Supplement for TIMEOR: a web-based tool to uncover temporal regulatory mechanisms from multi-omics data

TABLE OF CONTENTS

- Resource Table
- Method Details
  - Installation
  - Pre-processing
    - Processing Raw Data
    - Load Count Matrix, Normalize, and Correct Data
  - Primary Analysis
  - Secondary Analysis
    - Enrichment
    - Factor Binding
    - Temporal Relations
    - Current Stand-Alone GRN Inference Methods
- Method and Question Choice Assistance
- Result Details
  - Simulations
  - Real Data Processing
- Tutorial for Web Server
- Full Bibliography

RESOURCE TABLE

Resource	**Name**	**Access**

Deposited Data

RNA-seq data	NCBI	https://www.ncbi.nlm.nih.gov/
Chip-seq data	ENCODE	https://encodeproject.org/
Transcription factor gene lists for *Drosophila melanogaster, Homo sapiens, Mus musculus*	Animal TFDB 3.0 (41)	http://bioinfo.life.hust.edu.cn/AnimalTFDB/#!/download
Protein-protein interactions	STRINGdb	https://string-db.org/cgi/input.pl
Drosophila melanogaster genome for HISAT2 (BDGP6)	Kim Lab	http://daehwankimlab.github.io/hisat2/download/#d-melanogaster
Mus musculus genome for HISAT2 (GRCm38)	Kim Lab	http://daehwankimlab.github.io/hisat2/download/#m-musculus
Homo sapiens genome for HISAT2 (GRCh38)	Kim Lab	http://daehwankimlab.github.io/hisat2/download/#h-sapiens
Drosophila melanogaster	Illumina	https://support.illumina.com/sequencing/sequen

genome for Bowtie2 (BDGP6)		cing_software/igenome.html
Homo sapiens genome for Bowtie2 (GRCh38)	Illumina	https://support.illumina.com/sequencing/sequencing_software/igenome.html
Mus musculus genome for Bowtie2 (GRCm38)	Illumina	https://support.illumina.com/sequencing/sequencing_software/igenome.html
Simulations	This paper	https://github.com/ashleymaeconard/TIMEOR.git
Motif collection for *Drosophila melanogaster, Homo sapiens,* and *Mus musculus*	cisTarget databases	https://resources.aertslab.org/cistarget/databases/homo_sapiens/hg38/refseq_r80/mc9nr/gene_based/hg38__refseq-r80__10kb_up_and_down_tss.mc9nr.feather https://resources.aertslab.org/cistarget/databases/drosophila_melanogaster/dm6/flybase_r6.02/mc8nr/gene_based/dm6-5kb-upstream-full-tx-11species.mc8nr.feather https://resources.aertslab.org/cistarget/databases/mus_musculus/mm10/refseq_r80/mc9nr/gene_based/mm10__refseq-r80__10kb_up_and_down_tss.mc9nr.feather
Genes gtf *Drosophila melanogaster*		https://support.illumina.com/sequencing/sequencing_software/igenome.html (Drosophila_melanogaster/Ensembl/BDGP6/Annotation/Archives/archive-2015-07-23-16-41-33/Genes/gene.gtf)
Genes gtf *Homo sapiens*	Illumina	https://support.illumina.com/sequencing/sequencing_software/igenome.html (Homo_sapiens/NCBI/GRCh38/Annotation/Archives/archive-2015-08-11-09-31-31/Genes/genes.gtf) http://ftp.ensembl.org/pub/release-103/gtf/homo_sapiens/
Genes gtf *Mus musculus*	Illumina	https://support.illumina.com/sequencing/sequencing_software/igenome.html (Mus_musculus/Ensembl/GRCm38/Annotation/Archives/archive-2015-07-17-14-32-40/Genes/genes.gtf)

Software and Algorithms

Polyester (v1.24.0)	(45)	https://github.com/alyssafrazee/polyester

Encodexplorer (v2.4.0)	(44)	https://www.bioconductor.org/packages/release/bioc/html/ENCODExplorer.html
STRING R (v1.4.0)	(60)	https://www.bioconductor.org/packages/release/bioc/html/STRINGdb.html
Samtools (v1.9)	(61)	http://www.htslib.org/
SRA-tools	(62)	http://ncbi.github.io/sra-tools/
FastQC (v0.11.8)	(63)	https://www.bioinformatics.babraham.ac.uk/projects/fastqc/
MultiQC (v1.8)	(64)	https://multiqc.info/
Deeptools (v3.1.3)	(65)	https://deeptools.readthedocs.io/en/develop/#
HTSeq (v0.11.2)	(66)	https://htseq.readthedocs.io/en/master/index.html
HISAT2 (v2.1.0)	(67)	http://daehwankimlab.github.io/hisat2/
Bowtie2 (v2.3.5)	(68)	http://bowtie-bio.sourceforge.net/bowtie2/manual.shtml
DESeq2 (v1.26)	(34)	http://bioconductor.org/packages/release/bioc/html/DESeq2.html
ImpulseDE2 (v1.10.0)	(9)	http://bioconductor.org/packages/release/bioc/html/ImpulseDE2.html
Next maSigPro (v	(12)	https://www.bioconductor.org/packages/release/bioc/manuals/maSigPro/man/maSigPro.pdf
MEME (v4.11.2)	(39)	http://meme-suite.org/tools/meme
Pathview (v1.26.0)	(38)	https://bioconductor.org/packages/release/bioc/html/pathview.html
R Shiny (v 1.4.0)	(24)	https://shiny.rstudio.com/
ClusterProfiler (v3.0.4)	(69)	https://bioconductor.org/packages/release/bioc/html/clusterProfiler.html
Plotly (v4.5.2)	(25)	https://github.com/ropensci/plotly
Rcistarget (1.6.0)	(40)	https://bioconductor.org/packages/release/bioc/html/RcisTarget.html
Intervene (v0.6.4)	(70)	https://intervene.readthedocs.io/en/latest/how_to_use.html

Cluster (v2.1.0, for partition around medoids function)	(71)	https://cran.r-project.org/web/packages/cluster/cluster.pdf
Analysis pipeline method	This paper	https://github.com/ashleymaeconard/TIMEOR.git

METHOD DETAILS

TIMEOR has extensive tutorials and videos on Github for new users (https://github.com/ashleymaeconard/TIMEOR). TIMEOR is the first pipeline method to focus on comprehensive and comparative analysis of time series and multi-omics data with the goal to infer gene regulatory mechanisms. Specifically, TIMEOR pre-processes raw time series RNA-seq data by catering to its unique features through adaptive default methods before performing quality control and downstream processing (**Figure S1A, S1C, S1D**). This novel method also compares multiple methods for alignment and differential expression (DE, both categorical and continuous time points) to dynamically determine the best tools for the user's analyses (**Figure S1B, S1E**). TIMEOR automatically determines the number of gene trajectory clusters before performing enrichment and *de novo* motif analyses on each cluster (**Figure S1E, S3A**). Because RNA-seq data alone does not provide information about direct interactions between transcription factors and their target genes, it is important to consider other types of omics data. To that end, TIMEOR predicts, provides, and integrates transcription factor (TF) binding (mostly ChIP-seq) data from ENCODE (43; 44) to help the user validate temporal gene-gene interactions (**Figure S2B**). Using these validated interactions, TIMEOR constructs the temporal TF gene regulatory network (**Figure S2D, S2E**). Overall, TIMEOR's accessible RShiny (24) web interface enables researchers to determine the most suitable tools for their analyses, while producing paper ready and interactive figures. Furthermore, TIMEOR enables the user to inspect details on the methods used, their parameters and output, thereby facilitating reproducible research. See **Table 1** and **Supplementary Data 3** itemized main features.

INSTALLATION

To run TIMEOR outside of website, users may use Docker and Docker Hub. First, the TIMEOR repository must be cloned (https://github.com/ashleymaeconard/TIMEOR.git). To use Docker, it must be installed (version 20.10.0 recommended).

Docker Hub and Docker:

1. Download contents of organism genome folder (`/genomes_info/`) into desired location (e.g. `/Users/USERNAME/Desktop/test_folder/genomes_info/`) to mount later.
    a. The user is welcome to gather only the organism of interest. For example, for *Drosophila melanogaster* simply download `/genomes_info/dme/`
    b. Mouse is `/genomes_info/mmu/`
    c. Human is `/genomes_info/hsa/`
    d. Link `/genomes_info/`: https://drive.google.com/drive/folders/1KEnpCOU0dQU5p1tnEy3o9l02NE0uYnpm?usp=sharing
2. Make sure contents of `/genomes_info/` are readable. For example if using *Drosophila melanogaster*, in a console type `chmod -R 777 /Users/USERNAME/Desktop/test_folder/genomes_info/dme/`.
3. Run TIMEOR via Docker
    a. On command line type
        i. `$ docker pull ashleymaeconard/timeor:latest`
        ii. `$ docker images`
        iii. `$ docker run -v /Users/USERNAME/Desktop/test_folder/:/srv/ -p 3838:3838 <IMAGE_ID>`
4. Open TIMEOR Application is available by typing:
    a. Shiny server will be running on port 3838. Thus, in a browser visit `localhost:3838`.

Or, build Docker image (could take a while) with the following commands:

1. `$ cd /PATH/TO/TIMEOR/`
2. Build Docker image in TIMEOR directory:
    a. `$ docker build -t timeor_env .`
3. Follow instructions 3 and 4 above.
4. In another command line window
    a. `$ docker container ls`
    b. `$ docker exec -it <CONTAINER_NAME> /bin/bash/`
5. Now you have a console within Docker to run commands.

