# T-rEx

*A S. cerevisiae* Transcription factor Explorer toolbox

David Bergenholm, Christoph Börlin, Petter Holland, & Jens Nielsen

Contents

[T-rEx 1](#_Toc21698908)

[Data 3](#_Toc21698909)

[Transcription factor summary 4](#_Toc21698910)

[Select transcription factor 4](#_Toc21698911)

[Peaklist 4](#_Toc21698912)

[Peaklist advanced 4](#_Toc21698913)

[Transcription factor binding data 5](#_Toc21698914)

[Select gene 5](#_Toc21698915)

[Select Transcription Factor 5](#_Toc21698916)

[Legend 5](#_Toc21698917)

[Motif 5](#_Toc21698918)

[Sequence relative to TSS 5](#_Toc21698919)

[Download Data 5](#_Toc21698920)

[Transcription Factor GO Analysis 6](#_Toc21698921)

[Dataset 6](#_Toc21698922)

[Exclude or include data 6](#_Toc21698923)

[GO-term 6](#_Toc21698924)

[Test 6](#_Toc21698925)

[Include dataset 10](#_Toc21698926)

[Download and use T-rEx as stand-alone 11](#_Toc21698927)

[T-rEx stand-alone usage 11](#_Toc21698928)

[Libraries 12](#_Toc21698929)

[References 13](#_Toc21698930)

## Data

All data included in the T-rEx toolbox is generated from the ChIP-exo data pipeline:

Börlin, C. S., Bergenholm, D., Holland, P., & Nielsen, J. (2019). A bioinformatic pipeline to analyze ChIP-exo datasets. Biology Methods and Protocols, 4(1). doi:10.1093/biomethods/bpz011

Where the data comes from the publications

Bergenholm, D., Liu, G., Holland, P., & Nielsen, J. (2018). Reconstruction of a Global Transcriptional Regulatory Network for Control of Lipid Metabolism in Yeast by Using Chromatin Immunoprecipitation with Lambda Exonuclease Digestion. mSystems, 3(4). doi:10.1128/mSystems.00215-17

Holland, P., Bergenholm, D., Börlin, C. S., Liu, G., & Nielsen, J. (2019). Predictive models of eukaryotic transcriptional regulation reveals changes in transcription factor roles and promoter usage between metabolic conditions. Nucleic Acids Res. doi:10.1093/nar/gkz253

## Transcription factor summary

The first page in the toolbox includes a summary of the transcription factors included in the toolbox.

Select transcription factor

Generates a table of identified targets and number of peaks. The consensus motif (generated from the meme-suit) and the reads profile as well as the peak distribution profile is also included.

The targets list is downloadable in two formats.

Peaklist: Contains the Target gene, the condition and amount of bindings identified on said gene.

Peaklist advanced: Contains the Target gene, chromosome, position of gene start, strand of gene, distance of the peak from TSS, the strength of binding for each individual peak and the condition. Were the strength referring to log2(S/N).

## Transcription factor binding data

Select gene

allows you to find a gene either by the common name or the systematic name where one can search through the list.

A Gene info box will appear with relevant information about the selected gene. This includes the systematic and common name, the gene start and chromosome, orientation, what gene is upstream and what gene is downstream.

Select Transcription Factor

allows the user to include one or more TFs. By pressing “Load Data” the transcription factor binding profile will be displayed.

Legend

has three options are available.

TF BS: Include all found binding sites based on the same data as displayed in Transcription factor summary.

TATA: Include the TATA/like box identified by Rhee, H. S., & Pugh, B. F. (2012). Genome-wide structure and organization of eukaryotic pre-initiation complexes. Nature, 483, 295. doi:10.1038/nature10799

Fixed y-axis: Make scaling of the data easier to compare between conditions

### Motif

allows the user to include one or more motifs

### Sequence relative to TSS

Allows the user to get a specific sequence out relative to the TSS

### Download Data

The user can download the reads that are displayed for the individual conditions.

## Transcription Factor GO Analysis

This page allows the user to do statistical testing of the transcription factors included in the toolbox.

