UMass Chan Medical School
eScholarship@UMassChan

Open Access Publications by UMMS Authors

2015-10-08

# TRANSIT - A Software Tool for Himar1 TnSeq Analysis

Michael A. DeJesus
*Texas A&M University*

*Et al.*

Let us know how access to this document benefits you.
Follow this and additional works at: https://escholarship.umassmed.edu/oapubs

 Part of the Bioinformatics Commons, and the Computational Biology Commons

## Repository Citation

DeJesus MA, Ambadipudi C, Baker RE, Sassetti CM, Ioerger TR. (2015). TRANSIT - A Software Tool for Himar1 TnSeq Analysis. Open Access Publications by UMMS Authors. https://doi.org/10.1371/journal.pcbi.1004401. Retrieved from https://escholarship.umassmed.edu/oapubs/2600

Creative Commons License

This work is licensed under a Creative Commons Attribution 4.0 License.
This material is brought to you by eScholarship@UMassChan. It has been accepted for inclusion in Open Access Publications by UMMS Authors by an authorized administrator of eScholarship@UMassChan. For more information, please contact Lisa.Palmer@umassmed.edu.

PLOS | COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

# TRANSIT - A Software Tool for Himar1 TnSeq Analysis

**Michael A. DeJesus[1]*, Chaitra Ambadipudi[1], Richard Baker[2], Christopher Sassetti[2], Thomas R. Ioerger[1]**

**1** Department of Computer Science, Texas A&M University, College Station, Texas, United States of America, **2** Department of Microbiology and Physiological Systems, University of Massachusetts Medical School, Worcester, Massachusetts, United States of America

* mad@cs.tamu.edu

CrossMark
click for updates

OPEN ACCESS

**Citation:** DeJesus MA, Ambadipudi C, Baker R, Sassetti C, Ioerger TR (2015) TRANSIT - A Software Tool for Himar1 TnSeq Analysis. PLoS Comput Biol 11(10): e1004401. doi:10.1371/journal.pcbi.1004401

**Editor:** Paul P Gardner, University of Canterbury, NEW ZEALAND

**Received:** April 27, 2015

**Accepted:** June 10, 2015

**Published:** October 8, 2015

**Copyright:** © 2015 DeJesus et al. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Data Availability Statement:** Data and Soure Code are available at the following GitHub Repository: https://github.com/mad-lab/transit

**Funding:** This work was supported by the National Institutes of Health (www.nih.gov/) grant U19 AI107774. TRI and CS received funding. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

**Competing Interests:** The authors have declared that no competing interests exist.

## Abstract

TnSeq has become a popular technique for determining the essentiality of genomic regions in bacterial organisms. Several methods have been developed to analyze the wealth of data that has been obtained through TnSeq experiments. We developed a tool for analyzing Himar1 TnSeq data called TRANSIT. TRANSIT provides a graphical interface to three different statistical methods for analyzing TnSeq data. These methods cover a variety of approaches capable of identifying essential genes in individual datasets as well as comparative analysis between conditions. We demonstrate the utility of this software by analyzing TnSeq datasets of *M. tuberculosis* grown on glycerol and cholesterol. We show that TRANSIT can be used to discover genes which have been previously implicated for growth on these carbon sources. TRANSIT is written in Python, and thus can be run on Windows, OSX and Linux platforms. The source code is distributed under the GNU GPL v3 license and can be obtained from the following GitHub repository: https://github.com/mad-lab/transit

This is a *PLOS Computational Biology* Software paper

## Introduction

Transposon insertion sequencing (TnSeq for short) is a popular experimental methodology for determining essential (and conditionally essential) regions in bacterial genomes [1]. TnSeq (in the broad sense used in this paper) refers to a family of related methods that use deep sequencing to survey a transposon insertion library and quantify the abundance of insertions at different sites in the genome [2–5]. The specific methodologies differ in the details of library preparation (such as use of shearing versus digestion, method of enrichment, or the choice of transposable element) [6]. While there are several tranposons that can be used to construct Tn

insertion libraries, one of the most commonly used is the Himar1 transposon, which is used in several specific protocols including HITS [3], Tn-seq [4], and INSeq [2]. The Himar1 transposon inserts at random TA dinucleotide sites during the library generation process [7]. Depending on size of gene and GC-content there are typically between 5 to 50 TA sites per gene. Essential regions are inferred by the lack of insertions observed in a region (presumably because the insertion of the transposon (Tn) disrupts the protein product, making it non functional). Conditionally essential regions have insertions in one condition but not in another (See Fig 1). Knowledge of (conditionally) essential genes plays an important role in drug discovery, as these could be drug targets, and the ability to detect conditionally essential genes is helpful in working out pathways (e.g. comparative analysis between samples with and without supplementation of a critical metabolite).

The preparation of TnSeq samples for sequencing involves fragmenting genomic DNA, attaching sequencing adapters, and amplifying with PCR primers to enrich the sample for fragments carrying Tn:genomic junctions. Illumina next-generation sequencers are the most frequently used platform to sequence TnSeq libraries. The datasets generated from an Illumina sequencer contain short reads ( $\sim$ 100 bp) that have the terminus of the Tn as a prefix and a genomic suffix that can be mapped (aligned) to the genome to identify which TA site they represent.

While the relative abundance of insertion mutants can be estimated based on the frequency of read counts or template counts corresponding to an insertion site, stochastic effects during amplification and library generation can also influence these measurements. Despite these fluctuations, some regions show systematically suppressed (or inflated) counts, which could reflect a gene whose disruption causes a growth defect (or growth advantage). In addition, there can also be missing sites not represented in the library. Analysis of TnSeq data is challenging, especially with low density libraries where there are large number of TA sites not represented in the library. Several methods have been proposed for rigorously quantifying the statistical significance of essential regions, including models using the Negative Binomial distribution [8] and the Poisson distribution [9], a non-parametric test based on re-sampling counts within a sliding window [10], Bayesian methods [11] and Hidden Markov Models [12, 13].

TRANSIT is a new software tool that automates the analysis of Himar1 TnSeq datasets. It has a graphical interface that allows a user to load TnSeq datasets and apply several built-in analyses to identify essential (and conditionally essential) regions, calculate statistical significance, and visualize the results in different ways. For essentiality analysis, TRANSIT provides two alternative methods: a Bayesian method that quantifies the significance of "gaps" (or long consecutive sequences of TA sites lacking insertions) [11], complemented by a Hidden Markov Model (HMM) that also incorporates local differences in read counts [13]. For comparative analysis, TRANSIT utilizes a permutation test that compares the difference of the counts in a genomic region between two different conditions to determine if there is a statistically significant difference (e.g. putatively reflecting selection for or against disruption in one of the two conditions). A pre-processor called TPP (for TRANSIT Pre-Processor) is provided which extracts read counts from raw sequence data files (in .fasta,.fastq or fastq.gz format), maps them to the reference genome, optionally reduces them to unique template counts, and outputs them in .wig format for loading into TRANSIT. Finally, numerous statistics are generated for analyzing the quality of TnSeq datasets and diagnosing any potential problems (i.e. with the library or sample preparation).