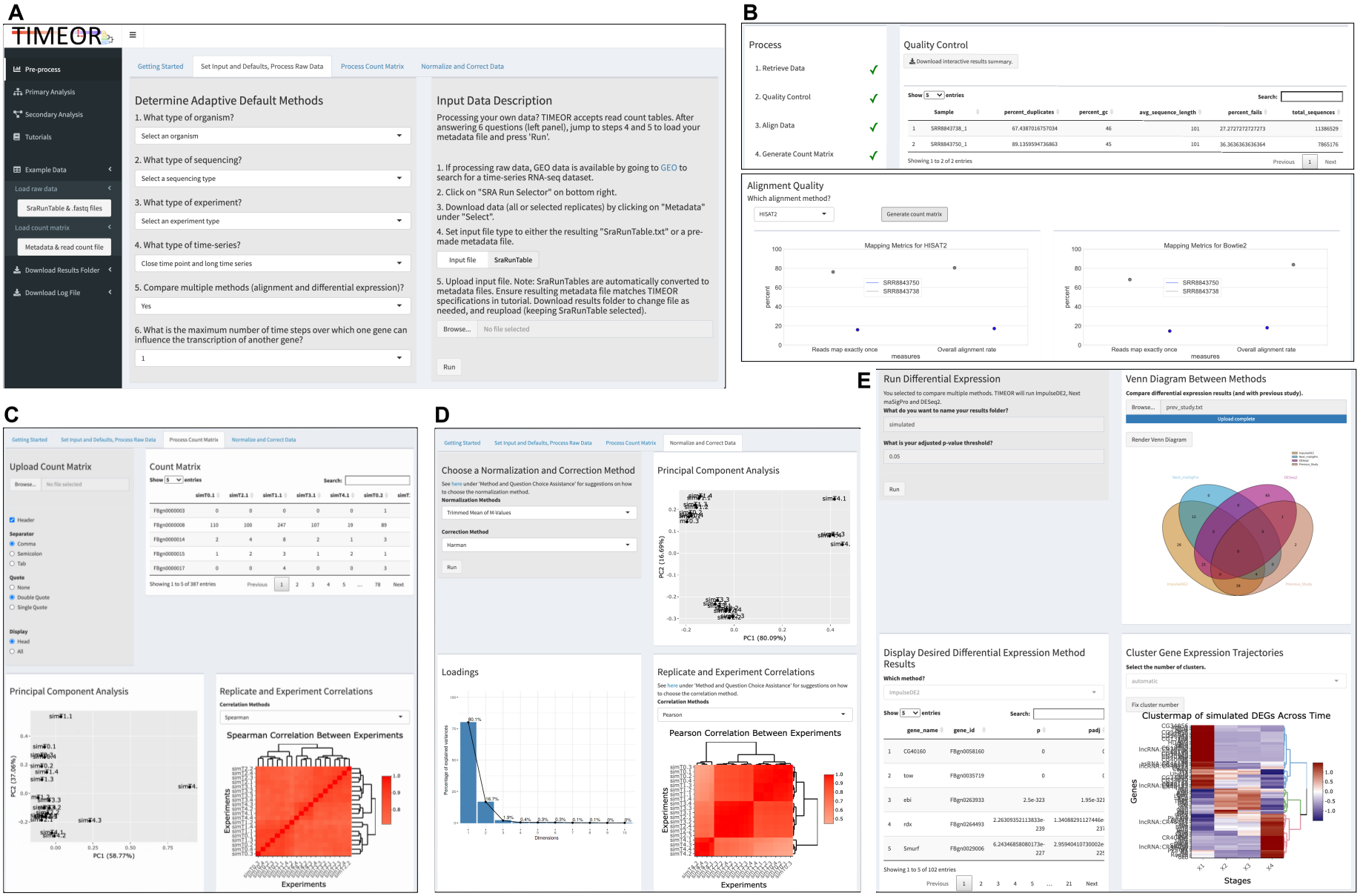
A
TIMEOR
Pre-process
Primary Analysis
Secondary Analysis
Tutorials
Example Data
Load raw data
SraRunTable & .fastq files
Load count matrix
Metadata & read count file
Download Results Folder
Download Log File
Getting Started
Set Input and Defaults, Process Raw Data
Process Count Matrix
Normalize and Correct Data
Determine Adaptive Default Methods
1. What type of organism?
Select an organism
2. What type of sequencing?
Select a sequencing type
3. What type of experiment?
Select an experiment type
4. What type of time-series?
Close time point and long time series
5. Compare multiple methods (alignment and differential expression)?
Yes
6. What is the maximum number of time steps over which one gene can influence the transcription of another gene?
1
Input Data Description
Processing your own data? TIMEOR accepts read count tables. After answering 6 questions (left panel), jump to steps 4 and 5 to load your metadata file and press 'Run'.
1. If processing raw data, GEO data is available by going to GEO to search for a time-series RNA-seq dataset.
2. Click on "SRA Run Selector" on bottom right.
3. Download data (all or selected replicates) by clicking on "Metadata" under "Select".
4. Set input file type to either the resulting "SraRunTable.txt" or a pre-made metadata file.
Input file
SraRunTable
5. Upload input file. Note: SraRunTables are automatically converted to metadata files. Ensure resulting metadata file matches TIMEOR specifications in tutorial. Download results folder to change file as needed, and reupload (keeping SraRunTable selected).
Browse...
No file selected
Run
B
Process
1. Retrieve Data
2. Quality Control
3. Align Data
4. Generate Count Matrix
Quality Control
Download interactive results summary.
Alignment Quality
Which alignment method?
HISAT2
Generate count matrix
Mapping Metrics for HISAT2
Mapping Metrics for Bowtie2
SRR8843750
SRR8843738
percent
Reads map exactly once
Overall alignment rate
measures
C
Upload Count Matrix
Count Matrix
Principal Component Analysis
Replicate and Experiment Correlations
Spearman Correlation Between Experiments
Experiments
D
Choose a Normalization and Correction Method
Trimmed Mean of M-Values
Harman
Principal Component Analysis
PC1 (80.09%)
PC2 (16.69%)
Loadings
Replicate and Experiment Correlations
Pearson
Pearson Correlation Between Experiments
E
Run Differential Expression
What do you want to name your results folder?
simulated
What is your adjusted p-value threshold?
0.05
Venn Diagram Between Methods
Compare differential expression results (and with previous study).
prev_study.txt
Upload complete
Render Venn Diagram
Display Desired Differential Expression Method Results
Which method?
ImpulseDE2
Cluster Gene Expression Trajectories
Select the number of clusters.
automatic
Fix cluster number
Clustermap of simulated DEGs Across Time
Genes
Stages

Figure S1. TIMEOR Application Pre-processing and Primary Analysis.
A. In the Pre-processing stage in the "Process Raw Data" tab, the user chooses RNA-seq time series data and answers six questions about their data and desired analyses. TIMEOR subsequently sets the adaptive default methods that are customized to the user data.
B. TIMEOR then retrieves and checks the data quality, providing interactive results to the user. These data are then aligned using either one or two alignment methods (Bowtie2 and HISAT2), and the user can choose which gives optimal results. The alignment .bam files are then used to create a gene-read count matrix which is then passed to the next tab "Load Count Matrix".
C. The gene-read count matrix is visualized through both interactive principal component analysis (PCA) and replicate (i.e. sample) correlation plots. Note that the user can pause the analysis at any time by clicking the "save your place" button and saving the link.
D. The gene-read count matrix can then be passed to the "Normalize and Correct Data" tab where it is normalized (trimmed mean of M-values and upper quartile) and corrected. These altered data are again visualized using PCA and replicate (i.e. sample) correlation plots.
E. The gene-read count matrix is passed to Primary Analysis (one tab only) where differential expression (DE) is performed using at least one of DESeq2, ImpulseDE2, and Next maSigPro, depending on how the user answered the adaptive default questions. The DE results between methods can be visualized using a Venn diagram. The user can also compare results with an outside gene list (e.g. from a past study) (**Figure 1B**). The results of each DE method are presented in the bottom left. The user can toggle to each method and TIMEOR will generate the associated gene trajectory clustermap. TIMEOR automatically chooses the number of clusters in the DE results. The number can also be chosen manually. Once the user chooses which DE method's result to use, results are passed to Secondary Analysis (**Figure S2**).

PRE-PROCESSING

When you visit TIMEOR, you land on the Pre-processing Stage first tab "Getting Started" with important and helpful information about how to run TIMEOR effectively. TIMEOR can take as input either 1) raw time-series RNA-seq .fastq files from SRA or GEO and that experiment's associated SRA Run Table (both gathered from GEO), or 2) a count matrix and its associated metadata file. We strongly recommend the user to upload a read-count matrix to the webserver as far as possible, given that computational resources are limited to 10GB and also might result in long wait times. Specifically, for processing large raw time-series RNA-seq .fastq files, we advise the user to the ready-to-use either Docker environment on an external server with adequate compute power and process the raw .fastq files using a local version of TIMEOR, which will create a read-count matrix. Then the user can complete their analysis within the same instance or the user can upload the generated read-count matrix to the TIMEOR website to continue the rest of the analysis at a later date. In what follows we describe each step of TIMEOR.

On the Pre-process Stage second tab "Set Inputs and Defaults, Process Raw Data" tab, first the user must answer six questions about their time-series data, and input data details. Here we provide a couple points for now the user might choose to answer each of these six questions. These points are also available on the TIMEOR website and associated documentation. Overall the user is first prompted to read the "Getting Started" tab, followed by inputting details about their experiment (answering questions and uploading data details). When answering the six questions about the user's experiment, they must select at least the organism, sequencing, and experiment type, then load metadata or SraRunTable.txt (if the user is processing raw time-series data). Here are a couple tips when answering these questions:

- **Question 1 asks**: "What type of organism?" The user can choose from fruit fly, human, or mouse.

- **Question 2 asks**: "What type of sequencing?" If the user is uploading a read count matrix, (strongly encouraged), the user can choose "not applicable".

- **Question 3 asks**: "What type of experiment?" There are two options - "case vs. control", and "just case or control" types of time-series that TIMEOR supports. This means:

- control at 1st time point vs case (i.e. treatment) at subsequent time points

- control **or** case over time

- **Question 4 asks**: "What type of time-series?" There are three options - "close time point and long time series", "close time point and short time series", and "distant time point". Based on the user's understanding of the biological system, the user should decide whether the time points are considered close or far in time. This question is important to determine how to model differential gene expression (DE) trajectories over time.

- DESeq2 is a categorical DE method generally used to analyze timepoints separately. When time points are far apart this is a good option. TIMEOR uses DESeq2 if the user toggles to "distant time point".

- When we are interested to model gene trajectories (when time points are close), we assess the temporal dynamic expression between time point t given $t-1$. In this context, it is advised a continuous DE method. TIMEOR uses ImpulseDE2 if the user toggles to "close time point ...". ImpulseDE2 employs an impulse model to determine differentially expressed genes.

- Importantly, it is **strongly** advised to compare all three (ImpulseDE2, Next maSigPro, and DESeq2) DE methods' results *by keeping 'Yes' for Question 5,* especially when there are "close time points and short time-series". Recent studies such as Spies et al. 2019 show that DESeq2 performs well when determining differentially expressed genes when time-series is short. To compare all three these, keep 'Yes' as the answer for Question 5 (below).

- **Question 5 asks**: "Compare multiple methods (alignment and differential expression)?" If this question is left to 'Yes' (which is **strongly** encouraged), TIMEOR will run all methods for the user to determine the best suited method. This is important because in many cases the categorical method DESeq2 which does not consider gene trajectories, still returns a robust set of differentially expressed genes. If this is set to 'No', TIMEOR will run for alignment (if applicable): HISAT2, and for DE: DESeq2 (if distant time points selected in Question 4), or ImpulseDE2 (if close time points selected in Question 4).

- **Question 6 asks**: "What is the maximum number of time steps over which one gene can influence the transcription of another gene?" This question prompts the user to tell TIMEOR the window of time over which one gene can *directly* influence another. Within this window all interactions are considered. It is advised to keep this value small if the time points are spaced out. Said differently, at each time point t for a differentially expressed gene g, if Question 6's answer were 2, TIMEOR would be asking, what are potential interactions of g with other TFs across $t+1$ and $t+2$.