### Dataset

### Exclude or include data

The user can here choose to include all genes in the Cen.PK genome or exclude dubious ORFs and only include genes available in Yeast8.

### GO-term

The user can select one or several GO-terms. All the GO-terms included is displayed in the table “The selected GO-terms from search result”.

### Test

User selected statistical test to be used on the selected GO-term.

#### Data treatment on statistical tests

The Stat\_plot\_func is the overall function that generates the plots displayed on the transcription factor GO analysis page.

Stat\_plot\_func(CCM, downloadVal, name.Cond, GOterm, name.TF, Test, data.New.TF, nclu)

CCM = exclude or include data (user defined)

downloadVal = should the generated data be downloaded or not (user defined)

condtionname = the current condition

goterm = the GO-term (user defined)

name.TF = all TF

Test = statistical test to be performed (user defined)

New.TF = if there is a user included dataset (user defined)

nclu = used in cluster\_function to set number of cluster (user defined)

goterm\_func returns the GO-term and extracts the genes.

GOterm<-goterm\_func(datain)

datain = GO-terms (user defined)

Stat\_data\_func loads the data “geneTargetList” and returns a list with all targets. This data contains the number of peaks for each gene and TF.

Stat.Data <-Stat\_data\_func(name.TF, conditionname, New.TF)

name.TF = all TF

condtionname = the current condition

New.TF = if there is a user included dataset (user defined)

data\_func returns a file containing the genes and TFs that has associated peaks to that gene.

The function takes the Stat.Data and merges it with val1 (include or exclude data) and then merges it with the genes identified from GO-term.

Stat.GOterm.data <-data\_func(Stat.Data, conditionname, GOterm.Data,val1)

Stat.Data = generated from Stat\_data\_func

conditionname = current condition

GOterm.Data = generated from goterm\_func

The data\_treatment\_func returns a list containing the TPM value of the genes and the TF binding. This function takes the Stat.GOterm.data as an input. The data is merged with the TPM value of each gene. The function then only allows genes which has a TPM>1 to be included. Further the function only includes genes that has at least one binding of one TF. If there are no genes left after this treatment this function will throw an error. This error is propagated to the Stat\_plot\_func that will return “Not enough genes selected to do statistical testing” that is displayed for the user. All plots will display “Selected genes n of total k” where n is the genes that clears the cutoff from data\_treatment\_func and k is the total genes in the selected GO-term.

Treated.Data<-data\_treatment\_func(Stat.GOterm.data$data1, conditionname, name.TF)

Stat.GOterm.data$data1 = generated from data\_func

conditionname = current condition

name.TF = all TF

The fishers\_test\_func returns a heatmap with the oddsratio and pvalue for transcription factor occurrence. First all TF bindings are turned to 1 or 0 (binding or no binding). Thus, each TF will have a value “Bound” or “NotBound” for each gene. Each TF is then tested against all the other TFs. The table used has the format of

|  |  |
| --- | --- |
| Bound by both TF | Bound by only TF1 |
| Bound by only TF2 | Not Bound by any TF |

Example

TF1

|  |  |
| --- | --- |
| GeneA | Bound |
| GeneB | NotBound |
| GeneC | Bound |
| GeneD | NotBound |
| GeneE | NotBound |

TF2

|  |  |
| --- | --- |
| GeneA | Bound |
| GeneB | Bound |
| GeneC | Bound |
| GeneD | NotBound |
| GeneE | NotBound |

Resulting table

|  |  |
| --- | --- |
| 2 | 0 |
| 1 | 2 |

The function then uses the fisher.test function in R

Fisher<-fisher.test(table)

From here the oddsratio and pvalues are extracted

Plot<-fishers\_test\_func(Treated.Data$x, txtstr, downloadVal)

Treated.Data$x = generated from data\_treatment\_func

Txtstr = the plot title

downloadVal = should the data be downloaded or not

heatmap\_func returns a heatmap with the selected GO-term genes and the TF binding

Plot<-heatmap\_func(Treated.Data,txtstr,downloadVal)

Treated.Data = generated from data\_treatment\_func

Txtstr = the plot title

downloadVal = should the data be downloaded or not

net\_func returns a network plot with weighted node sizes based on the number of edges. The function used is the ggnet2 and the network type is “bipartite”

Plot<-net\_func(Treated.Data, txtstr)

Treated.Data = generated from data\_treatment\_func

Txtstr = the plot title

The cluster\_func returns a Partioning Around Medoids (PAM) for the selected GO-terms using the number of peaks detected for each TF and each gene. The PAM takes the pairwise distance into account and are therefore much more robust when it comes to outliers compared to k-means, which might be the case when selecting genes based on GO-terms since there are overlaps of genes between different GO-terms. For more info see the cluster package ?pam.

