

# Package ‘HEssRNA’

May 7, 2026

**Title** Heritability-Based Estimation of Sample Size for RNA-Seq Data

**Version** 1.0.1

**Description** Provides tools for estimating sample sizes primarily based on heritability, while also considering additional parameters such as statistical power and fold change. The package normalizes heritability values according to trait-specific heritability and classification to enhance accuracy in sample size estimation.

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**Suggests** car, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Imports** tidyr, dplyr, magrittr, stats, utils, DESeq2, ssizeRNA, Rdpack

**NeedsCompilation** no

**Author** Naina Kumari [aut],  
Jagajjit Sahu [aut],  
Sarika Jaiswal [aut, cre],  
Mir Asif Iquebal [aut],  
Dinesh Kumar [aut]

**Maintainer** Sarika Jaiswal <sarika@icar.gov.in>

**Repository** CRAN

**Date/Publication** 2025-01-10 13:50:01 UTC

## Contents

hIndxMeanCalc4Traits . . . . .	2
powerCalc . . . . .	3
prcsDF4modelInhouse . . . . .	4
smplSizPred . . . . .	5
smplSizPredModel . . . . .	7
<b>Index</b>	<b>9</b>

---

hIndxMeanCalc4Traits *Calculate Mean Heritability Index for Traits*

---

## Description

This function processes heritability index data, filtering out empty trait names, and calculates the mean heritability for each unique trait. The resulting output is a data frame with traits and their corresponding mean heritability values.

## Usage

```
hIndxMeanCalc4Traits(hIndexValDF)
```

## Arguments

`hIndexValDF` A data frame containing heritability index values with at least two columns: `Trait.name` and `Heritability`. The `Trait.name` column should contain trait identifiers, and the `Heritability` column should contain numeric heritability values.

## Value

A data frame with two columns: `Trait.name` and `MeanValue`, where `MeanValue` represents the mean heritability for each trait.

## References

Hu et al. (2018) [doi:10.1093/nar/gky1084](https://doi.org/10.1093/nar/gky1084)

## Examples

```
# Example of usage:
hIndexValDF <- data.frame(Trait.name = c("Trait1", "Trait2", "Trait1", "Trait2"),
                        Heritability = c(0.5, 0.6, 0.7, 0.8))
result <- hIndxMeanCalc4Traits(hIndexValDF)
print(result)
```

---

`powerCalc`*Power Calculation from gene expression data information*

---

### Description

This function takes the required input information such as count data, sample data, etc. to calculate the power. It filters the input count data, performs DESeq2 analysis to calculate differentially expressed genes (DEGs), and then calculates the power of detecting these DEGs based on simulations.

### Usage

```
powerCalc(  
  countDat,  
  smplDat,  
  alpha = 0.05,  
  thrsholdFC = 2,  
  inptNoOfReplicates = 3,  
  sims = 10  
)
```

### Arguments

<code>countDat</code>	A matrix or data frame of raw count data where rows represent genes and columns represent samples.
<code>smplDat</code>	A data frame of sample information, with at least a condition column that specifies the experimental condition of each sample.
<code>alpha</code>	The significance level (FDR threshold) used to identify differentially expressed genes. Default is 0.05.
<code>thrsholdFC</code>	The threshold for the absolute value of log <sub>2</sub> fold change used to filter DEGs. Default is 2.
<code>inptNoOfReplicates</code>	The input number of replicates based on which the power will be calculated. Default is 3.
<code>sims</code>	The number of simulations to run for power calculation. Default is 10.

### Details

Example files included with this package:

- `exmplCountDat.csv`: A toy dataset with count data.
- `exmplSampleDat.csv`: A sample dataset with metadata.

These files are stored in the `inst/extdata` directory and can be accessed using the `system.file()` function in R.

**Value**

A data frame containing the calculated power values and related parameters.

**References**

Bi et al. (2016) [doi:10.1186/s1285901609949](https://doi.org/10.1186/s1285901609949) Love et al. (2014) [doi:10.1186/s1305901405508](https://doi.org/10.1186/s1305901405508)

**Examples**

```
# Load example files
countDatPath <- system.file("extdata", "exmplCountDat.csv", package = "HEssRNA")
smp1DatPath <- system.file("extdata", "exmplSampleDat.csv", package = "HEssRNA")

if (file.exists(countDatPath) && file.exists(smp1DatPath)) {
  countDat <- read.csv(countDatPath)
  smp1Dat <- read.csv(smp1DatPath)

  result <- powerCalc(countDat, smp1Dat)
  print(result$PowerResults)
} else {
  warning("Example data files not found.")
}
```

---

prcesDF4modelInhouse *Process Data Frame in In-House Format for Model Building*

---

**Description**

This function takes a data frame in an in-house format and processes it to make it in longer format and round the value of the power to 3 digits for building a model. It reshapes the data from a wide format to a long format, extracting and manipulating columns related to replicate numbers and power values. This function is needed when user has a data frame similar to the in-house format. For the purpose of creating model the user should also have Heritability class and log fold change value too.

**Usage**

```
prcesDF4modelInhouse(df4modelInhouseFmt)
```

**Arguments**

df4modelInhouseFmt

A data frame containing the input data in in-house format. The columns should include replicate columns named starting with "R" (e.g., R1, R2, etc.).

**Value**

A data frame in long format with columns:

NoOfReplicates Numeric representation of the replicate number extracted from column names (R1, R2, etc.).

pwr Power values rounded to 3 decimal places corresponding to the replicate number.