Once these questions are answered, the user can either upload 1) an SraRunTable from GEO to be converted into a metadata file and matched with IDs to download and pre-process raw .fastq files, or 2) a metadata file and associated read count matrix, which is to be uploaded and visualized on the next tab “Process Count Matrix”. The SraRunTable should be checked to make sure it has the columns TIMEOR requires, and these specifications can be found on the first tab “Getting Started” of TIMEOR. Input specifications can be found on the “Getting Started” tab of TIMEOR, and in the “Tutorials” section on the left side-bar of TIMEOR.

Pre-Processing - Process Raw Data

Next on the same Pre-process Stage second tab “Set Inputs and Defaults, Process Raw Data” tab, if the user is interested to process raw time-series RNA-seq data, the user is directed to the Gene Expression Omnibus (GEO) to find their desired time series RNA-seq data (**Figure S1A**). The user is instructed to upload the GEO generated data information file (SraRunTable.txt), which TIMEOR parses into a metadata file and generates the user’s personal analysis session folder for use in subsequent analysis. The user is instructed to answer six questions about their data. Namely, 1) “what type of organism (*Drosophila melanogaster, Homo sapiens,* or *Mus musculus*)”, 2) “what type of sequencing (paired-end or single-end)”, 3) “what type of experiment (case vs. control, or just case or control)”, and 4) “what type of time series (close time point and long time series, distant time point, and close time point and short time series)”. In addition, TIMEOR can compare two methods for multiple alignment (HISAT2 (67) and Bowtie2 (68)) (**Figure 1A, S1B, S3A, S3B**) and several methods for differential expression (DE, DESeq2 (34), ImpulseDE2 (9), and Next maSigPro (12) (**Figure 1B, S1E, S3C**), should the user choose ‘yes’ to 5) “compare multiple methods (that is, both alignment and DE methods)”. Lastly TIMEOR infers direct interactions

between genes up to the maximum number of time steps input by the user by asking: 6) “What is the maximum number of time steps over which one gene can influence the transcription of another gene?”.

Once the user clicks “Run”, TIMEOR begins processing the answers to the six questions to choose the most appropriate adaptive default methods to run, followed by retrieving the raw .fastq data supplied in the uploaded GEO file. The retrieved .fastq files are then run through quality control using FastQC (63) and summarized using MultiQC (64). TIMEOR outputs the results in a summary table and interactive results for download. Next, each .fastq file (i.e. sample) is aligned to the reference genome for the organism designated in question 1. If the user chose to compare alignment methods, an alignment plot is output to the screen showing the overall alignment scores for each sample for both alignment methods (**Figure S1B**). If the user chose not to compare, HISAT2 (67) alignment results are plotted to show the uniquely mapped read percentage and overall alignment. These alignment results are then passed to HTSeq (66) to produce a read count matrix of samples by read counts per gene. Then for each sample and for each gene, a read count method is used to calculate the number of mapped reads to each gene, resulting in a vector of transcript counts per gene for each sample. All read count vectors for all samples are then merged to form one large transcript count matrix of genes by samples. TIMEOR indicates that each of the aforementioned steps is complete with a check mark (**Figure S1B**).

Pre-Processing - Load Count Matrix, Normalize and Correct Data

The read count matrix is then passed to the next tab "Load Count Matrix" (**Figure S1C**) where filtering and principal component analysis (PCA) are performed. Importantly, the user can also begin on this tab by simply loading the pre-computed read count matrix and associated metadata file. TIMEOR visualises PCA in an interactive plot, along with a loadings plot below it. TIMEOR performs replicate (or sample) correlations using either the Spearman or Pearson correlation, which are displayed in an interactive symmetrical heatmap (**Figure S1C**).

This read count matrix can be normalized and corrected on the next tab "Normalize and Correct Data" (**Figure S1D**). The user can choose between two normalization methods: trimmed mean of M-values, and upper quartile, and correction is performed by Harman (37). Once the user clicks "Run", TIMEOR outputs the resulting PCA plots of the normalized and corrected read count data. The user can again perform replicate (or sample) correlations on the normalized and corrected data.

PRIMARY ANALYSIS

In the Primary Analysis stage, the user is asked to name the results folder and set up the adjusted p-value threshold (**Figure S1E**). If the user selected to compare multiple differential expression methods (as part of the six initial questions to set the adaptive default methods) and the data are close time series, TIMEOR will run DESeq2 (34), ImpulseDE2 (9), and Next maSigPro (12). If the user selected that these timepoints are distant, DESeq2 (34) will be run using the gene-by-sample read count matrix. Note that TIMEOR inputs the raw read count matrix for DESeq2 (34) and ImpulseDE2 (9), as they require non-normalized and corrected read count data and the pre-normalized and corrected data for Next maSigPro (12). When the user clicks “Go” the method(s) are run. Once finished, the user can compare the overlap of differentially expressed genes between methods in a Venn diagram (**Figure S1E**). The user can also compare these results with a list of genes (from a past study for example). TIMEOR outputs the results for each method in a table that can be selected using the dropdown menu. When a method is selected, TIMEOR produces the associated gene trajectory clustermap on the right (**Figure S1E**).

The clustermap (25) interactively highlights clusters of differentially expressed genes’ expression trajectories in the form of a log2 fold change or z-score, formed using Euclidean distance and Ward.D2 (72) hierarchical clustering. Importantly, TIMEOR addresses the challenge of clustering by taking the mode of 3 unsupervised clustering methods (partition around medoids (73), Silhouette (74), and Calinski criterion (75)) to

automatically return the number of gene trajectory clusters to the user. Given that clustering is an NP-hard problem (76), TIMEOR also provides an Elbow plot (**Figure S3D**) to aid the user in choosing the number of clusters manually. Once the user determines which DE method to use, those results are passed to Secondary Analysis to determine temporal relations between genes and TFs.

SECONDARY ANALYSIS

TIMEOR performs three types of Secondary Analysis, all in separate tabs. Namely "Enrichment" (**Figure S2A**), "Factor Binding" (**Figure S2B**), and "Temporal Relations" (**Figure S2D, S2E**) to uncover both what processes are enriched (i.e. affected) in the user's time series experiment, and how those genes are regulating that enriched process. In the following sections we describe the details of each tab separately.

Secondary Analysis - Enrichment

Figure S2. TIMEOR Application Secondary Analysis.

A. In the Secondary Analysis Stage "Enrichment" tab, each gene trajectory is assessed for gene ontology, pathway, network, and de novo motif enrichment.

B. TIMEOR scans for the observed and top four predicted transcription factors (TFs) to bind to each gene trajectory cluster (table top right) from the *Top Predicted Transcription Factor Binding by Orthology* table. TIMEOR returns ENCODE IDs where ChIP-seq data exist for each TF. The user can then choose which TFs to plot over each gene trajectory cluster to help validate key regulators.

C. TIMEOR displays four predicted TF tables: 1) *Top Predicted Transcription Factors by Orthology* table, giving information regarding: the observed, predicted TFs by orthology, and associated ENCODE IDs (not shown), which is built from a consensus between multiple methods' results, visible in 2) *Top Predicted Transcription Factors by Orthology per Method* table consisting of the top predicted TFs for each individual method by orthology. If there is a consensus among methods in table 2 for any of the top four TFs above 40%, those results are displayed in table 1. 3) *Top Predicted Transcription Factors by Motif Similarity* table, giving information regarding: the observed,

predicted TFs by motif similarity, and associated ENCODE IDs (not shown), which is built from a consensus between multiple methods' results, visible in 4) *Top Predicted Transcription Factors by Motif Similarity per Method* table, consisting of the top predicted TFs for each individual method. If there is a consensus among methods in table 3 for any of the top four TFs above 40%, those results are displayed in table 4.
D. TIMEOR infers the temporal relations between the observed and predicted transcription factors.
E. The user can move nodes in the desired order using the STRINGdb interface.
F. Each session's results are clearly organized by stage and step in the user's personal analysis session folder to download for future use on the left side of each the website. The user can also download the log file for the session.
G. TIMEOR provides two tutorials, one for the web-app and one for the command line version of TIMEOR.

The resulting gene trajectory clusters are passed to the "Enrichment" tab in the "Secondary Analysis" stage. The user can toggle through the clusters to see the gene trajectory cluster (i.e. gene set) they wish to analyze (**Figure S2A**). Once the user clicks on the analyze button, the following enrichment analyses are performed: gene ontology (GO), pathway, network, and *de novo* motif. For GO analysis, ClusterProfiler (69) is run with a Benjamini Hochberg adjusted p-value cutoff of 0.05. If particular GO terms (molecular function, biological process, and cellular component) are significantly enriched for that group of genes, those results are depicted as "dot plots" under the clustermap on the "Enrichment" tab. *Dot plots* highlight the enriched GO terms (x-axis) vs. the ratio of the genes within the cluster that are enriched for that GO term (y-axis). The radius of the dot shows the count of genes enriched for that GO term and the color of the dot indicates how significant the GO term is.

TIMEOR stores other GO result formats including *ontology plots*, where the leaves of the tree are the most specific enriched terms, and a *relationship graph*, highlighting which GO terms are related to each other. The radius of each node shows the count of genes enriched for that GO term and the color of the dot indicates how significant the GO term is. TIMEOR also outputs spreadsheets of all results into the user's personal analysis session folder **(Figure S2F**).

Next, pathway level analysis is performed on each selected gene trajectory cluster using PathView (38), where an enriched pathway (Benjamini Hochberg adjusted p-value ≤ 0.05) is output with temporal highlighting for genes (-1 is most downregulated and 1 is most upregulated).

TIMEOR then uses STRINGdb (35) to search for enriched protein-protein interaction networks, if the enrichment for that group of genes is below a q-value ≤ 0.05. STRINGdb highlights various types of known interactions between genes, namely 'experimentally determined' and from 'curated databases'. StringDB also highlights predicted edges from 'gene neighborhoods', 'gene fusion', and 'gene co-occurrence events'. Lastly STRINGdb provides 'text-mining', 'co-expression', and 'protein-homology' interaction edge types. TIMEOR also stores a table of these interactions, along with the PubMed IDs that support 'experimentally determined' edges in the user's personal analysis session folder for downloading (**Figure S2F**).

Lastly, TIMEOR uses MEME (39) to discover novel (i.e. *de novo*), ungapped motifs in the sequences within each gene trajectory cluster, and returns the top three motifs. The positive strand motifs are shown, along with MEME's downloadable interactive results. TIMEOR stores the reverse complement motifs, and each gene's DNA sequence files in the user's personal analysis session folder.