TRANSIT was initially designed to analyze TnSeq libraries prepared by the protocol in [14], which uses a custom barcoding scheme (unique nucleotides which occur in read 2). TPP has a default mode to recognize these barcodes and use them to reduce read counts to template counts, as described below. However, TRANSIT was intentionally designed in a modular way to decouple the preprocessing from the computational analysis to allow the statistical analysis

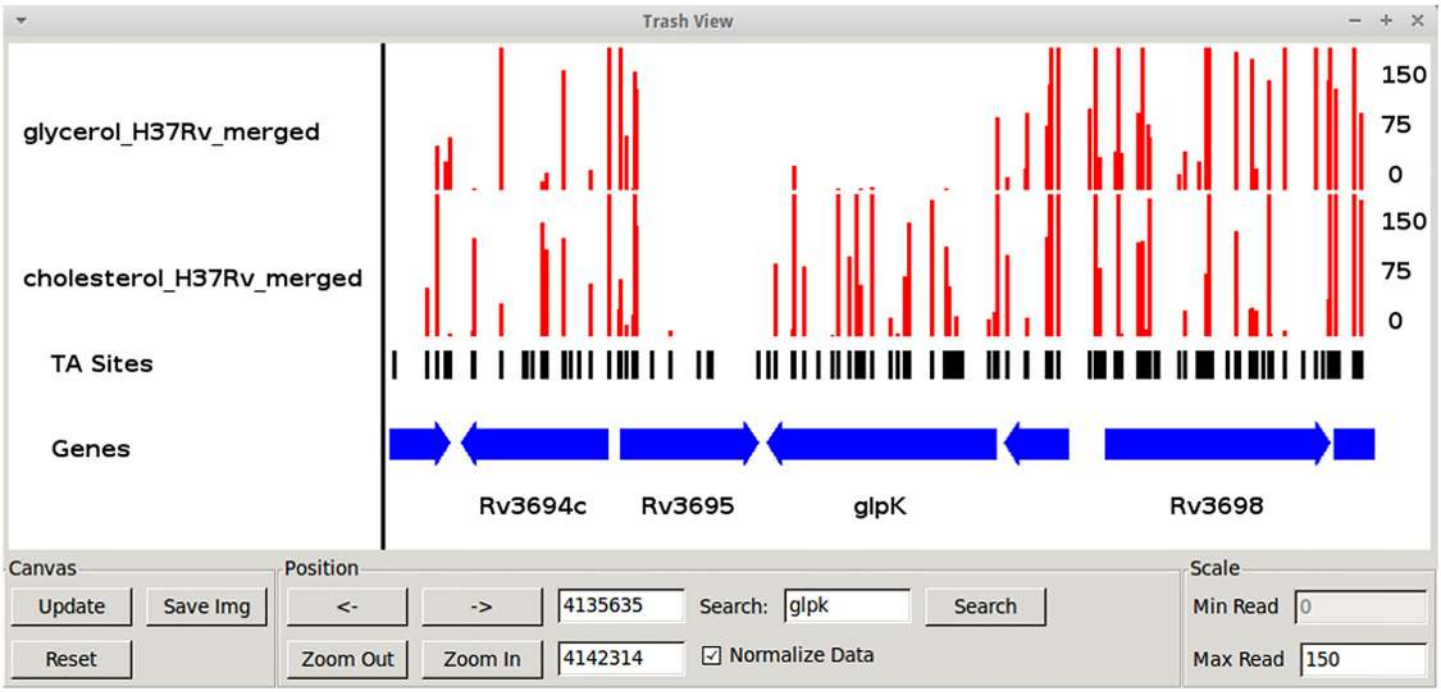



**Fig 1. Track View of read counts for datasets grown in glycerol and cholesterol.** This region spans approximately 12 kb, and includes 5 genes. TA dinucleotides, which are candidate insertion sites, are indicated in the middle track. Vertical height of each bar reflects # of reads or Tn insertions at each TA site. Some sites with no insertions are probably missing from the library, while others may reflect essential regions. Note that GlpK lacks insertions in the glycerol condition, indicating that it is essential when grown on glycerol.

doi:10.1371/journal.pcbi.1004401.g001

tools of TRANSIT to be applied to datasets obtained from other (Himar1) TnSeq protocols [2–4]. For example, if a dataset is collected with just single-ended reads, or a protocol without barcoding was used for sample preparation, TPP can configured to simply process read 1 without read 2. If an alternative barcoding scheme were used, TPP might have to be modified, or users can implement their own processing pipeline for mapping reads and quantifying insertions at genomic locations. As long as these counts are written out as intermediate files in .wig format, they can be input to TRANSIT for subsequent statistical analysis. TRANSIT could in principle be modified to analyze other TnSeq libraries such as those generated with the Tn5 transposon [5].

Several other software tools have been developed for analysis of TnSeq data. Some are purely computational [12, 15] and do not have the convenient graphical features of TRANSIT, such as TrackView (to display insertion patterns at various loci) or Volcano plots (to visualize the distribution of hits in comparative analysis). The most similar alternatives are ESSENTIALS [8] and Tn-seq Explorer [16]. ESSENTIALS uses the Negative Binomial distribution to identify essential genes and quantify their statistical significance. However, it has been observed to output an excessive number of essential genes when utilizing its reported p-values for classification [16] and can be susceptible to insertions in the N- or C- terminii, causing essential genes to appear to be non-essential [11]. Tn-seq Explorer uses a sliding window approach to identify regions where there is a deficit of reads relative to the rest of the genome. However, there is no calculation of statistical significance for the Essentiality Index (EI) computed for each gene. The permutation test in TRANSIT also provides a more statistically rigorous way to identify

conditionally essential genes (comparative analysis between conditions), in contrast to the simple comparison of EI values in Tn-seq Explorer.

## Design and Implementation

### Analysis Methods

TRANSIT provides several statistical methods capable of accomplishing two common types of tasks:

1. Identifying essential genes in a single growth condition
    a. Bayesian/Gumbel Method
    b. Hidden Markov Model
2. Identifying conditionally essential genes between conditions (comparative analysis)
    a. Resampling (permutation test)

All TnSeq analysis methods are sensitive to insertion density or library saturation (to different degrees). It is important to have sufficient diversity (or saturation) of the Tn mutant library, so that as many of the TA sites in non-essential regions are represented as possible. While saturation rarely achieves 100%, good libraries often have density greater than 50%, whereas libraries with lower density in the 20-30% range are more challenging to analyze and give less confident predictions (because a sequence of TA sites could be missing insertions due to chance).

**Bayesian/Gumbel Method.** For analyzing essentiality in single conditions, TRANSIT incorporates a Bayesian method that identifies the longest consecutive sequence of TA sites lacking insertion in a gene (or "gap"), and calculates the probability of this using the Gumbel or Extreme Value distribution [11]. The basis of this approach is that, in non-essential regions, gaps will occur by chance (depending on degree of saturation), and the probability of a long gap decreases geometrically. Thus essential genes can be recognized by unusually long gaps, and the posterior probability of the longest gap can be calculated using a Bayesian formula. The Bayesian formula is a joint probability density function Eq (1) with unobservable parameters that must be estimated using the Metropolis-Hasting sampling algorithm [11]. In the end, the Bayesian method calculates a posterior probability of essentiality (called $\bar{Z}_i$) for each gene. While p-values are not traditionally used in a Bayesian framework, a technique for controlling the false-discovery rate is used to select a threshold of posterior probabilities that can be used to determine a set of confident essentials and non-essentials that is adjusted for multiple comparisons [17]. Some genes might be labeled as Uncertain, which is important if data is too sparse or the gene is too short to make a confident call. While the Gumbel method does not take into consideration the magnitudes of the read counts, an important advantage of this analysis is that it is not sensitive to a few insertions at the N- and C-termini of essential genes (since there is often still a large gap in the middle of an essential ORF), or insertions in non-essential linker regions between domains, and it can identify "domain-essentials", which are genes

containing both an essential and non-essential domain.

$$\begin{aligned} p(Y, Z, \phi_0, \omega_1) &= p(Y \mid Z, \phi_0, \omega_1) \times \pi(\phi_0) \times \pi(Z \mid \omega_1) \times \pi(\omega_1) \\ &= \left[\prod_{i=1}^{non} Gumbel(r_i|\mu, \sigma) \times N(s_i - \lambda_r r_i, \sigma_r^2)\right] \\ &\times \left[\prod_{i=1}^{ess} \Omega(s_i) \times N(r_i - \lambda_s s_i, \sigma_s^2)\right] \times Beta(\phi_0; \alpha_0, \beta_0) \\ &\times Bin(K_z; G, \omega_1) \times Beta(\omega_1; \alpha_w, \beta_w) \end{aligned} \quad (1)$$

In this formula (from [11]), $Y_i$ represents the observed insertions patterns in each gene, $Z_i$ is a binary variable that indicates whether each gene is essential, and the other variables are internal parameters of the model that get estimated through the sampling process.

