

# Package ‘TTCA’

May 7, 2026

**Type** Package

**Title** Transcript Time Course Analysis

**Version** 0.1.1

**Date** 2017-01-29

**Author** Marco Albrecht

**Maintainer** Marco Albrecht <marco.albrecht@posteo.de>

**Description** The analysis of microarray time series promises a deeper insight into the dynamics of the cellular response following stimulation. A common observation in this type of data is that some genes respond with quick, transient dynamics, while other genes change their expression slowly over time. The existing methods for detecting significant expression dynamics often fail when the expression dynamics show a large heterogeneity. Moreover, these methods often cannot cope with irregular and sparse measurements. The method proposed here is specifically designed for the analysis of perturbation responses. It combines different scores to capture fast and transient dynamics as well as slow expression changes, and performs well in the presence of low replicate numbers and irregular sampling times. The results are given in the form of tables including links to figures showing the expression dynamics of the respective transcript. These allow to quickly recognise the relevance of detection, to identify possible false positives and to discriminate early and late changes in gene expression. An extension of the method allows the analysis of the expression dynamics of functional groups of genes, providing a quick overview of the cellular response. The performance of this package was tested on microarray data derived from lung cancer cells stimulated with epidermal growth factor (EGF). Paper: Albrecht, Marco, et al. (2017)<[DOI:10.1186/s12859-016-1440-8](https://doi.org/10.1186/s12859-016-1440-8)>.

**LazyData** TRUE

**License** EUPL

**Depends** R(>= 2.10)

**Imports** Matrix, quantreg, tcltk2, RISmed, VennDiagram, MASS

**RoxygenNote** 5.0.1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2017-01-29 18:21:13

## Contents

annot . . . . .	2
annotation . . . . .	3
Control . . . . .	3
EGF . . . . .	4
TTCA . . . . .	5
<b>Index</b>	<b>9</b>

---

annot	<i>Data file annot</i>
-------	------------------------

---

## Description

Links probeset ID with gene\_name

## Usage

```
data("annot")
```

## Format

A data frame with 5000 observations on the following 2 variables.

probeset\_id a character vector

gene\_name a character vector

## Source

TTCA: An R Package for the identification of differentially expressed genes in time course microarray data

## Examples

```
data(annot)
```

---

annotation

*Data file annotation*

---

**Description**

Gives the possibility to link the resulttable with further characteristics.

**Usage**

```
data("annotation")
```

**Format**

A data frame with 5000 observations on the following 2 variables.

GeneOntology a character vector

Property a character vector

**Source**

TTCA: An R Package for the identification of differentially expressed genes in time course microarray data

**Examples**

```
data(annotation)
```

---

Control

*Control time course*

---

**Description**

The second time course representing the control. Lung cancer cell line.

**Usage**

```
data("Control")
```

**Format**

A data frame with 5000 observations on the following 11 variables.

B\_C\_2\_0 a numeric vector  
B\_C\_3\_0 a numeric vector  
B\_C\_2\_0.5 a numeric vector  
B\_C\_1\_1 a numeric vector  
B\_C\_3\_4 a numeric vector  
B\_C\_2\_6 a numeric vector  
B\_C\_1\_24 a numeric vector  
B\_C\_3\_24 a numeric vector  
B\_C\_1\_48 a numeric vector  
B\_C\_2\_48 a numeric vector  
B\_C\_3\_48 a numeric vector

**Source**

TTCA: An R Package for the identification of differentially expressed genes in time course microarray data

**Examples**

```
data(Control)
```

---

EGF

*EGF time course data*

---

**Description**

The first time course representing the sample with epidermal growth factor stimulation. Lung cancer cell line.

**Usage**

```
data("EGF")
```

**Format**

A data frame with 5000 observations on the following 16 variables.

B\_C\_2\_0 a numeric vector  
B\_C\_3\_0 a numeric vector  
B\_EGF\_1\_0.5 a numeric vector  
B\_EGF\_3\_0.5 a numeric vector

B\_EGF\_1\_1 a numeric vector  
B\_EGF\_3\_2 a numeric vector  
B\_EGF\_1\_4 a numeric vector  
B\_EGF\_1\_6 a numeric vector  
B\_EGF\_1\_8 a numeric vector  
B\_EGF\_2\_12 a numeric vector  
B\_EGF\_2\_18 a numeric vector  
B\_EGF\_1\_24 a numeric vector  
B\_EGF\_3\_24 a numeric vector  
B\_EGF\_1\_48 a numeric vector  
B\_EGF\_2\_48 a numeric vector  
B\_EGF\_3\_48 a numeric vector

### Source

TTCA: An R Package for the identification of differentially expressed genes in time course microarray data

### Examples

```
data(EGF)
```

---

TTCA

*TTCA: Transcript Time Course Analysis*

---

### Description

Background: The analysis of microarray time series promises a deeper insight into the dynamics of the cellular response following stimulation. A common observation in this type of data is that some genes respond with quick, transient dynamics, while other genes change their expression slowly over time. The existing methods for detecting significant expression dynamics often fail when the expression dynamics show a large heterogeneity. Moreover, these methods often cannot cope with irregular and sparse measurements. Results: The method proposed here is specifically designed for the analysis of perturbation responses. It combines different scores to capture fast and transient dynamics as well as slow expression changes, and performs well in the presence of low replicate numbers and irregular sampling times. The results are given in the form of tables including links to figures showing the expression dynamics of the respective transcript. These allow to quickly recognise the relevance of detection, to identify possible false positives and to discriminate early and late changes in gene expression. An extension of the method allows the analysis of the expression dynamics of functional groups of genes, providing a quick overview of the cellular response. The performance of this package was tested on microarray data derived from lung cancer cells stimulated with epidermal growth factor (EGF). Paper: Albrecht, Marco, et al. (2017)<DOI:10.1186/s12859-016-1440-8>.