Secondary Analysis - Factor Binding

On the next tab "Factor Binding" within Secondary Analysis, TIMEOR scans across all clusters to identify both observed and predicted TFs (**Figure S2B**), and list any ENCODE (43; 44) IDs associated with predicted TF binding data. TIMEOR displays the following four predicted TF tables: 1) *Top Predicted Transcription Factors by*

Orthology table, consisting of three pieces of information: a. observed TFs, predicted TFs by orthology, and c. associated ENCODE IDs. 2) *Top Predicted Transcription Factors by Orthology per Method* table lists the four top predicted TFs by orthology for each individual method. 3) *Top Predicted Transcription Factors by Motif Similarity* table lists three pieces of information: a. observed TFs, b. predicted TFs by motif similarity, and c. associated ENCODE IDs. 4) *Top Predicted Transcription Factors by Motif Similarity per Method* table lists the four top predicted TFs by motif similarity for each individual method (**Figure S2C**). In what follows we describe how each table is made.

TIMEOR performs this TF prediction using RcisTarget (41) to first identify TF binding motifs which are over-represented (i.e. enriched) in a gene list. RcisTarget uses a database (**Resource Table, Deposited Data**) containing genome-wide motif rankings, collected through multiple methods including HOCOMOCO (77), Transfac (78), Jaspar (52), CIS-BP (79), HOMER (53), and several labs' datasets including Dr. Jussi Taipale (40). RcisTarget performs motif-enrichment analysis on a gene list by first estimating the over-representation of each motif on the gene set. RcisTarget displays the resulting selection of significant motifs in ranked order by the Normalized Enrichment Score (NES). This is calculated for each motif based on the area under the curve distribution of all the motifs for the gene-set. Those motifs that pass the given threshold (3.0 by default by changeable by user) are considered significant.

Rcistarget's results are mostly long lists of redundant significantly enriched motifs and associated candidate TFs that are derived from multiple databases. TIMEOR prioritizes this long list by first forming a consensus about the top four (default but changeable by user) candidate TFs which bind to the input group of genes. This consensus is formed in three steps. First, enriched motif search is a widely studied problem, and TIMEOR leverages this fact to scan across all motif databases for which TF binding motifs are enriched. TIMEOR does this by first splitting the RcisTarget ranked list results into smaller ranked lists, one for each database used to identify enriched motifs. If at least 40% (default but changeable by user in command line version of TIMEOR) of the databases agree on their top ranked TF, then ENCODExplorer (44) is used to scan for any ChIP-seq data that exists for this TF. This procedure continues for the second, third, and fourth candidate TFs. TIMEOR repeats the run of RcisTarget and ENCODExplorer on each cluster to identify its recommendation for at most the top four candidate TFs. This process is repeated to find two types of candidate TFs.

Each enriched motif is associated with candidate TFs of two kinds, and are output into two tables: 1) based on orthologous sequences and placed in *Top Predicted Transcription Factors by Orthology* table, and 2) based on similarities between annotated and unknown motifs and placed in *Top Predicted Transcription Factors by Motif Similarity* table (40; 42). TIMEOR then uses ENCODExplorer to find associated factor binding data from ENCODE where possible. The user can also see the associated *Top Predicted Transcription Factors for Method* table, consisting of the top predicted TFs for each individual method for both orthology and motif similarity (**Figure S2C**).

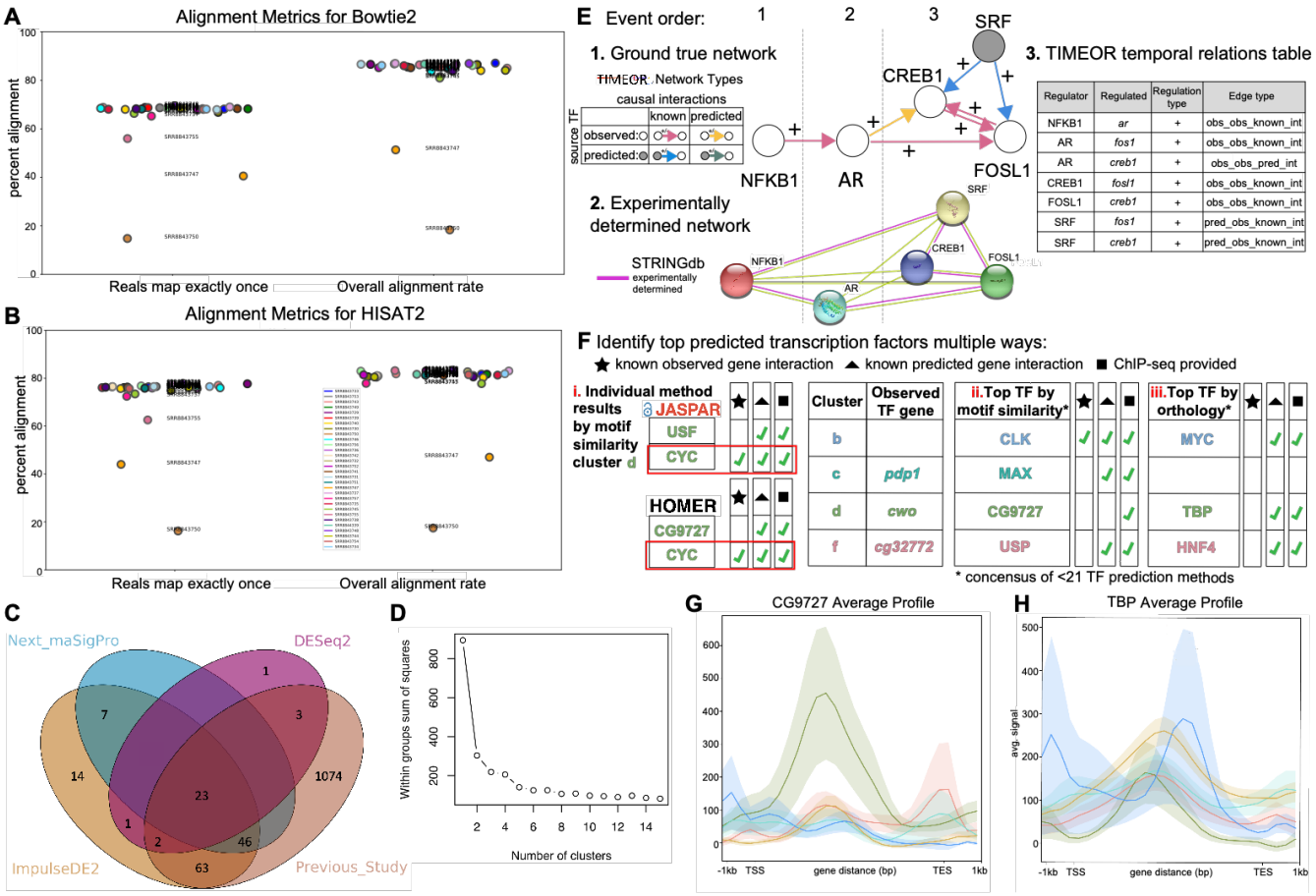

Figure S3. TIMEOR's visual outputs assist the user to make analysis decisions.

A. Alignment metrics from Bowtie2 for data from Zirin et al. 2019.

B. Alignment metrics from HISAT2 for data from Zirin et al. 2019.

C. TIMEOR produces a Venn diagram comparing categorical (DESeq2) and continuous (ImpulseDE2, Next maSigPro) differential expression methods and the previous study (Zirin). ImpulseDE2 shows the highest percent overlap with the previous study and other methods.

D. TIMEOR uses multiple methods to assess stability of the number of clusters. The Elbow method showed decreasing error stabilized at six clusters.

E. Simulation design, 1. The simulated ground truth network, 2. STRINGdb's TF network highlighting *known* (i.e. experimentally determined) interactions, and 3. TIMEOR's ground truth temporal relations table (gene regulatory network). Note the time column is removed here due to space constraints, and is visualized in the graphic.

F. Multiple predicted TF tables were used to help identify the top predicted TFs binding to each cluster from Zirin et al., 2019 data. In i. are the top transcription factors called by popular TF prediction methods highlighting that CYC has known observed and predicted TF interactions and ChIP-seq data to validate interactions. In ii. are the top transcription factors by orthology (table titled *Top Predicted Transcription Factors by Orthology*), and in iii. are the top transcription factors by motif similarity (table titled *Top Predicted Transcription Factors by Motif Similarity*). Each table also provides any ENCODE identifiers for ChIP-seq data if applicable (43; 44).

G. TBP average ChIP-seq profile over the gene body ±1KB for all six clusters.

H. CG9727 average ChIP-seq profile over the gene body ±1KB for all six clusters.

TIMEOR then prompts the user to upload their own or processed (using the ENCODE IDs provided by TIMEOR) TF binding data in the form of a .bigWig file, to see how the candidate TF(s) bind to genes within each gene trajectory cluster. DeepTools (61) is used to generate an average profile and heatmap over the gene body

±1kb from the transcription start and end site (**Figure S2B**). TIMEOR can process three TFs at a time through the web interface. All results are stored in the user's personal analysis session folder.

Secondary Analysis - Temporal Relations

Please see the main text and **Figure 2** where we describe the details of how TIMEOR forms the TF GRN.

METHOD AND QUESTION CHOICE ASSISTANCE

Below are several tips when determining the choice of method for various parts of the Pre-processing, Primary, and Secondary Analyses Stages. These can also be found on the "Getting Started" tab of TIMEOR's Pre-processing Stage.

- **"Normalize and Correct" tab**: there are two normalization options - upper quartile and trimmed mean of M-values. It is advised to try both methods through TIMEOR's interactive interface because the influence of normalization differs depending on the RNA-seq data structure.
    - There are several recent papers that discuss these differences such as Zyprych-Walczak et al. 2015, Pereira et al. 2018, and Abbas-Aghababazadeh et al. 2018.
- **"Normalize and Correct" tab**: there are two options for correlating samples/replicates using the Pearson or Spearman correlation. The choice of correlation method depends heavily on the assumptions the user wants to make about their data, and it is encouraged to try both in TIMEOR's interface. The user knows more about which samples/replicates (e.g. time points) should cluster together and how to identify outliers.
    - Both correlation methods define the strength of the relationship between the samples/replicates. The Pearson correlation accounts for differences in the samples/replicates mean and standard deviation when defining the linear relationship. The Spearman correlation is actually a nonparametric measure that uses the rank values of the samples/replicates.
    - Importantly, the more similar the expression profiles between samples/replicates, the higher the correlation coefficient will be.
    - Furthermore, the user is encouraged to remove any outliers (if needed) for further analysis.
- **"Primary Analysis" stage**: the user can choose to allow TIMEOR to automatically cluster the DE gene trajectories, or the user can choose the number of gene trajectory clusters. Importantly, finding the optimal solution to this hierarchical clustering problem is an NP-hard. Thus, user input is needed to assess a reasonable number of clusters for downstream analysis. To help, TIMEOR provides an automatic clustering option (PDF visible when folder is downloaded) which takes the mode between three unsupervised clustering methods (partition around medoids (73), Silhouette (74), and Calinski criterion (75)) to automatically return the number of gene trajectory clusters to the user. TIMEOR also provides an Elbow plot to show the user how the explained variation changes as a function of the number of clusters. The user can leverage this plot by picking the elbow of the curve. The user is encouraged to use the interactive clustermap and the clustering plots (available on download) to determine whether the automatic clustering option provides suitable clusters.
- **"Primary Analysis" stage**: NOTE, there is not a fold change cut-off for the DE gene trajectories, only an adjusted p-value cutoff. This allows the user to view significant differences in expression trajectories while the fold change might be smaller. This is useful to observe changes for genes including non-coding genes and genes involved in dosage compensation.
- **"Secondary Analysis: Factor Binding" tab**: the user is encouraged to "see each method's predicted transcription factors" and search for protein-DNA data (in .bigWig format) to view the binding profile of that transcription factor across each gene trajectory cluster.