**Hidden Markov Model.** To complement the Bayesian method, TRANSIT also offers an HMM to identify essential regions in a *non-gene-centric* way (not limited by ORF boundaries) [13]. Thus the HMM can be used to identify essential loci larger than one gene, e.g. an operon, or smaller, e.g. an essential protein domain. Also it can potentially identify essential intergenic regions [10].

An HMM is a popular choice for analyzing sequential data. In this context, the HMM is applied to the sequence of TA sites to obtain the most probable state (essentiality) assignment based on the read count at the site and the distribution over the surrounding sites. In this manner, the HMM enforces a local consistency among the state assignments, despite not explicitly using a sliding window. The HMM in TRANSIT is a 4-state model (See Fig 2) that with states for: a) essential regions (ES), b) non-essential regions (NE), c) growth-defect regions (GD, with suppressed read counts), and d) growth advantaged regions (GA; with excess read counts, inflated above the global mean). The likelihood of read counts in each state is determined by a geometric distribution (based on the observation that low read counts are more frequent and sites with higher counts are more rare), where the mean is near-0 for ES, near the global mean for NE, intermediate for GD, and high for GA.

The parameters of the HMM (transition probabilities, etc.) are dynamically adjusted to the attributes of the dataset (such as insertion density and mean read count) in such a way that the model tends to remain within a state (despite a few sites that may not fit) until enough evidence accumulates to justify a transition to another state (See Fig 2). Given the transition probabilities and other parameters of the model, the state distributions for each TA site are estimated from the observed counts using the Viterbi algorithm [18]. The HMM has been shown to perform well and make reasonable essentiality calls even in datasets with density as low as 20% [13]. At the end of the analysis, the proportions of sites labeled by each of the 4 states is reported. Typically around 15% of the genome would be expected to be essential in most bacteria [19], and most of the rest of the genome would be non-essential, while only a small fraction (on the order of 5-10%) might be labeled as GD or GA. An example of a putative GA region would be one containing virulence genes, which are required *in vivo* for infection but are often lost *in vitro* because of the taxing energy requirements on the organism. As a post-processing step, the essentiality state of each gene is called based on the labeling of the majority of TA sites within the ORF.

**Resampling.** For comparative analysis, TRANSIT uses a variation of the classical permutation test in statistics [20]. For each gene, the read counts at all the TA sites and all replicates in each condition are summed, treating replicates within a condition as independent and identically distributed. The difference between the sum of read-counts at each condition is then calculated. The significance of this difference is evaluated by comparing to a resampling

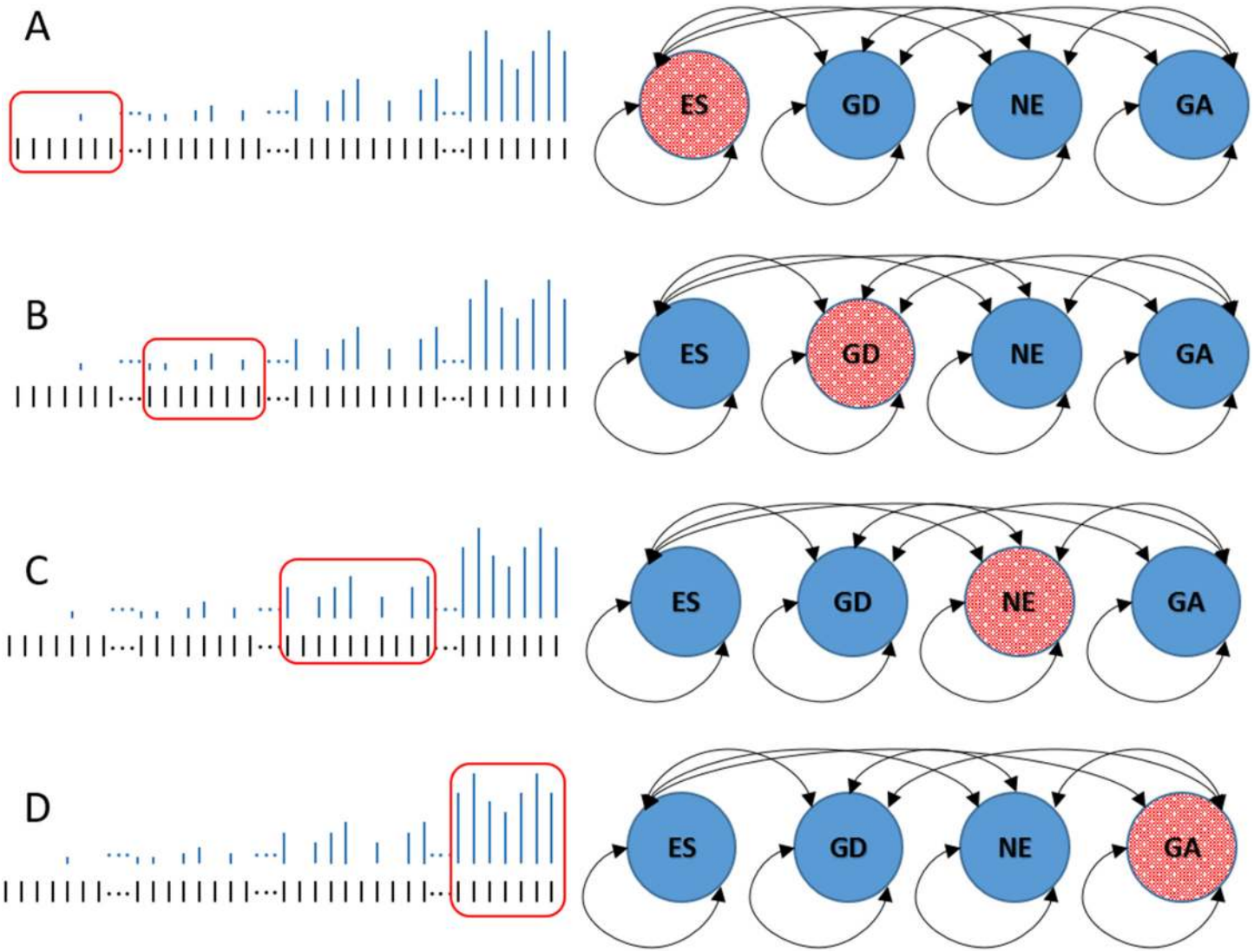



**Fig 2. Hidden Markov Model Diagram.** The HMM is fully connected, allowing transitions between each of the states. Transition probabilities and parameters are estimated in such a way that the HMM will remain in the state which best represents the read-counts observed. (a) Essential regions ("ES") are mostly devoid of insertions, (c) while non-essential regions ("NE") contain read-counts around the global mean. (b) Growth-defect regions ("GD"), and (d) growth-advantage regions ("GA") represent those areas with significantly suppressed or inflated read-counts.

doi:10.1371/journal.pcbi.1004401.g002

distribution generated from randomly reshuffling the observed counts at TA sites in the region among all the datasets. This creates a distribution of read count differences that might be observed by chance, assuming a null hypothesis that the two conditions are not in fact different. A p-value is then derived from the proportion of reshuffled samples that have a difference more extreme than that observed in the actual experimental data.