Plot<-cluster\_func(Treated.Data, downloadVal, nclu)

Treated.Data = generated from data\_treatment\_func

downloadVal = should the data be downloaded or not

nclu = number of clusters (user defined)

The model\_zero\_func takes the transcription factors binding and the TPM value of the genes and makes a linear model with no TF-TF interactions

Plot<-model\_zero\_func(Treated.Data, txtstr, downloadVal, name.TF)

Treated.Data = generated from data\_treatment\_func

downloadVal = should the data be downloaded or not

Txtstr = the plot title

name.TF = all TF

## Include dataset

The user can include a new dataset directly to T-rEx by browsing the files.

The files must be generated through the ChIP-exo data pipeline.

For binding profiles the files must containing one or more of the condition name: Glu, Nit, Eth, Ana. i.e “TF\_Glu\_\*.wigLike”

To do statistical testing the file "TF\_geneTargetList\_\*.csv” needs to be provided

To get the individual binding sites the file “TF\*\_GEManalysis\_\*.csv” needs to be provided

## Download and use T-rEx as stand-alone

If the user has more transcription factors to include or want to use a complete other set of conditions T-rEx is downloadable through GitHub.

### T-rEx stand-alone usage

To start using T-rEx on your computer the libraries needs to be installed. The data used must be run through the ChIP-exo data pipeline.

To change the set of transcription factor (or any other DNA binding protein) go to server.R

name.TF = the name of the TFs included i.e name.TF<-c(“Aaa1”)

myColors.TF = number of colors that match number of TFs included i.e myColors.TF<-c(“red”)

The WigLike files should be placed in the “WigLike” folder. The “GEManalysis” and the “geneTargetList” files should be placed in the “GEMPeaks” folder

To change the conditions go to server.R

name.Cond = the names of conditions used i.e name.Cond<-c(“Gal”)

TPM.data = a .csv file named “TPM.csv” that contains at least three columns with the names i.e “Gene”, “CommonName”, “Gal”. This file should replace the “TPM.csv” file in folder “Resources”

If a TPM file is not provided then a dummyfile is generated, thus there will be no cutoff based on the gene TPM value.

## Libraries

The T-reX application relies on several packages to function properly. All functions are listed below

CRAN and Bioconductor packages: Shiny, RColorBrewer, dplyr, gdata, gplots, gtools, scales, GGally, network, ggpubr, factoextra, shinyWidgets, shinycssloaders, cluster, Biostrings and sna.

Other packages used ggplot2 (Wickham., 2016), MASS (Venables, 2002), biomaRt (Durinck, Spellman, Birney, & Huber, 2009), IRanges (Lawrence et al., 2013).

## References

Durinck, S., Spellman, P. T., Birney, E., & Huber, W. (2009). Mapping identifiers for the integration of genomic datasets with the R/Bioconductor package biomaRt. *Nat Protoc, 4*(8), 1184-1191. doi:10.1038/nprot.2009.97

Lawrence, M., Huber, W., Pagès, H., Aboyoun, P., Carlson, M., Gentleman, R., . . . Carey, V. J. (2013). Software for Computing and Annotating Genomic Ranges. *PLOS Computational Biology, 9*(8), e1003118. doi:10.1371/journal.pcbi.1003118

Venables, W. N. R., B. D. . (2002). Modern Applied Statistics with S. Fourth Edition. *Springer, New York.* doi:ISBN 0-387-95457-0

Wickham., H. (2016). ggplot2: Elegant Graphics for Data Analysis. . *Springer-Verlag New York*.