- **"Secondary Analysis: Temporal Relations" tab**: the user can add additional genes or transcription factors (potentially viewed on Factor Binding tab) to the final gene regulatory network (GRN) within STRINGdb. NOTE: TIMEOR only reports the TF GRN using the observed and top one predicted TFs from the "Observed and Top Predicted Transcription Factors" table. The user is **encouraged** to view the results from individual methods (on Factor Binding tab) when constructing the final GRN, and view Temporal Relations Table to uncover the lead and lag relationships between TFs.

Current Stand-Alone GRN Inference Methods: do not fully leverage available data

To predict the order of action of transcriptional regulators, there are several stand-alone methods which infer GRNs given processed data and selected genes, regardless of experimental design (80). Specifically, they address two main questions: 1) What genes are relevant (or altered) and exhibit dynamic expression changes?, 2) How are these genes within the GRN regulating each other? However, most of these stand-alone GRN inference methods only consider RNA-seq data from a single time point, i.e. at steady-state, (81; 82; 83; 84; 85; 86; 87; 88; 89; 90; 91). While fewer methods consider consecutive time points (92; 93; 94; 95; 96). Importantly, several consecutive time point GRN inference methods are limited in focus, constructing GRNs only when a gene of interest is present such as TSNI (93) and may inaccurately interpret the network because they do not consider the experimental design. Thus, there is an urgent need for a pipeline method to leverage time-series, known experimental design, gene interactions, and multi-omics data to provide evidence of direct and epistatic interaction between genes, focusing on the master regulators of transcription, TFs, to reconstruct interpretable and high-fidelity GRNs in a user-guided fashion.

RESULT DETAILS

Simulations

Polyester version 1.24.0 (45) was used to simulate four RNA-seq expression cascading activation patterns over six time points two biological replicates for 63 genes at approximately 20X sequencing coverage **(Figure S3E)**. Note that the first time point was used as the control. Those fold change expression cascades were trajectory 1: 0.5,2,2,3,4,4.5; 2: 0.2,0.5,2,4,4.5,5; 3: 0.1,0.5,0.8,0.9,1,3; and 4: 0.05,0.1,0.1,0.5,0.5,4. To obtain an accurate baseline gene expression across the five timepoints, we analyzed a five time point RNA-seq experiment of fusobacterium nucleatum-stimulated human gingival fibroblasts control samples (GEO: GSE118691) taken from two donors (1 and 2) at 2, 6, 12, 24, 48 hours (46; 47). We simulated a constant RNA-seq expression of one fold across all five timepoints for all those genes that were active in at least one time point. This culminated in simulating expression for 17723 genes of which four are observed TFs and one TF is predicted. That is, the fourth expression trajectory activated last was enriched for the predicted TF SRF to bind by 22 genes (97).

To simulate the temporal relationships between these five TFs, TIMEOR enables the user to choose the window of possible interaction between TFs along the time course, which we set to be one time point. In this window, **Figure S3E** step 1 highlights our ground truth simulation of four observed TFs (one for each activation cascade trajectory) and one predicted TF. Specifically, at the first time point, we simulated initial activation of AR by NFKB1 which is a *known* (i.e. experimentally determined) interaction (**Figure S3E**). At the second time point we simulated AR activating FOSL1 (known interaction) and CREB1 (predicted interaction). At the third time point, we simulated SRF to activate the two observed genes (FOSL1 and CREB1) in known interactions. At this same time point, we simulated FOSL1 and CREB1 to activate each other in known interactions.

This simulation can be described in nine steps. First, as input, Polyester takes annotated transcript nucleotide sequences in the form of cDNA sequences in FASTA format. We used bioMart

(https://m.ensembl.org/biomart/martview/) to extract the first transcript sequence for each of 63 genes. Second, TIMEOR ran Bowtie2 version 2.3.5 (68) to align all sample .fasta files to the human genome (GRCh38):

(bowtie2 -f -p 3 -x /genomes_info/hsa/genome_bowtie2/genome -U sample_t01.fasta --un-gz out_un.sam.gz --al-gz out_al.sam.gz --met-file out_met-file.tsv -S out.sam 2> summaryfile.txt),

and achieved an average 83.9% alignment rate. Third, samtools version 1.9 (61) then converted and sorted all .sam files to .bam files:

(samtools view -S -b out.sam > out.bam; rm -rf out.sam out.sorted.sam; samtools sort out.bam -o out.sorted.bam; rm -rf out.bam).

Fourth, TIMEOR ran HTSeq to produce a read count matrix of samples by read counts per gene for each sample, and TIMEOR then merged the samples in chronological order to create the final read count matrix of genes by sample (i.e. replicates).

(htseq-count -f bam -r pos -i gene_id out.sorted.bam BDGP6/genes.gtf > htseq_counts).

Fifth, TIMEOR ran ImpulseDE2 version 1.10.0 (9) at an adjusted p-value threshold of 0.05 and found 65 gene DE. Neither of the additional genes were TFs and thus not selected for downstream TF temporal regulation analysis. Sixth, TIMEOR clustered the gene trajectories for all gene DE into 4 clusters. Seventh, for each cluster TIMEOR identified and summarized the top four (default by changeable by user) predicted TFs across at most 21 TF prediction methods when toggling the percent concordance at various thresholds (2, 5, 15, 25, 35, 45, 55, 65, 75, and 85 percent), and reported the percent concordance among these methods' ranks for the top four predicted TFs. Note that TIMEOR uses only the methods that output a TF when calculating percent concordance. Eighth, TIMEOR infers the temporal relationships between observed and top one predicted TFs for each cluster. This process of calling the top predicted TFs and inferring the temporal relations happened 10 times by toggling this percent concordance parameter between 2% - 85%.

Ninth, we calculated recall and precision to assess TIMEOR's robustness to recover the true underlying (i.e. ground truth) gene regulatory network (GRN) after determining the TF temporal relationships. We defined the quadruple (source TF, target TF, regulation type, interaction type). *Interaction types* are: "predicted to observed known interaction", "predicted to observed predicted interaction", "observed to observed known interaction", or "observed to observed predicted interaction". *Regulation types* are "activation" or "repression". A true positive consists of all correct quadruples. A false negative is failing to report one of the ground truth quadruples. And a false positive consists of introducing a new quadruple. TIMEOR recovered the true network with perfect recall at all percent concordance thresholds except at high percent concordance (65% and above) which lead to just one ground truth TF to drop out. As expected, at low percent concordance between TF prediction methods TIMEOR predicted other TF interactions which were not simulated (i.e. not part of ground truth). At a concordance threshold of at least 35 and above we obtained perfect precision (**Figure 3A**).

Real Data Processing

TIMEOR uncovered the complex molecular functions after insulin stimulation in *Drosophila* S2R+ cell (project ID SRP190499, **Supplementary Data 1**) by integrating both temporal RNA-seq and ChIP-seq data (**Supplementary Data 2, Figure 5**). Through TIMEOR, the steps described next were followed. The data were gathered from NCBI project SRP190499 using fastq-dump (61). Note only 2 of 30 samples shown. The specific command used was:

(*parallel -j $1 --bar fastq-dump {1} -split-files -O $2 -gzip ::: SRR8843729 SRR8843730*).

The sequenced reads were run through FastQC (63) with default parameters to check the quality of raw sequence data and filter out any sequences flagged for poor quality. No sequences were flagged. The specific command used was:

(*fastqc SRR.fastq.gz --outdir=/fastQC_results/*)

Sequenced reads were then mapped to release 6 *Drosophila melanogaster* genome (dm6) using Bowtie2 (68) and HISAT2 (67) (**Figure S3A, S3B**). Bowtie2 was used to align reads to the dm6 genome with this command:

(bowtie2 -p 3 -x /genomes_info/dme/genome_bowtie2/genome -U SRR.fastq.gz --un-gz out_un.sam.gz --al-gz out_al.sam.gz --met-file out_met-file.tsv -S out.sam 2> summaryfile.txt)

HISAT2 was used to align reads to the dm6 genome, followed by sorting and converting to a bam file:

(*hisat2 -p 3 --dta -x /genomes_info/dme/genome_hisat2/genome -U SRR.fastq.gz -t --un-gz out_unalign.sam.gz --al-gz out_al_atleastonce.sam.gz --known-splicesite-infile /genomes_info/dme/splicesites.txt --novel-splicesite-outfile novel_splicesite --summary-file summaryfile.txt --met-file met-file.txt -S out.sam 2> alignmentsummary.txt*).

HISAT2 was chosen due to higher unique read mapping (**Figure S3A, S3B**) and being splice-site aware. All sample .sam files were sorted and converted to .bam files:

(samtools view -S -b out.sam > out.bam; rm -rf out.sam out.sorted.sam; samtools sort out.bam -o out.sorted.bam; rm -rf out.bam).

TIMEOR plots the percent of reads mapping exactly once and overall alignment rate, which highlights that replicate SRR8843750_1.fastq.gz (timepoint 40 minutes) and replicate SRR8843747_1.fastq.gz (timepoint 180 minutes) show low alignment (**Figure S3A, S3B**). Due to contamination at the experimental level, replicate SRR8843750_1.fastq.gz was removed from further analysis. Next all replicates were converted to sorted .bam files (61) to produce read counts per gene using HTSeq (66). Note that htseq assumes that .bam is sorted by name by default and not position, so -r pos is necessary if samtools sort is used without the -n flag. TIMEOR ran HTSeq to produce a read count matrix of samples by read counts per gene for each sample, and TIMEOR then merged the samples in chronological order to create the final read count matrix of genes by sample (i.e. replicates). Specifically, the used command for HTSeq is:

(htseq-count -f bam -r pos -i gene_id out.sorted.bam BDGP6/genes.gtf > htseq_counts).

Removing contamination: Note that replicate SRR8843750 shows an overall mapping rate of 17.44%. Further, SRR8843747 shows an overall mapping rate of 47.02%. When we use BLAST (both blastn and megablast, 98) to identify several sequences from replicate SRR8843750 we see synthetic construct external RNA control and Methanocaldococcus jannaschii as the top subject sequence. When we use BLAST to identify several sequences from replicate SRR8843747 we see *Drosophila simulans* uncharacterized protein as the top subject sequence. In contrast, looking at several sequences from replicate SRR8843757 and SRR8843749 the top subject sequences are derived from *Drosophila melanogaster*. Therefore, the SRR8843750 replicate was removed.