Due to the stochastic nature of read counts, there will almost always be some measurable difference between these sums. If this difference in sums of read counts falls within the bounds of the resampling distribution, this is interpreted as being due to chance. On the other hand, true conditionally essential genes will show a highly significant difference as insertions in the locus will be observed in one condition but not the other, resulting in a difference which is

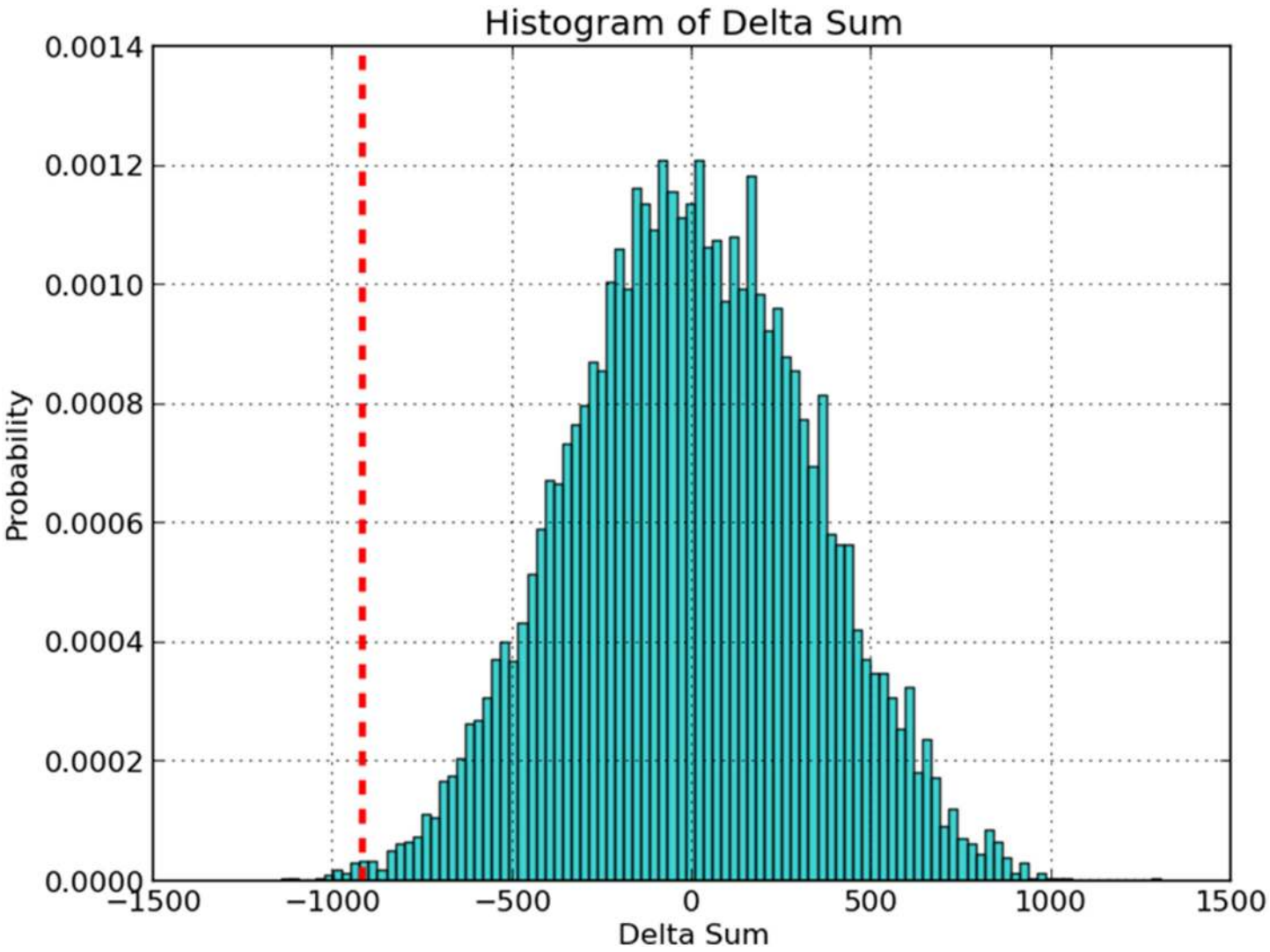



**Fig 3. Resampling histogram for gene Rv0017c.** Rv0017c has 23 TA sites, and the sum of the observed counts at the TA sites in this genes *in vitro* was 1,318 and *in vivo* was 399, therefore the observed difference in counts is -918. To determine the significance of this difference, 10,000 permutations of the counts at the TA sites among the datasets was generated and the observed differences plotted as a histogram showing that a difference as extreme as -918 almost never occurs by chance. The p-value is determined by the tail of this distribution to be 0.003 (30 out of 10,000).

doi:10.1371/journal.pcbi.1004401.g003

typically much larger than any of the differences observed by randomly re-shuffling. Furthermore, this method can detect genes whose disruption leads to a reduction in fitness; that is, genes which are not absolutely essential in one of the conditions, but instead have lower read-counts in one of the conditions compared to the other. The permutation test distinguishes which of these differences is statistically significant. p-values are derived from the fraction of samples that exceed the observed difference (See Fig 3), and this is adjusted for multiple comparisons by the Benjamini-Hochberg procedure.

The permutation test requires that the datasets be comparably normalized. TRANSIT provides several alternative ways to normalize TnSeq data, each with different strengths and weaknesses in dealing with various sources of noise in real datasets. The default normalization procedure is the non-zero mean (NZmean) method to normalize datasets to have the same mean over non-zero sites. In our experience, this is better than normalizing by the total read-counts, which is sensitive to the degree of saturation of a library. The normalization is achieved by dividing by the total number of reads in the dataset by the total number of sites with at least one insertion, and using this (and the desired mean) as a scaling factor:

$$\sigma_j = \mu_g \times \frac{\text{Number of sites with an insertion}}{\text{Total number of reads in dataset } j}$$

where $\mu_g$ is the global mean read-count across all datasets. The normalized counts at each site $i$ in dataset $j$ are the raw counts times the scaling factor ($c'_{ij} = c_{ij} \times \sigma_j$).

## Pre-Processing

TRANSIT takes .wig files as input, which contain counts of reads (or unique templates) observed at each each TA site. In this way, TRANSIT accepts datasets prepared with any pre-processing or custom protocol. An optional pre-processor called called TPP is provided with the software distribution for extracting these counts from raw sequencing files (typically in. fastq format, paired-end reads in two files called "read1" and "read2"). Note that this pre-processing procedure is designed for libraries prepared in adherence with the protocol described in [14]. However, other labs might want to apply their own custom pre-processing procedure, particularly if they use an alternative protocol for preparing TnSeq samples for sequencing or if they use a different transposon.