**Usage**

```
TTCA(grp1, grp1.time, grp2, grp2.time, lambda = 0.6, annot = NA,
     annotation = "annotation", timeInt = NULL, pVal = 0.05,
     codetest = FALSE, file = getwd(), MaxPics = 10000, Stimulus1 = "",
     Stimulus2 = "", S = "gene", mapGO = "", PeakMode = "norm")
```

**Arguments**

grp1	Data set with longitudinal sampled data (data.frame)
grp1.time	Time points for data set 1 (vector like: c(0,0,0.5,1,2,4,6,8,12,12))
grp2	Data set with longitudinal sampled data for comparison (data.frame)
grp2.time	Time points for data set 2 (vector like: c(0,0,0.5,3,2,4,6,8,12,12,24))
lambda	Smoothing parameter in penalty term of quantil regression (default: lambda=0.6). Adjust, if fit is too strict or too flexible.
annot	Annotation for pictures and result (Data.frame with 2 columns with ID and GeneName). (Default: annot=NA)
annotation	Merges the TTCA by rowname with a table of your wish. Example: annotation<-annotation[,c("probeset_id", "gene_name", "transkript_id", "GO_BP", "GO_CC", "GO_mf")] (default: annotation="annotation")
timeInt	Defines early, middle and late time period. Defines the middle time period between 4 h and 12 h with timeInt<-c(4,12). (default: timeInt=NULL)
pVal	P-value for the local hypothesis test (default: 0.05).
codetest	Reduces the data set to 200 features for a quick run of the program. (default: codetest=FALSE)
file	Result folder will be saved at this location (default: file=getwd()).
MaxPics	Limits the number of plots (default: MaxPics=10000)
Stimulus1	Searches this term together with the gene name in PubMed. Stimulus1="Insulin+like+growth+factor" ( default: Stimulus1="")
Stimulus2	Searches this term together with the gene name in PubMed. Stimulus2="epidermal+growth+factor" ( default: Stimulus2="")
S	Defines mode. S == "GO" changes programm to gene ontology mode (default: S="gene")
mapGO	Link genes to Gene Ontology terms (default: mapGO="")
PeakMode	Peakmode "norm" uses variance between replicates. If changed to another character value, a normal hypothesis test will be conducted (default: PeakMode="norm")

**Details**

The package has not be applied to Hi-Seq data yet. The problem is the huge variety in the read counts. An additional transformation of normalized Hi-Seq data might be an option to scale the values between two values like 0 and 1 (Simple idea: `Datalog<-log(data, base = max(data))`). Not tested. IF you are interested to adjust my package to sequence data, feel free to contact me.

**Value**

The R-package delivers a table with different significance values, rankings, p-values. Moreover, it will plot the most important time courses and quality control images.

**Examples**

```
## Not run:

#####
#### Gene-ANALYSE
#####
require(quantreg);require(VennDiagram);require(tcltk2); require(tcltk);
require(RISmed);require(Matrix)
data(EGF,Control,annot,annotation)

S="gene"
Control.time <- c(0,0,0.5,1,4,6,24,24,48,48,48)
EGF.time <- c(0,0,0.5,0.5,1,2,4,6,8,12,18,24,24,48,48,48)
file = paste0(getwd(),"/TTCA_Gene")
dir.create(file)
#####
TTCAresult<-TTCA(grp1=EGF, grp1.time=EGF.time, grp2=Control, grp2.time=Control.time,S="gene",
lambda=0.6, annot=annot, annotation=annotation,pVal=0.05,codetest=FALSE,
file=file, Stimulus1="epidermal+growth+factor", timeInt=c(4,12), MaxPics =10000)

## End(Not run)

## Not run:
#####
#### GO-ANALYSE
#####
require(quantreg);require(VennDiagram);require(tcltk2); require(tcltk);
require(RISmed);require(Matrix)
#source("https://bioconductor.org/biocLite.R")
#biocLite("biomaRt")
library(biomaRt)
data(EGF,Control,annot,annotation)

require(biomaRt)
ensembl <- useMart("ENSEMBL_MART_ENSEMBL",dataset="hsapiens_gene_ensembl")
mapGO <- getBM(attributes=c("go_id","name_1006","affy_hugene_2_0_st_v1"),
filters = 'affy_hugene_2_0_st_v1', values=rownames(annot), mart =ensembl)
colnames(mapGO)<-c("go_id","GO_name","probeset_id")

S="GO"
Control.time <- c(0,0,0.5,1,4,6,24,24,48,48,48)
EGF.time <- c(0,0,0.5,0.5,1,2,4,6,8,12,18,24,24,48,48,48)
file = paste0(getwd(),"/TTCA_GO")
dir.create(file)
```

```
TTCAResult<-TTCA(grp1=EGF, grp1.time=EGF.time, grp2=Control, grp2.time=Control.time,  
S="G0", pVal=0.05,lambda=0.6,codetest=FALSE, file=file,  
Stimulus1="epidermal+growth+factor", timeInt=c(4,12),  
MaxPics=10000, mapG0=mapG0)  
  
## End(Not run)
```

# Index

## \* datasets

- annot, [2](#)
- annotation, [3](#)
- Control, [3](#)
- EGF, [4](#)

- annot, [2](#)
- annotation, [3](#)

- Control, [3](#)

- EGF, [4](#)

- TTCA, [5](#)