Normalization and correction: After filtering out rows where the mean read count is 5 or less across all replicates and samples, 8909/17558 genes remained. This normalization and correction was used for Next

maSigPro (12) as it requires pre-normalization and correction. See further explanation in section “Next maSigPro results”.

Differential expression: According to comparisons made by Spies et al., 2019 (8): all time course tools were outperformed by classical pairwise comparison approaches (in this case DESeq2, 34) on short time series (<8 time points) in terms of overall performance and robustness to noise, mostly because of high number of false positives, with the exception of ImpulseDE2 (9). On longer time series, pairwise approaches were performed less well than splineTimeR (11) and Next maSigPro, which did not identify any false-positive candidates. SplineTimeR does not support Zirin et al. 2019’s experimental design and thus was unable to be run. Other methods consider this dataset experimental set-up within their modeling framework.

There is significant overlap between the differentially expressed (DE) genes previously called by Zirin et al., 2019 and these three methods. ImpulseDE2 (9) showed the most significant set of differentially expressed genes overlapping with the previous study (26). This gene set was used downstream in TIMEOR to produce a clustermap of seven gene trajectory clusters. We recapitulate the findings of Zirin et al., 2019 by finding two clusters of differentially expressed genes (adjusted p-value < 0.05) enriched in the GO categories nucleolus and ribosome biogenesis, some of which were not identified previously. TIMEOR identified several additional clusters with novel characteristics.

Zirin et al., 2019 identified 1211 differentially expressed genes and followed-up to identify a subset of 33 differentially expressed genes involved in several biological processes including ribosome biogenesis and transcription (26). Converting those gene symbols to Flybase IDs gives 36 genes, as some gene names correspond to one of several Flybase IDs. Comparing the gene list of 36 Flybase IDs between all methods, we observe the most significant gene overlap with ImpulseDE2 (p-value < 5.518766e-08 using the hypergeometric test of significance) compared to Next maSigPro (p-value <0.0008382603) and DESeq2 (p-value <0.002140549). Comparing our results with the 1211 differentially expressed genes from Zirin et al.,TIMEOR finds that ImpulseDE2 showed the highest overlap with the list of 1211 differentially expressed genes (p-value = 5.33342e-127 using the hypergeometric test of significance) and the highest overlap with other methods (**Figure S3C**).

ImpulseDE2 results: ImpulseDE2 requires unnormalized and uncorrected counts and replicate data. At an adjusted p-value cutoff of 0.05, we found 156 significantly differentially expressed genes. As aforementioned in the previous paragraph, ImpulseDE2 shows the most significant gene overlap with the previous study when looking at the full set of 1211 differentially expressed genes (p-value = 5.33342e-127) and the follow-up subset of 36 differentially expressed genes (p-value < 5.518766e-08).

Next maSigPro results: Data must be normalized before the application of Next maSigPro because there is no integrated normalization method (12). Thus, the data are first filtered to remove any genes across all replicates with less than a mean of 5 reads. The data are then normalized by the mean ratio normalization and batch effect corrected with Combat (99). Considering all case conditions with 3 batches, DESeq2 reported 76 differentially expressed genes. When comparing with the full set of 1211 differentially expressed genes there is a less significant overlap (p-value = 3.748126e-69) than with ImpulseDE2. The overlap between these 76 genes and the 156 from ImpulseDE2 is 76 genes. Furthermore, when comparing with the follow-up subset of 36 differentially expressed genes, Next maSigPro reports a less significant overlap with Zirin et al., 2019 than ImpulseDE2, with 3 overlapping genes, resulting in a p-value of 0.0006961443.

DESeq2 results: DESeq2 requires unnormalized and uncorrected counts with replicate data. Given that this is a categorical differential expression method, we must test for any differences over multiple time points all compared to time point 0 (i.e. the reference). Here we use a design including the factor of time, and use the likelihood ratio test to see if there are significant changes in expression for genes between T0 and the each

additional time point. Considering all case conditions with 3 batches, we obtain 30 differentially expressed genes. When comparing with the full set of 1211 differentially expressed genes there is the least significant overlap (p-value = 2.010729e-29) with Zirin et al., 2019. Furthermore, when comparing with the follow-up subset of 36 differentially expressed genes, DESeq2 reports the least significant overlap with 3 overlapping genes, resulting in a p-value of 4.322579e-05.

TIMEOR clustermap of z-scored gene expression trajectories at an adjusted p-value<0.05: **Figure 3B and 4A** show the clustermap of DEG expression as z-scores, which correspond to the observed read counts of a sample rescaled to a standard normal distribution (with mean 0 and standard deviation 1). Specifically, the z-score of an observation is the mean divided by the standard deviation of all observations in the sample. Thus, the z-score describes how many standard deviations an observation is from the mean of all observations, with positive values indicating upregulation of the gene and negative values indicating downregulation of a gene. These z-scored DEG trajectories were then clustered using Euclidean distance and Ward.D2 (72) hierarchical clustering. Note that z-scored expression values are generated for ImpulseDE2 and Next maSigPro. DESeq2 is a categorical method that outputs log2 fold change values across each timepoint. TIMEOR uses a variety of methods to assess stability of the number of clusters. In particular, the Elbow method showed that the decreasing error stabilized at six clusters (**Figure S3D**). Further human inspection showed a clear delineation of six clusters in the heatmap.

TIMEOR GO analysis: ClusterProfiler (69) is run to determine gene ontology (GO) enrichment of each cluster of genes (**Figure 3B and 4A**) within the categories of biological process, cellular component, and molecular function. Within each of our six clusters using a Benjamini Hochberg adjusted p-value cutoff of 0.05, TIMEOR found the most significant GO term for cluster d to be cellular response to endogenous stimulus within “biological process”. In cluster c, the nucleolus (cellular component), ribosome biogenesis (biological process) and catalytic activity acting on RNA (molecular function) are the most enriched terms. Within the largest cluster b, ribosome biogenesis (biological process), nucleolus (cellular component) and snoRNA binding (molecular function) are the most enriched terms. The molecular function enrichment for snoRNA binding is particularly interesting because in the next activated cluster e there is a significant number of snoRNAs present (**Figure 3E**). Our full TIMEOR results folder with various GO analysis visualizations can be found on Github.

TIMEOR pathway analysis: Pathview (38) is run to highlight genes and their trajectories (-1 is most downregulated and 1 is most upregulated) in any enriched pathways for each cluster of genes. **Figure 3D** shows the differentially expressed genes in cluster b which are part of the ribosome biogenesis pathway. This pathway was also identified in the previous study which generated these data from Zirin et al., 2019.

TIMEOR factor binding analysis: For each cluster, TIMEOR found both observed and predicted TFs via the *Top Predicted Transcription Factors by Orthology* table and *Top Predicted Transcription Factors by Motif Similarity* table, and each individual method’s results which are visible in their associated *Top Transcription Factors per Method Table* for both orthology motif similarity. In TIMEOR’s factor binding script, the threshold normalized enrichment score was set to 3, and the percent concordance threshold was set to 30. **Figure 4B** shows the prioritized list of TFs per cluster. TIMEOR’s provided list of ENCODE IDs (43; 44) enabled us to choose which factor data to upload and view via average profiles. See next section for those details.

TIMEOR average profile generation: For each top predicted transcription factor, we identified the most closely related ChIP-seq dataset (in .bigwig format) on ENCODE using the IDs provided by TIMEOR (**Supplementary Data 2**). Specifically, using data from Dr. Kevin White, we used 6-24 whole organism embryo ChIP-seq data for CWO (ENCFF491LHJ). We used 0-16 hours whole organism embryo ChIP-seq data for MYC (ENCFF829HXS). We used whole organism prepupa ChIP-seq data for CYC (ENCFF082KKV). And we used whole organism wandering third instar larva for HNF4 (ENCFF680FFM), TBP (ENCFF553PBY), and CG9727

(ENCFF145ATU). TIMEOR then uses deepTools (65) to plot the average peak distribution over each gene trajectory cluster, specifically using the set of commands below:

(computeMatrix scale-regions -S ${INPUT_BIGWIG} -R ${LIST_BEDS[@]} --binSize 250 --beforeRegionStartLength 1000 --afterRegionStartLength 1000 --regionBodyLength 5000 -o ${OUTPUT_DIR}/matrix.genes.clusters.mat.gz --skipZeros --smartLabels --sortRegions descend)

(plotHeatmap -m ${OUTPUT_DIR}/matrix.genes.clusters.mat.gz -out ${OUTPUT_DIR}/heatmap_genes.clusters.${TF}.png --heatmapHeight 25 --heatmapWidth 15 --labelRotation 45 --missingDataColor red --whatToShow="heatmap only")

(plotProfile -m ${OUTPUT_DIR}/matrix.genes.clusters.mat.gz -out ${OUTPUT_DIR}/avg_profile.${TF}.png --plotType=se --labelRotation 45 --plotHeight 15 --plotWidth 15 --colors ${COLORS})

TIMEOR temporal relations: Between all clusters, TIMEOR highlighted the temporal relationships between all observed and predicted TFs (those listed by default in the *Top Predicted Transcription Factors by Orthology* table). In TIMEOR's temporal relations script, the number of time steps over which one gene can influence the transcription of another gene (question six) was set to two. Within this time window, all possible directed interactions are considered and characterized. We used the last tab of Secondary Analysis (i.e. "Temporal Relations") to visualize and interpret the temporal relations table (encoding the GRN) in combination with the transcription factor network (35), the clustermap, and the list of predicted TFs to identify the main altered regulatory mechanisms. Overall, TIMEOR predicts a GRN in which first, predicted TF Hnf4 is predicted to repress observed gene *CG32772*. *CG32772* is then predicted to activate the observed gene *cwo*. Predicted TF TBP is also predicted to activate *cwo*. Due to TBP's poor binding profile with its predicted cluster and overall, we examined CYC (**Figure 4B, S3F**) and determined that it is likely activating *cwo* (**Figure 4D**). Observed TF CWO is then predicted to activate *pdp1* (**Figure S4**). These temporal relations are validated by multiple studies (27; 28; 29; 30). Supplementary Data are available at NAR online.

TUTORIAL FOR WEB SERVER

Please see website for "Getting Started" tutorial, "Overview" tutorial, a copy of the "Web Server" tutorial, and the "Command Line" tutorial. They are all available in that order.

Please see next page for TIMEOR's main tutorial: Web Server Tutorial. The other tutorials are easily accessible in at timeor.brown.edu "Getting Started" and clicking on "Tutorials" in the side-bar.

NOTE: the hyperlinks for the tutorial below are accessible online at timeor.brown.edu.