TPP uses BWA (Burroughs-Wheeler Aligner; [21]) to map reads into the genome. The workflow performed by TPP (Fig 4) can be briefly summarized as follows: First, read 1 is analyzed to identify the subset of reads that have a prefix matching the terminus of the Himar1 transposon (ACTTATCAGCCAACCTGTTA). The transposon prefix is stripped off, and the genomic suffix is mapped onto the genome to identify the TA site (and strand) represented by each read. Then, read 2 is analyzed to extract both a random nucleic-acid barcode (See Fig 4) and genomic suffix. The genomic suffix is also mapped onto the genome and represents the end-point of the original DNA fragment (typically a few hundred bp away). All the reads mapping to a given TA site are reduced to unique "template" counts by discarding duplicates that have the same barcode and end-point, and these template counts are written out in .wig format (for input to TRANSIT). (If barcodes were not applied during sample prep or only single-ended data was collected, TPP can be run optionally without providing read 2, in which case raw read counts are output, without reduction to unique template counts.) In our experience, this barcoding data-reduction technique has proved useful in reducing PCR effects, which can artificially bias the read counts depending on which fragments amplify more efficiently. We have observed that this is especially true for noisier datasets, where read counts are highly variable (though recent optimizations in the PCR protocol have mitigated this problem somewhat [10]) and the resulting template counts better reflect the true abundance of distinct mutants in the population. It can take on the order of an hour to process each dataset (depending on size of dataset as well as speed of computer), which is dominated by the time required to align the genomic parts of the reads to the genome using BWA. Ideally, it is recommended to collect datasets with 5-10 million pairs of reads, which often gets reduced to $\sim$ 2 million unique templates, in order to have sufficient dynamic range of counts for analysis (for example, aiming for

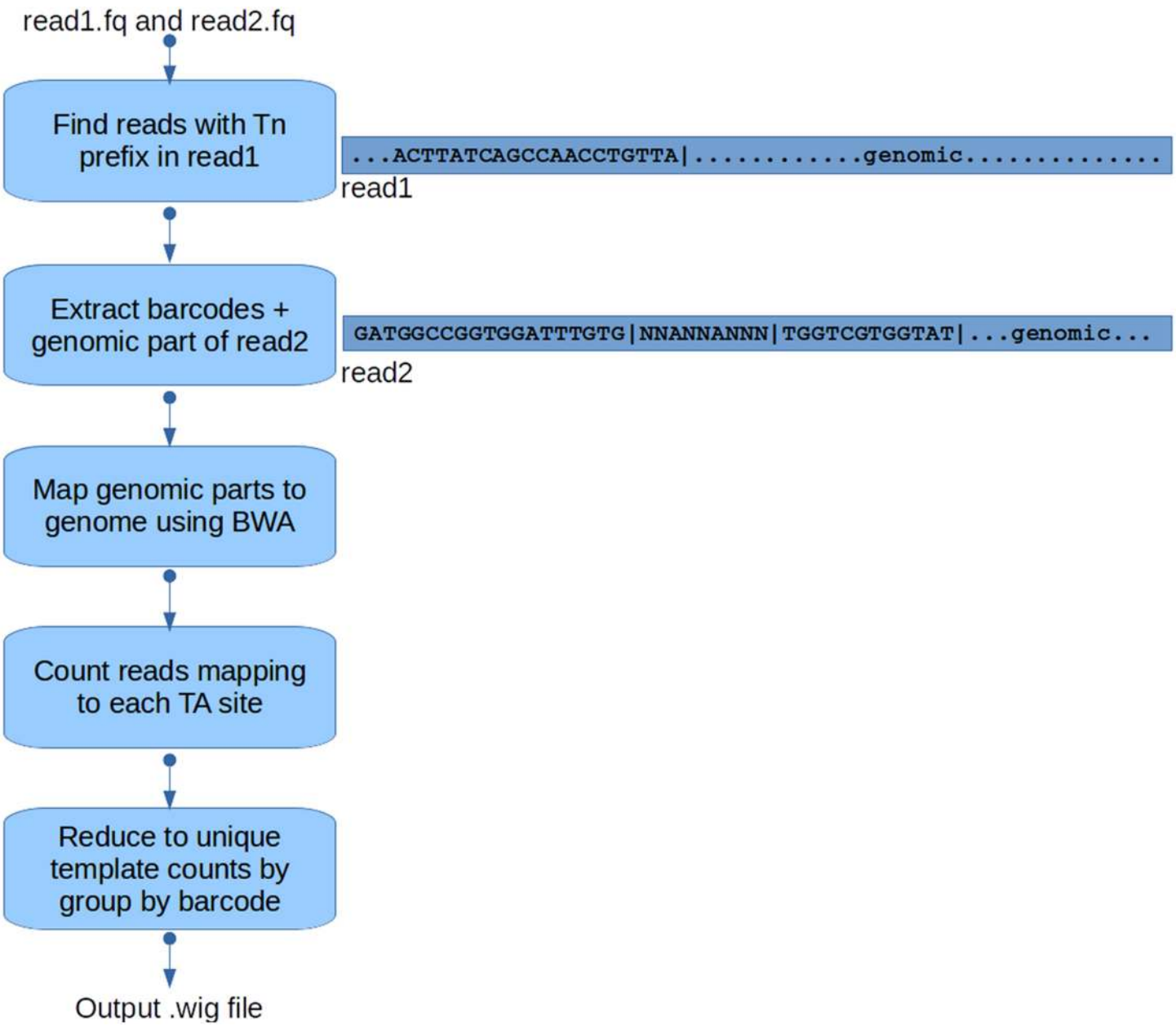



**Fig 4. TPP flowchart.** Reads in .fasta, .fastq or fastq.gz format are taken in as input, and mapped to the genome to get read-counts at individual TA sites. A .wig formatted file is returned as output, containing the coordinates and the read-counts at all TA sites in the genome.

doi:10.1371/journal.pcbi.1004401.g004

a nominal mean of $\sim$ 50 templates per TA sites, estimated based on $\sim$ 75,000 TA sites in the *M. tuberculosis* genome, with library saturation of around 50%).

Multiple statistics are calculated by TPP for diagnostic purposes. The primary metrics used to assess the quality of a TnSeq dataset are insertion density and mean read count, which should be $P_{ins} > 30\%$ and NZmean $> 10$. Additional statistics are reported, such as number of reads with valid Tn prefixes, number of reads mapping to genome (broken down into read 1, read 2, and both), correlation of reads at each TA sites on forward versus reverse strand, ratio

