed linear model with random effect using lme4</i>
------------	---

---

**Description**

Run mixed linear model with random effect using lme4

**Usage**

lme4\_model(X)

**Arguments**

X                    input data frame

**Value**

a dataframe

---

lm_column	<i>A convenient function for ordinary linear regression on two vectors</i>
-----------	--

---

**Description**

A convenient function for ordinary linear regression on two vectors

**Usage**

```
lm_column(x, y)
```

**Arguments**

x	first vector
y	second vector

**Value**

the coefficient and p value of linear regression

---

max_bin_001	<i>MaxBin2 Cluster 001</i>
-------------	----------------------------

---

**Description**

Generated by PyCov3 on simulated test data

**Usage**

```
max_bin_001
```

**Format**

max\_bin\_001:

A data frame with 79,740 rows and 5 columns:

**log\_cov** Log Coverage for Sliding Windows over Contigs

**GC\_content** GC Content for Sliding Windows over Contigs

**sample** Sample Name

**contig** Contig Name

**length** Length of Contig

**Source**

<https://sourceforge.net/projects/demic/files/>

---

max_bin_002	<i>MaxBin2 Cluster 002</i>
-------------	----------------------------

---

**Description**

Generated by PyCov3 on simulated test data

**Usage**

max\_bin\_002

**Format**

max\_bin\_002:

A data frame with 148,638 rows and 5 columns:

**log\_cov** Log Coverage for Sliding Windows over Contigs

**GC\_content** GC Content for Sliding Windows over Contigs

**sample** Sample Name

**contig** Contig Name

**length** Length of Contig

**Source**

<https://sourceforge.net/projects/demic/files/>

---

max_bin_003	<i>MaxBin2 Cluster 003</i>
-------------	----------------------------

---

**Description**

Generated by PyCov3 on simulated test data

**Usage**

max\_bin\_003

**Format**

max\_bin\_003:

A data frame with 124,578 rows and 5 columns:

**log\_cov** Log Coverage for Sliding Windows over Contigs

**GC\_content** GC Content for Sliding Windows over Contigs

**sample** Sample Name

**contig** Contig Name

**length** Length of Contig

**Source**

<https://sourceforge.net/projects/demic/files/>

---

pipeline	<i>A function representing the pipeline of four steps including GC bias correction, sample filtration, PCA and contig filtration</i>
----------	--

---

**Description**

A function representing the pipeline of four steps including GC bias correction, sample filtration, PCA and contig filtration

**Usage**

```
pipeline(Y, i)
```

**Arguments**

Y	a matrix of coverages
i	cutoff of filtering samples changes according to parameter i; i=1, cutoffRatio is 0.5; i=2, cutoffRatio is 1 as contig is clean

**Value**

a named list

- samples: final list of filtered samples
- correct\_ys: dataframe with correct Y values per contig/sample
- pc1: PC1 results of PCA per contig
- pc1\_range: range of PC1
- samples\_y: samples filtered for reliable coverage

---

reshape_filtered	<i>A function for reshape to facilitate PCA, removing all contigs with missing values for designated samples</i>
------------------	--

---

**Description**

A function for reshape to facilitate PCA, removing all contigs with missing values for designated samples

**Usage**

```
reshape_filtered(samples_filtered, Z)
```



**Arguments**

samples\_filtered  
a vector of samples  
Z  
a matrix of coverage

**Value**

a reshaped matrix of coverage

---

select\_by\_ks\_test      *A function to remove outlier contigs using KS test*

---

**Description**

A function to remove outlier contigs using KS test

**Usage**

```
select_by_ks_test(sort_values)
```

**Arguments**

sort\_values      a vector of sorted values

**Value**

a vector with all values following a uniform distribution

---

test\_reasonable      *A function to test whether the result is reasonable*

---

**Description**

A function to test whether the result is reasonable

**Usage**

```
test_reasonable(a, b)
```

**Arguments**

a                  first vector of values  
b                  second vector of values

**Value**

the test result



# Index

## \* datasets

- ContigCluster1, [5](#)
- ContigCluster2, [5](#)
- max\_bin\_001, [14](#)
- max\_bin\_002, [15](#)
- max\_bin\_003, [15](#)

[compare\\_contig\\_subsets](#), [2](#)

[compare\\_sample\\_subsets](#), [3](#)

[consist\\_transfer](#), [4](#)

[contig\\_pca](#), [6](#)

ContigCluster1, [5](#)

ContigCluster2, [5](#)

[cor\\_diff](#), [6](#)

[df\\_transfer](#), [7](#)

[est\\_ptr](#), [7](#)

[est\\_ptr\\_on](#), [9](#)

[est\\_ptr\\_on\\_all](#), [10](#)

[est\\_ptrs\\_subset](#), [8](#)

[filter\\_sample](#), [11](#)

[get\\_eptr\\_stats](#), [11](#)

[iterate\\_pipelines](#), [12](#)

[ks](#), [13](#)

[lm\\_column](#), [14](#)

[lme4\\_model](#), [13](#)

max\_bin\_001, [14](#)

max\_bin\_002, [15](#)

max\_bin\_003, [15](#)

[pipeline](#), [16](#)

[reshape\\_filtered](#), [16](#)

[select\\_by\\_ks\\_test](#), [17](#)

[test\\_reasonable](#), [17](#)

[verify\\_input](#), [18](#)