1. Chai, L. E., Loh, S. K., Low, S. T., Mohamad, M. S., Deris, S., & Zakaria, Z. (2014). A review on the computational approaches for gene regulatory network construction. Computers in biology and medicine, 48, 55-65.
2. Sahraeian, S.M.E., Mohiyuddin, M., Sebra, R., Tilgner, H., Afshar, P.T., Au, K.F., Bani Asadi, N., Gerstein, M.B., Wong, W.H., Snyder, M.P., et al. (2017). Gaining comprehensive biological insight into the transcriptome by performing a broad-spectrum RNA-seq analysis. Nat. Commun. *8*, 59.
3. Wang, Z., Gerstein, M., and Snyder, M. (2009). RNA-Seq: a revolutionary tool for transcriptomics. Nat. Rev. Genet. *10*, 57–63.
4. Skene, P.J., and Henikoff, S. (2017). An efficient targeted nuclease strategy for high-resolution mapping of DNA binding sites. Elife *6*.
5. Johnson, D.S., Mortazavi, A., Myers, R.M., and Wold, B. (2007). Genome-wide mapping of in vivo protein-DNA interactions. Science *316*, 1497–1502.

6. Brent, M.R. (2016). Past roadblocks and new opportunities in transcription factor network mapping. Trends Genet. *32*, 736–750.
7. Florez, J. C. (2017). Mining the genome for therapeutic targets. *Diabetes*, *66*(7), 1770-1778.
8. Spies, D., Renz, P.F., Beyer, T.A., and Ciaudo, C. (2019). Comparative analysis of differential gene expression tools for RNA sequencing time course data. Brief. Bioinformatics *20*, 288–298.
9. Fischer, D.S., Theis, F.J., and Yosef, N. (2018). Impulse model-based differential expression analysis of time course sequencing data. Nucleic Acids Res. *46*, e119.
10. Jensen, T.L., Frasketi, M., Conway, K., Villarroel, L., Hill, H., Krampis, K., and Goll, J.B. (2017). RSEQREP: RNA-Seq Reports, an open-source cloud-enabled framework for reproducible RNA-Seq data processing, analysis, and result reporting. [version 2; peer review: 2 approved]. F1000Res. *6*, 2162.
11. Michna, A., Braselmann, H., Selmansberger, M., Dietz, A., Hess, J., Gomolka, M., Hornhardt, S., Blüthgen, N., Zitzelsberger, H., and Unger, K. (2016). Natural Cubic Spline Regression Modeling Followed by Dynamic Network Reconstruction for the Identification of Radiation-Sensitivity Gene Association Networks from Time-Course Transcriptome Data. PLoS ONE *11*, e0160791.
12. Nueda, M.J., Tarazona, S., and Conesa, A. (2014). Next maSigPro: updating maSigPro bioconductor package for RNA-seq time-series. Bioinformatics *30*, 2598–2602.
13. Spurney, R.J., Van den Broeck, L., Clark, N.M., Fisher, A.P., de Luis Balaguer, M.A., and Sozzani, R. (2020). tuxnet: a simple interface to process RNA sequencing data and infer gene regulatory networks. Plant J. *101*, 716–730.
14. Wani, N., and Raza, K. (2019). Integrative approaches to reconstruct regulatory networks from multi-omics data: A review of state-of-the-art methods. Comput. Biol. Chem. *83*, 107120.
15. Spies, D., and Ciaudo, C. (2015). Dynamics in Transcriptomics: Advancements in RNA-seq Time Course and Downstream Analysis. Comput. Struct. Biotechnol. J. *13*, 469–477.
16. Barbosa, S., Niebel, B., Wolf, S., Mauch, K., and Takors, R. (2018). A guide to gene regulatory network inference for obtaining predictive solutions: Underlying assumptions and fundamental biological and data constraints. BioSystems *174*, 37–48.
17. Marbach, D., Costello, J.C., Küffner, R., Vega, N.M., Prill, R.J., Camacho, D.M., Allison, K.R., DREAM5 Consortium, Kellis, M., Collins, J.J., et al. (2012). Wisdom of crowds for robust gene network inference. Nat. Methods *9*, 796–804.
18. Torre, D., Lachmann, A., and Ma'ayan, A. (2018). BioJupies: Automated Generation of Interactive Notebooks for RNA-Seq Data Analysis in the Cloud. Cell Syst. *7*, 556-561.e3.
19. Cornwell, M., Vangala, M., Taing, L., Herbert, Z., Köster, J., Li, B., Sun, H., Li, T., Zhang, J., Qiu, X., et al. (2018). VIPER: Visualization Pipeline for RNA-seq, a Snakemake workflow for efficient and complete RNA-seq analysis. BMC Bioinformatics *19*, 135.
20. Kartashov, A.V., and Barski, A. (2015). BioWardrobe: an integrated platform for analysis of epigenomics and transcriptomics data. Genome Biol. *16*, 158.
21. Ge, S.X., Son, E.W., and Yao, R. (2018). iDEP: an integrated web application for differential expression and pathway analysis of RNA-Seq data. BMC Bioinformatics *19*, 534.
22. de Jong, A., van der Meulen, S., Kuipers, O.P., and Kok, J. (2015). T-REx: Transcriptome analysis webserver for RNA-seq Expression data. BMC Genomics *16*, 663.
23. Afgan, E., Baker, D., Batut, B., van den Beek, M., Bouvier, D., Cech, M., Chilton, J., Clements, D., Coraor, N., Grüning, B.A., et al. (2018). The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. Nucleic Acids Res. *46*, W537–W544.
24. Chang W., Cheng J., Allaire J.J., Xie Y., and McPherson, J. (2020). shiny: Web Application Framework for R. R package version 1.4.0.2. https://CRAN.R-project.org/package=shiny
25. Sievert C (2020). Interactive Web-Based Data Visualization with R, plotly, and shiny. Chapman and Hall/CRC. ISBN 9781138331457, https://plotly-r.com.
26. Zirin, J., Ni, X., Sack, L.M., Yang-Zhou, D., Hu, Y., Brathwaite, R., Bulyk, M.L., Elledge, S.J., and Perrimon, N. (2019). Interspecies analysis of MYC targets identifies tRNA synthetases as mediators of growth and

survival in MYC-overexpressing cells. Proc Natl Acad Sci USA *116*, 14614–14619.
27. Fathallah-Shaykh, H.M. (2010). Dynamics of the Drosophila circadian clock: theoretical anti-jitter network and controlled chaos. PLoS ONE *5*, e11207.
28. Kadener, S., Stoleru, D., McDonald, M., Nawathean, P., and Rosbash, M. (2007). Clockwork Orange is a transcriptional repressor and a new Drosophila circadian pacemaker component. Genes Dev. *21*, 1675–1686.
29. Maury, E. (2019). Off the clock: from circadian disruption to metabolic disease. Int. J. Mol. Sci. *20*.
30. Zhou, J., Yu, W., and Hardin, P.E. (2016). CLOCKWORK ORANGE Enhances PERIOD Mediated Rhythms in Transcriptional Repression by Antagonizing E-box Binding by CLOCK-CYCLE. PLoS Genet. *12*, e1006430.
31. Qian, J., and Scheer, F.A.J.L. (2016). Circadian system and glucose metabolism: implications for physiology and disease. Trends Endocrinol. Metab. *27*, 282–293.
32. Sharp, B., Paquet, E., Naef, F., Bafna, A., and Wijnen, H. (2017). A new promoter element associated with daily time keeping in Drosophila. Nucleic Acids Res. *45*, 6459–6470.
33. Stenvers, D.J., Scheer, F.A.J.L., Schrauwen, P., la Fleur, S.E., and Kalsbeek, A. (2019). Circadian clocks and insulin resistance. Nat. Rev. Endocrinol. *15*, 75–89.
34. Love, M.I., Huber, W., and Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biol. *15*, 550.
35. Franceschini, A., Szklarczyk, D., Frankild, S., Kuhn, M., Simonovic, M., Roth, A., Lin, J., Minguez, P., Bork, P., von Mering, C., et al. (2013). STRING v9.1: protein-protein interaction networks, with increased coverage and integration. Nucleic Acids Res. *41*, D808-15.
36. Edgar, R., Domrachev, M., and Lash, A.E. (2002). Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. Nucleic Acids Res. *30*, 207–210.
37. Oytam, Y., Sobhanmanesh, F., Duesing, K., Bowden, J. C., Osmond-McLeod, M., & Ross, J. (2016). Risk-conscious correction of batch effects: maximising information extraction from high-throughput genomic datasets. *BMC bioinformatics*, *17*(1), 332.
38. Luo, W., and Brouwer, C. (2013). Pathview: an R/Bioconductor package for pathway-based data integration and visualization. Bioinformatics *29*, 1830–1831.
39. Bailey, T.L., and Elkan, C. (1994). Fitting a mixture model by expectation maximization to discover motifs in biopolymers. Proc. Int. Conf. Intell. Syst. Mol. Biol. *2*, 28–36.
40. Aibar, S., González-Blas, C.B., Moerman, T., Huynh-Thu, V.A., Imrichova, H., Hulselmans, G., Rambow, F., Marine, J.-C., Geurts, P., Aerts, J., et al. (2017). SCENIC: single-cell regulatory network inference and clustering. Nat. Methods *14*, 1083–1086.
41. Hu, H., Miao, Y. R., Jia, L. H., Yu, Q. Y., Zhang, Q., & Guo, A. Y. (2019). AnimalTFDB 3.0: a comprehensive resource for annotation and prediction of animal transcription factors. *Nucleic acids research*, *47*(D1), D33-D38.
42. Verfaillie, A., Imrichová, H., Van de Sande, B., Standaert, L., Christiaens, V., Hulselmans, G., ... & Atak, Z. K. (2014). iRegulon: from a gene list to a gene regulatory network using large motif and track collections. PLoS Comput Biol, 10(7), e1003731.
43. Davis, C.A., Hitz, B.C., Sloan, C.A., Chan, E.T., Davidson, J.M., Gabdank, I., Hilton, J.A., Jain, K., Baymuradov, U.K., Narayanan, A.K., et al. (2018). The Encyclopedia of DNA elements (ENCODE): data portal update. Nucleic Acids Res. *46*, D794–D801.
44. Joly Beauparlant C., Lemacon A., Fournier E., Droit A. (2020). *ENCODExplorer: A compilation of ENCODE metadata*. R package version 2.14.0.
45. Frazee, A. C., Jaffe, A. E., Langmead, B., & Leek, J. T. (2015). Polyester: simulating RNA-seq datasets with differential transcript expression. *Bioinformatics*, *31*(17), 2778-2784.
46. Kang, W., Jia, Z., Tang, D., Zhao, X., Shi, J., Jia, Q., He, K., and Feng, Q. (2019a). Time-Course Transcriptome Analysis for Drug Repositioning in Fusobacterium nucleatum-Infected Human Gingival Fibroblasts. Front. Cell Dev. Biol. *7*, 204.