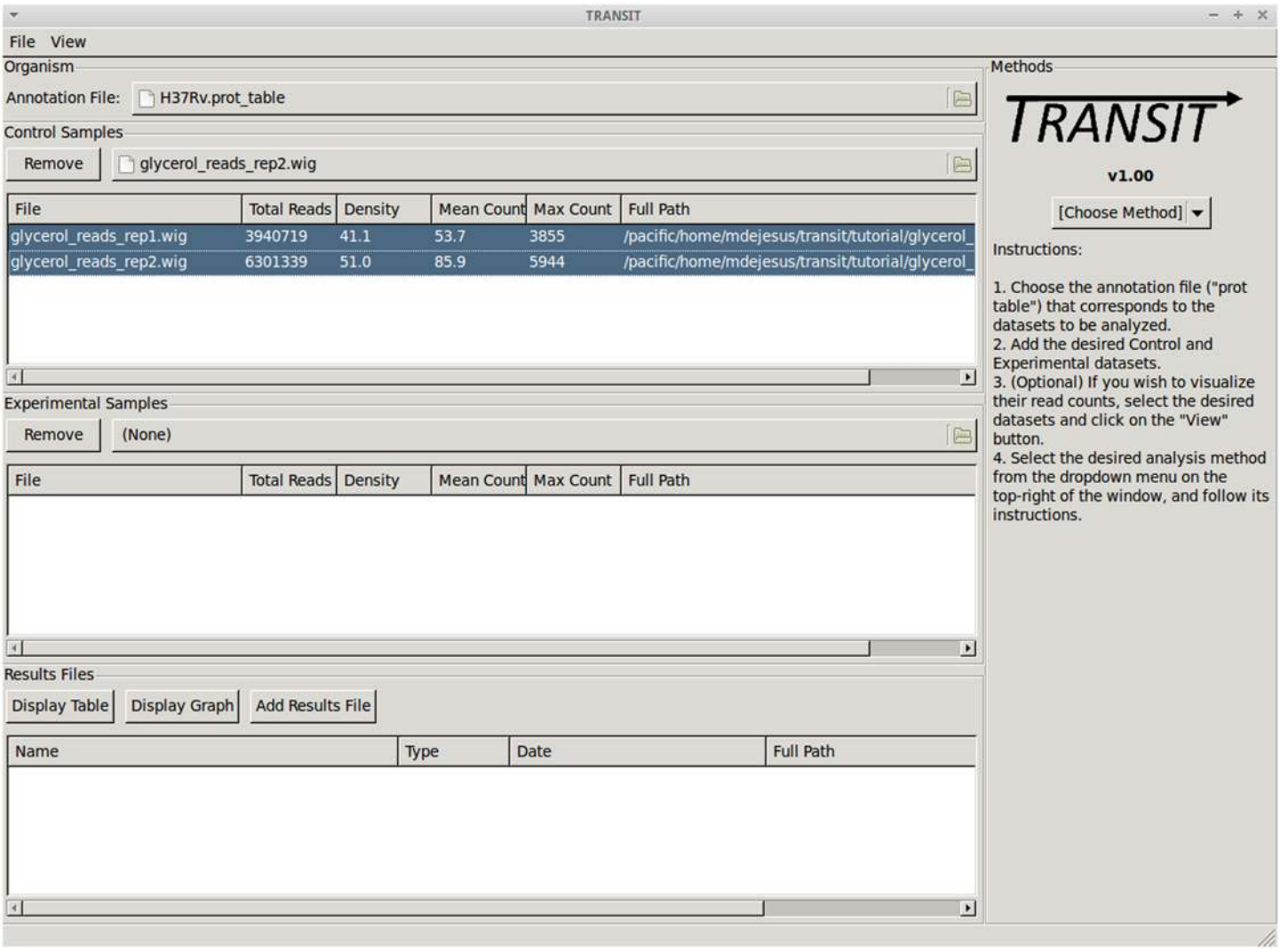



**Fig 5. Picture of the main TRANSIT interface.**

doi:10.1371/journal.pcbi.1004401.g005

of reads to templates, etc. These metrics are important for diagnostic purposes. In addition, specific nucleotide sequences representing the vector or primer are counted. If the number of mapped reads is low, then the user could check to see if there is a large fraction of reads matching these sequences, which could indicate phage contamination in the library (left over from the original transfection in constructing the library) or excessive primer-dimers lacking genomic inserts generated during sample preparation.

## Interface

The main TRANSIT interface (Fig 5) allows the user to select an annotation file (in a tab-separated format called " .prot_table"), which contains the definitions of genes and their coordinates. It must match the reference genome to which reads were mapped by TPP. Several .

prot_tables are provided online for commonly used genomes, and a script for converting annotations for other organisms from Genbank to .prot_table format is also available.

From the main interface the user can also load datasets in .wig format (e.g. output by TPP), which contain the coordinates and template counts at TA coordinates throughout the genome. Once a dataset is loaded, the corresponding table will be populated with diagnostic information about the dataset, like density and mean read count. This information can be used to compare datasets and identify potential problems. TRANSIT also provides a way to create a scatter plot of the read counts in two datasets from the menu-bar at the top of the interface. In addition, the user can visualize read-counts throughout the genome using TRANSIT's Track View (also found in the menu-bar; see Fig 1).

Once the user has picked the desired datasets for comparison, they can choose which analysis they wish to perform from the drop-down menu to the right. As soon as a method is selected, the right-hand panel of TRANSIT's interface is automatically populated with the appropriate parameters for the method selected. The user may use the default parameters (which are intended to work well on most datasets) or change individual parameters as needed. Parameter definitions are provided in the documentation included with TRANSIT.

After TRANSIT completes an analysis, it will create output files in the specified location and automatically add them to a list in the results window to keep track of the results files created in a session. Output files are tab-separated so they can be opened in the user's preferred spreadsheet software (e.g. Excel). TRANSIT also has the capability of opening results files in a new window, by selecting them from the list of results files and clicking on the "Display Table" button. This list also allows the user to generate custom graphs of the results, such as volcano plots (which plots log-fold change in read counts and adjusted p-values; See Fig 6).

The results window (Fig 7) allows the user to sort on the desired column (e.g. p-values) thus facilitating the identification of genes of interest. The user can right-click on a gene to display the gene in Track View to examine the insertion patterns (Fig 1), or get other method-specific options (like histograms of the permutations obtained with the resampling method; See Fig 3).

## Results

We illustrate the utility of TRANSIT by analyzing several published TnSeq datasets of *M. tuberculosis* H37Rv as well as *H. influenza*. The H37Rv strain has a total of 74,605 TA sites distributed randomly throughout the genome. It has 3989 genes with an average of 14 TA sites per gene, with almost all genes containing at least one TA site. The TnSeq datasets analyzed came from libraries grown on glycerol, a common *in vitro* carbon source, and cholesterol, a carbon source required for infection [22]. Datasets were obtained in multiple replicates, with two replicates grown on glycerol, and three replicates grown on cholesterol. The insertion density of the replicates was in the range of 40% to 60%, with mean template-counts ranging from 50-90 per TA site.

### Bayesian/Gumbel Method

To identify essential genes we analyzed the datasets grown on glycerol using the Bayesian/Gumbel Method, which performs an analysis on an individual condition. After loading the glycerol replicates and the annotation file into TRANSIT, and running the Gumbel method with default parameters, we obtained an output file with results.

A total of 674 genes was found to be essential ($\bar{Z}_i > \theta_e$, where $\theta_e$ is a threshold determined by the method used to control the FDR) by the Gumbel method (16.3%), matching expectations that typically 15% of bacterial genomes are necessary for growth [19, 23]. The Gumbel method also identified 2670 non-essential genes ($\bar{Z}_i < \theta_n$), with the remainder being labeled as