**Examples**

```
# Example of usage:
df <- data.frame(
  Gene = c("Gene1", "Gene2"),
  R1 = c(0.85, 0.90),
  R2 = c(0.88, 0.91),
  R3 = c(0.83, 0.89)
)
result <- prcesDF4modelInhouse(df)
print(result)
```

---

smp1SizPred	<i>Predict Number of Replicates Based on Heritability, Power, and Fold Change</i>
-------------	---

---

**Description**

This function predicts the number of replicates required for a given experiment based on heritability, power, fold change, and tissue type. The model is constructed using the provided data, and the prediction is adjusted based on the selected trait's mean heritability value. The function ensures that the predicted replicates are valid, rounding negative or unrealistic values to sensible minimums based on the heritability class.

**Usage**

```
smp1SizPred(
  df4model = df4modelInpt,
  hIndexMeanDFinput = hIndexMeanDF,
  heritabilityClass,
  inptPwr,
  fc,
  trait = NULL,
  tissue = NULL
)
```

**Arguments**

df4model	A data frame containing the input data for the model. It should include the following columns: NoOfReplicates, HeritabilityValue, pwr, FoldChange, and optionally Tissue.
hIndexMeanDFinput	A data frame containing the mean heritability values for each trait. It should include at least the columns Trait.name and MeanValue.
heritabilityClass	A character string specifying the heritability class used for filtering and adjusting the prediction. Possible values are "low", "mid", and "high".
inptPwr	A numeric value representing the power used in the model.
fc	A numeric value representing the fold change used in the model.
trait	An optional parameter specifying the trait. If provided, the heritability value for the trait will be used to adjust the heritability class values.
tissue	An optional parameter specifying the tissue type. If provided, the model will include tissue as a factor in the regression. If not provided, tissue is excluded.

**Value**

A numeric value representing the predicted number of replicates. The value is rounded to the nearest whole number and adjusted to ensure it is valid for the selected heritability class.

**References**

Sun et al. (2017) [doi:10.1093/nar/gkx204](https://doi.org/10.1093/nar/gkx204)

**Examples**

```
# Example usage:
df4modelInpt <- data.frame(
  NoOfReplicates = c(3, 5, 7, 9, 11),
  HeritabilityClass = c("high", "mid", "low", "high", "mid"),
  HeritabilityValue = c(0.5, 0.6, 0.7, 0.5, 0.6),
  pwr = c(0.8, 0.9, 0.85, 0.88, 0.86),
  FoldChange = c(2, 3, 2.5, 2.8, 3.2),
  Tissue = c("Liver", "Liver", "Kidney", "Liver", "Kidney")
)
hIndexMeanDF <- data.frame(Trait.name = c("Trait1", "Trait2"),
  MeanValue = c(0.3, 0.5))
NoOfReplicatesPred <- smp1SizPred(df4model = df4modelInpt,
  hIndexMeanDFinput = hIndexMeanDF,
  heritabilityClass = "mid",
  inptPwr = 0.85,
  fc = 2.5,
  trait = "Trait1",
  tissue = "Liver")
print(NoOfReplicatesPred)
```

---

smp1SizPredModel      *Generate a Linear Model for Sample Size Prediction*

---

## Description

This function generates a linear regression model to predict the number of replicates (NoOfReplicates) based on heritability, power, fold change, and tissue type. The model is generated depending on whether the tissue information is provided in the data. The function returns the fitted model.

## Usage

```
smp1SizPredModel(  
  df4model = df4modelInpt,  
  heritabilityClass,  
  inptPwr,  
  fc,  
  trait = NULL,  
  tissue = NULL  
)
```

## Arguments

df4model	A data frame containing the input data for the model. It should include the following columns: NoOfReplicates, HeritabilityValue, pwr, FoldChange, and optionally, Tissue.
heritabilityClass	A character value indicating the class of heritability used for filtering the data.
inptPwr	A numeric value representing the power used in the model.
fc	A numeric value representing the fold change used in the model.
trait	An optional parameter to specify the trait. If provided, it can be used for further filtering, but it's not currently used in the function.
tissue	An optional parameter specifying the tissue type. If provided, the model will include the tissue information in the regression. If not provided, the model will exclude tissue information.

## Value

A linear model object (lm class), which contains the fitted linear regression model for the number of replicates prediction.

## References

Sun et al. (2017) [doi:10.1093/nar/gkx204](https://doi.org/10.1093/nar/gkx204)

**Examples**

```
# Example usage:
df4modelInpt <- data.frame(
  NoOfReplicates = c(3, 5, 7, 9, 11),
  HeritabilityClass = c("high", "mid", "low", "high", "mid"),
  HeritabilityValue = c(0.5, 0.6, 0.7, 0.5, 0.6),
  pwr = c(0.8, 0.9, 0.85, 0.88, 0.86),
  FoldChange = c(2, 3, 2.5, 2.8, 3.2),
  Tissue = c("Liver", "Liver", "Kidney", "Liver", "Kidney")
)

# Fit the model
model <- smp1SizPredModel(
  df4model = df4modelInpt,
  heritabilityClass = "high",
  inptPwr = 0.8,
  fc = 2,
  tissue = "Liver"
)

# Summarize the results
summary(model)
```

# Index

hIndxMeanCalc4Traits, [2](#)

powerCalc, [3](#)

prcesDF4modelInhouse, [4](#)

smp1SizPred, [5](#)

smp1SizPredModel, [7](#)