47. Kang, W., Jia, Z., Tang, D., Zhang, Z., Gao, H., He, K., and Feng, Q. (2019b). Fusobacterium nucleatum Facilitates Apoptosis, ROS Generation, and Inflammatory Cytokine Production by Activating AKT/MAPK and NF-κB Signaling Pathways in Human Gingival Fibroblasts. Oxid. Med. Cell. Longev. *2019*, 1681972.
48. Wang, Y.-H., and Huang, M.-L. (2009). Reduction of Lobe leads to TORC1 hypoactivation that induces ectopic Jak/STAT signaling to impair Drosophila eye development. Mech. Dev. *126*, 781–790.
49. Berk, A.J. (2016). Discovery of RNA splicing and genes in pieces. Proc Natl Acad Sci USA *113*, 801–805.
50. Sharp, P.A. (1991). “Five easy pieces”. Science *254*, 663.
51. Sharp, P.A. (2009). The centrality of RNA. Cell *136*, 577–580.
52. Sandelin, A., Alkema, W., Engström, P., Wasserman, W. W., & Lenhard, B. (2004). JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic acids research*, *32*(suppl_1), D91-D94.
53. Heinz, S., Benner, C., Spann, N., Bertolino, E., Lin, Y. C., Laslo, P., ... & Glass, C. K. (2010). Simple combinations of lineage-determining transcription factors prime cis-regulatory elements required for macrophage and B cell identities. *Molecular cell*, *38*(4), 576-589.
54. Tweedie, S., Ashburner, M., Falls, K., Leyland, P., McQuilton, P., Marygold, S., ... & FlyBase Consortium. (2009). FlyBase: enhancing Drosophila gene ontology annotations. *Nucleic acids research*, *37*(suppl_1), D555-D559.
55. Bolger, A. M., Lohse, M., & Usadel, B. (2014). Trimmomatic: A flexible trimmer for Illumina Sequence Data. *Bioinformatics*, btu170.
56. Almende B. V. Thieurmel B. & Robert T. (2019). visNetwork: Network Visualization using ‘vis. js’ Library. R package version 2.0. 8.
57. Boden, G., Ruiz, J., Urbain, J.L., and Chen, X. (1996). Evidence for a circadian rhythm of insulin secretion. Am. J. Physiol. *271*, E246-52.
58. James, S.M., Honn, K.A., Gaddameedhi, S., and Van Dongen, H.P.A. (2017). Shift Work: Disrupted Circadian Rhythms and Sleep-Implications for Health and Well-Being. Curr. Sleep Med. Rep. *3*, 104–112.
59. Barry, W.E., and Thummel, C.S. (2016). The Drosophila HNF4 nuclear receptor promotes glucose-stimulated insulin secretion and mitochondrial function in adults. Elife *5*.
60. Szklarczyk, D., Gable, A. L., Lyon, D., Junge, A., Wyder, S., Huerta-Cepas, J., Simonovic, M., Doncheva, N. T., Morris, J. H., Bork, P., Jensen, L. J., & Mering, C. V. (2019). STRING v11: protein-protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic acids research*, *47*(D1), D607–D613. https://doi.org/10.1093/nar/gky1131
61. Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., ... & Durbin, R. (2009). The sequence alignment/map format and SAMtools. *Bioinformatics*, *25*(16), 2078-2079.
62. Leinonen, R., Sugawara, H., Shumway, M., & International Nucleotide Sequence Database Collaboration. (2010). The sequence read archive. Nucleic acids research, 39(suppl_1), D19-D21.
63. Andrews, S. (2010). FastQC: a quality control tool for high throughput sequence data.
64. Ewels, P., Magnusson, M., Lundin, S., & Käller, M. (2016). MultiQC: summarize analysis results for multiple tools and samples in a single report. *Bioinformatics*, *32*(19), 3047-3048.
65. Ramírez, F., Ryan, D. P., Grüning, B., Bhardwaj, V., Kilpert, F., Richter, A. S., ... & Manke, T. (2016). deepTools2: a next generation web server for deep-sequencing data analysis. *Nucleic acids research*, *44*(W1), W160-W165.
66. Anders, S., Pyl, P. T., & Huber, W. (2015). HTSeq—a Python framework to work with high-throughput sequencing data. *Bioinformatics*, *31*(2), 166-169.
67. Kim, D., Paggi, J. M., Park, C., Bennett, C., & Salzberg, S. L. (2019). Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype. *Nature biotechnology*, *37*(8), 907-915.
68. Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. *Nature methods*, *9*(4), 357.
69. Yu, G., Wang, L. G., Han, Y., & He, Q. Y. (2012). clusterProfiler: an R package for comparing biological themes among gene clusters. *Omics: a journal of integrative biology*, *16*(5), 284-287.

70. Khan, A., & Mathelier, A. (2017). Intervene: a tool for intersection and visualization of multiple gene or genomic region sets. *BMC bioinformatics*, *18*(1), 287.
71. Kaufman, L., & Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis* (Vol. 344). John Wiley & Sons.
72. Murtagh, F., & Legendre, P. (2014). Ward's hierarchical agglomerative clustering method: which algorithms implement Ward's criterion?. *Journal of classification*, *31*(3), 274-295.
73. Reynolds, A. P., Richards, G., de la Iglesia, B., & Rayward-Smith, V. J. (2006). Clustering rules: a comparison of partitioning and hierarchical clustering algorithms. *Journal of Mathematical Modelling and Algorithms*, *5*(4), 475-504.
74. Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, *20*, 53-65.
75. Caliński, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, *3*(1), 1-27.
76. Křivánek, M., & Morávek, J. (1986). NP-hard problems in hierarchical-tree clustering. *Acta informatica*, *23*(3), 311-323.
77. Kulakovskiy, I. V., Vorontsov, I. E., Yevshin, I. S., Sharipov, R. N., Fedorova, A. D., Rumynskiy, E. I., ... & Kolpakov, F. A. (2018). HOCOMOCO: towards a complete collection of transcription factor binding models for human and mouse via large-scale ChIP-Seq analysis. *Nucleic acids research*, *46*(D1), D252-D259.
78. Wingender, E., Dietze, P., Karas, H., & Knüppel, R. (1996). TRANSFAC: a database on transcription factors and their DNA binding sites. *Nucleic acids research*, *24*(1), 238-241.
79. Weirauch, M. T., Yang, A., Albu, M., Cote, A. G., Montenegro-Montero, A., Drewe, P., ... & Zheng, H. (2014). Determination and inference of eukaryotic transcription factor sequence specificity. *Cell*, *158*(6), 1431-1443.
80. Mochida, K., Koda, S., Inoue, K., and Nishii, R. (2018). Statistical and machine learning approaches to predict gene regulatory networks from transcriptome data sets. Front. Plant Sci. *9*, 1770.
81. Langfelder, P., and Horvath, S. (2008). WGCNA: an R package for weighted correlation network analysis. BMC Bioinformatics *9*, 559.
82. Villaverde, A.F., Ross, J., Morán, F., and Banga, J.R. (2014). MIDER: network inference with mutual information distance and entropy reduction. PLoS ONE *9*, e96732.
83. Wang, Y.K., Hurley, D.G., Schnell, S., Print, C.G., and Crampin, E.J. (2013). Integration of steady-state and temporal gene expression data for the inference of gene regulatory networks. PLoS ONE *8*, e72103.
84. Zhang, L., Feng, X.K., Ng, Y.K., and Li, S.C. (2016). Reconstructing directed gene regulatory network by only gene expression data. BMC Genomics *17 Suppl 4*, 430.
85. Huynh-Thu, V.A., Irrthum, A., Wehenkel, L., and Geurts, P. (2010). Inferring regulatory networks from expression data using tree-based methods. PLoS ONE *5*.
86. Bar-Joseph, Z., Gerber, G.K., Lee, T.I., Rinaldi, N.J., Yoo, J.Y., Robert, F., Gordon, D.B., Fraenkel, E., Jaakkola, T.S., Young, R.A., et al. (2003). Computational discovery of gene modules and regulatory networks. Nat. Biotechnol. *21*, 1337–1342.
87. Faith, J.J., Hayete, B., Thaden, J.T., Mogno, I., Wierzbowski, J., Cottarel, G., Kasif, S., Collins, J.J., and Gardner, T.S. (2007). Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. PLoS Biol. *5*, e8.
88. Haury, A.-C., Mordelet, F., Vera-Licona, P., and Vert, J.-P. (2012). TIGRESS: Trustful Inference of Gene REgulation using Stability Selection. BMC Syst. Biol. *6*, 145.
89. Liu, W., Zhu, W., Liao, B., and Chen, X. (2016). Gene Regulatory Network Inferences Using a Maximum-Relevance and Maximum-Significance Strategy. PLoS ONE *11*, e0166115.
90. Margolin, A.A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Dalla Favera, R., and Califano, A. (2006). ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. BMC Bioinformatics *7 Suppl 1*, S7.
91. Kang, Y., Liow, H.-H., Maier, E.J., and Brent, M.R. (2018). NetProphet 2.0: mapping transcription factor

networks by exploiting scalable data resources. Bioinformatics *34*, 249–257.
92. Huynh-Thu, V.A., and Geurts, P. (2018). dynGENIE3: dynamical GENIE3 for the inference of gene networks from time-series expression data. Sci. Rep. *8*, 3384.
93. Bansal, M., Della Gatta, G., and di Bernardo, D. (2006). Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. Bioinformatics *22*, 815–822.
94. Radovic, M., Ghalwash, M., Filipovic, N., and Obradovic, Z. (2017). Minimum redundancy maximum relevance feature selection approach for temporal gene expression data. BMC Bioinformatics *18*, 9.
95. Roy, S., Lagree, S., Hou, Z., Thomson, J.A., Stewart, R., and Gasch, A.P. (2013). Integrated module and gene-specific regulatory inference implicates upstream signaling networks. PLoS Comput. Biol. *9*, e1003252.
96. Zoppoli, P., Morganella, S., and Ceccarelli, M. (2010). TimeDelay-ARACNE: Reverse engineering of gene networks from time-course data by an information theoretic approach. BMC Bioinformatics *11*, 154.
97. Han, H., Cho, J.-W., Lee, S., Yun, A., Kim, H., Bae, D., Yang, S., Kim, C.Y., Lee, M., Kim, E., et al. (2018). TRRUST v2: an expanded reference database of human and mouse transcriptional regulatory interactions. Nucleic Acids Res. *46*, D380–D386.
98. Altschul, S. F., Gish, W., Miller, W., Myers, E. W., & Lipman, D. J. (1990). Basic local alignment search tool. *Journal of molecular biology*, *215*(3), 403-410.
99. Leek, J. T., Johnson, W. E., Parker, H. S., Jaffe, A. E., & Storey, J. D. (2012). The sva package for removing batch effects and other unwanted variation in high-throughput experiments. *Bioinformatics*, *28*(6), 882-883.