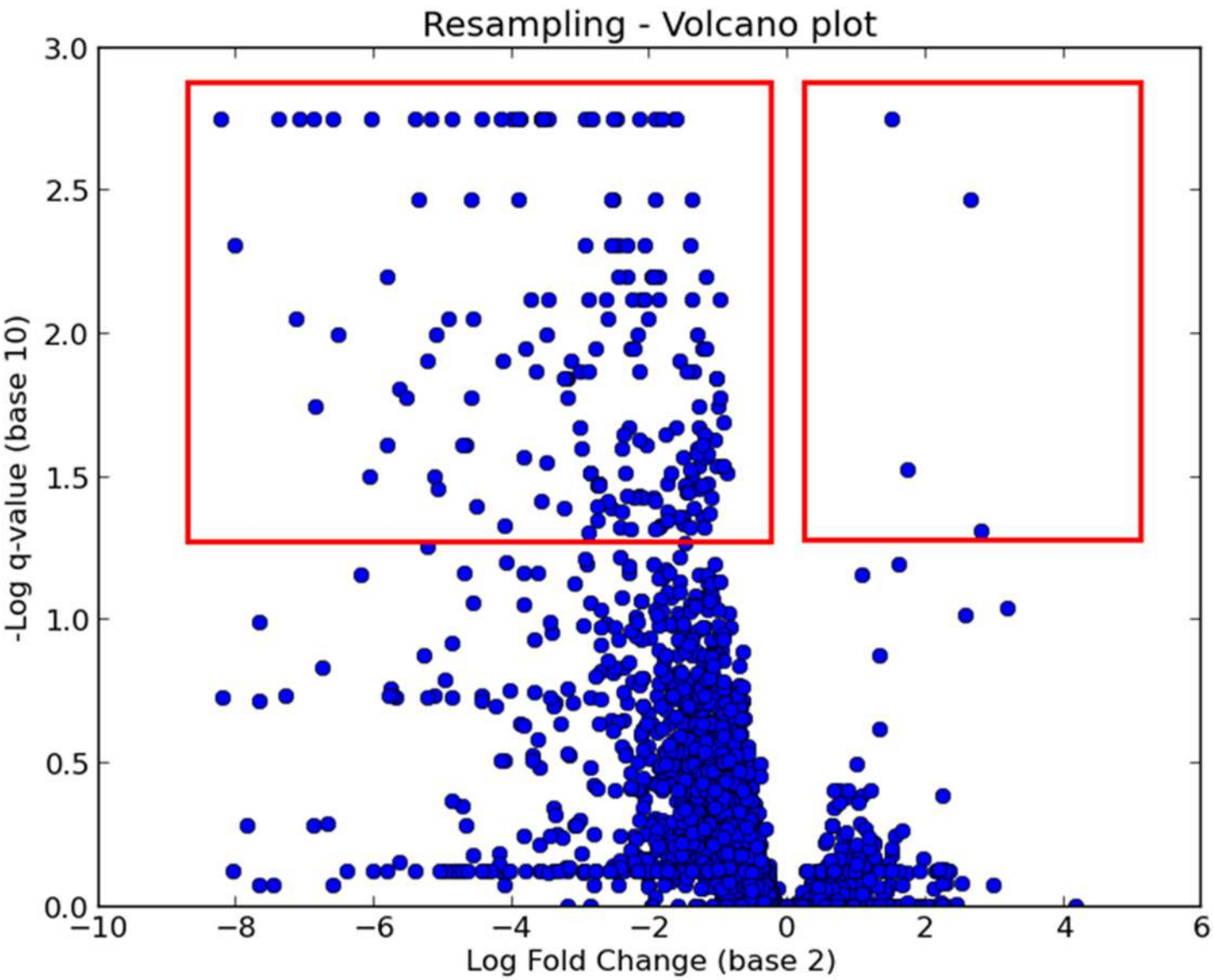
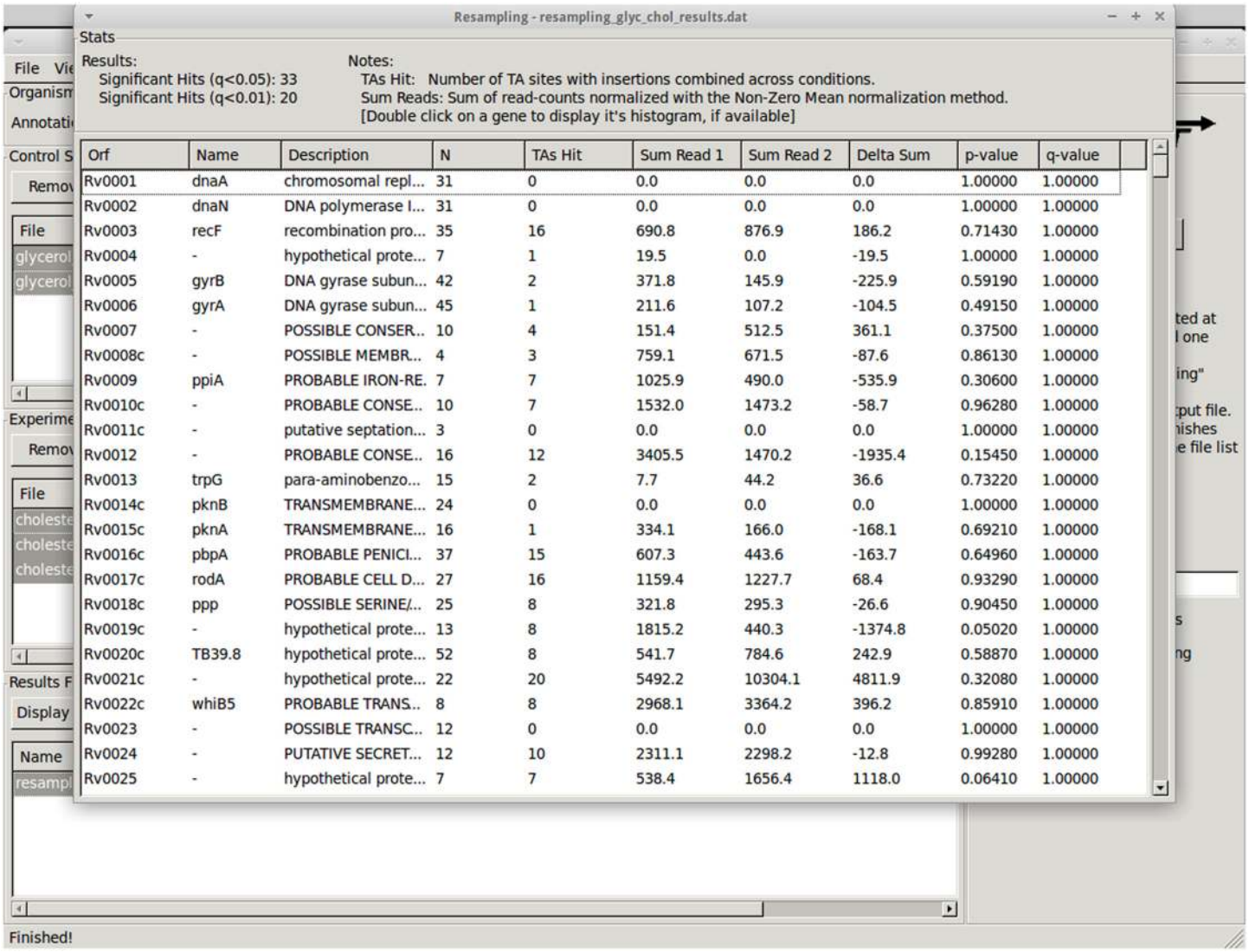



**Fig 6. Volcano plot of resampling results comparing replicates grown *in vitro* versus *in vivo*.** Significant hits have $q < 0.05$ or $-log_{10}\, q > 1.3$. Note that some genes have increased essentiality (fewer insertions; left side) and some decreased essentiality (right side).

doi:10.1371/journal.pcbi.1004401.g006

Uncertain (because the posterior probability did not exceed the significance thresholds), or were too short for reliable analysis. Table 1 contains a summary of the classifications obtained by the Gumbel method.

Well-known essential genes like GyrA (DNA gyrase A) and RpoB (DNA-directed RNA-polymerase) were identified as essential by the Gumbel method, both achieving a posterior probability of essentiality of 1.0. Those genes are completely devoid of insertions (aside from a few insertions at the N/C termini). However, one of the strengths of the Gumbel method is that it can also identify genes which contain both essential and non-essential regions, indicative of essential domains. An example of such a gene is Rv3910, which codes for an essential MviN




| Orf | Name | Description | N | TAs Hit | Sum Read 1 | Sum Read 2 | Delta Sum | p-value | q-value |
|---|---|---|---|---|---|---|---|---|---|
| Rv0001 | dnaA | chromosomal repl... | 31 | 0 | 0.0 | 0.0 | 0.0 | 1.00000 | 1.00000 |
| Rv0002 | dnaN | DNA polymerase I... | 31 | 0 | 0.0 | 0.0 | 0.0 | 1.00000 | 1.00000 |
| Rv0003 | recF | recombination pro... | 35 | 16 | 690.8 | 876.9 | 186.2 | 0.71430 | 1.00000 |
| Rv0004 | - | hypothetical prote... | 7 | 1 | 19.5 | 0.0 | -19.5 | 1.00000 | 1.00000 |
| Rv0005 | gyrB | DNA gyrase subun... | 42 | 2 | 371.8 | 145.9 | -225.9 | 0.59190 | 1.00000 |
| Rv0006 | gyrA | DNA gyrase subun... | 45 | 1 | 211.6 | 107.2 | -104.5 | 0.49150 | 1.00000 |
| Rv0007 | - | POSSIBLE CONSER... | 10 | 4 | 151.4 | 512.5 | 361.1 | 0.37500 | 1.00000 |
| Rv0008c | - | POSSIBLE MEMBR... | 4 | 3 | 759.1 | 671.5 | -87.6 | 0.86130 | 1.00000 |
| Rv0009 | ppiA | PROBABLE IRON-RE... | 7 | 7 | 1025.9 | 490.0 | -535.9 | 0.30600 | 1.00000 |
| Rv0010c | - | PROBABLE CONSE... | 10 | 7 | 1532.0 | 1473.2 | -58.7 | 0.96280 | 1.00000 |
| Rv0011c | - | putative septation... | 3 | 0 | 0.0 | 0.0 | 0.0 | 1.00000 | 1.00000 |
| Rv0012 | - | PROBABLE CONSE... | 16 | 12 | 3405.5 | 1470.2 | -1935.4 | 0.15450 | 1.00000 |
| Rv0013 | trpG | para-aminobenzo... | 15 | 2 | 7.7 | 44.2 | 36.6 | 0.73220 | 1.00000 |
| Rv0014c | pknB | TRANSMEMBRANE... | 24 | 0 | 0.0 | 0.0 | 0.0 | 1.00000 | 1.00000 |
| Rv0015c | pknA | TRANSMEMBRANE... | 16 | 1 | 334.1 | 166.0 | -168.1 | 0.69210 | 1.00000 |
| Rv0016c | pbpA | PROBABLE PENICI... | 37 | 15 | 607.3 | 443.6 | -163.7 | 0.64960 | 1.00000 |
| Rv0017c | rodA | PROBABLE CELL D... | 27 | 16 | 1159.4 | 1227.7 | 68.4 | 0.93290 | 1.00000 |
| Rv0018c | ppp | POSSIBLE SERINE/... | 25 | 8 | 321.8 | 295.3 | -26.6 | 0.90450 | 1.00000 |
| Rv0019c | - | hypothetical prote... | 13 | 8 | 1815.2 | 440.3 | -1374.8 | 0.05020 | 1.00000 |
| Rv0020c | TB39.8 | hypothetical prote... | 52 | 8 | 541.7 | 784.6 | 242.9 | 0.58870 | 1.00000 |
| Rv0021c | - | hypothetical prote... | 22 | 20 | 5492.2 | 10304.1 | 4811.9 | 0.32080 | 1.00000 |
| Rv0022c | whiB5 | PROBABLE TRANS... | 8 | 8 | 2968.1 | 3364.2 | 396.2 | 0.85910 | 1.00000 |
| Rv0023 | - | POSSIBLE TRANSC... | 12 | 0 | 0.0 | 0.0 | 0.0 | 1.00000 | 1.00000 |
| Rv0024 | - | PUTATIVE SECRET... | 12 | 10 | 2311.1 | 2298.2 | -12.8 | 0.99280 | 1.00000 |
| Rv0025 | - | hypothetical prote... | 7 | 7 | 538.4 | 1656.4 | 1118.0 | 0.06410 | 1.00000 |

**Fig 7. Table of results obtained from resampling, comparing replicates grown in glycerol versus cholesterol.**

doi:10.1371/journal.pcbi.1004401.g007

**Table 1. Table of Bayesian/Gumbel Results for H37Rv grown in glycerol.**

| Type of Gene | # of Genes |
|---|---|
| Essential | 674 |
| Uncertain | 307 |
| Non-Essential | 2670 |
| Too Small | 338 |
| Total | 3989 |

Breakdown of essentiality calls for the glycerol datasets obtained by the Bayesian/Gumbel method. Essential and Non-Essential genes are those genes whose posterior probability of essentiality exceeds the dynamic thresholds of essentiality. Uncertain genes are those who do not exceed these thresholds, and "Too Small" represents those genes who are too small for reliable analysis.

doi:10.1371/journal.pcbi.1004401.t001

**Table 2. Table of HMM Results for H37Rv grown in glycerol.**

| Type of Region | % of TA Sites (out of 74,605) |
|---|---|
| Essential | 16.3% |
| Growth Defect | 5.4% |
| Non-Essential | 77.1% |
| Growth Advantage | 1.2% |

Distribution of state calls for the glycerol datasets obtained by the HMM method. Essential states represent those regions which are mostly devoid of insertions. Non-Essential regions contain read-counts that are close to the mean read-count in the dataset. Growth-Defect regions and Growth-Advantage regions represent those regions which have significantly suppressed or increased read-counts.

doi:10.1371/journal.pcbi.1004401.t002

domain [24]. TRANSIT identifies this gene as essential, as it contains a large gap of 32 TA sites in a row without insertions, despite the fact that it has insertions on 10 out of the the remaining 17 sites. DeJesus et al. [11] discusses concordance of these results with previous essentiality analysis using the hybridization-based TraSH method [25].

## Hidden Markov Model

Another approach to analyzing an individual condition is the Hidden Markov Model. Like before, glycerol replicates were loaded into TRANSIT and the HMM method was run using default parameters.

The HMM analysis classified 16.3% of the sites in the genome as belonging to the "Essential" state, 5.4% belonging to the Growth-Defect state, 77.1% to the Non-Essential state, and 1.2% to the Growth Advantage state (See Table 2). One advantage of this method is that it is not limited to gene-boundaries but instead can assess essentiality of entire regions. For example, the PDIM locus (*fadD26*, *ppsABCDE*, *mas*; which spans $\sim$ 38kb, 594 TA sites, and 10 genes) is required for virulence *in vivo* but has high metabolic costs for the organism *in vitro* and therefore results in a Growth-Advantage for the organism when disrupted. Indeed, sites in this region (e.g. Rv2930-Rv2939) are labeled "GA" (the mean read count in this region is 502.0, a 1.9 fold increase from the global mean), thus identifying that disruption of the PDIM locus when growing on standard *in vitro* conditions affords an advantage to the organism.

**Table 3. Table of results for comparative analysis between glycerol and cholesterol.**

| Type | Count |
|---|---|
| # of genes with $q < 0.05$ in resampling | 28 |
| # of genes essential in glycerol but not cholesterol | 8 |
| # of genes essential in cholesterol but not glycerol | 20 |

Breakdown of the number of differentially essential genes identified by the resampling method, in each condition (glycerol and cholesterol). Differentially essential genes are those with an adjusted p-value $q < 0.05$.

doi:10.1371/journal.pcbi.1004401.t003

### Resampling

The resampling method can be used for comparative analysis of different growth conditions. After adding glycerol replicates as Control datasets and cholesterol replicates as Experimental datasets, the resampling method was run with default parameters.

A total of 28 genes were identified as differentially essential (adjusted p-value $< 0.05$; see Table 3 for the number of conditionally essential genes identified). Several of these genes are known to be uniquely required for growth in glycerol or cholesterol. For example, glycerol kinase (GlpK) is necessary for growth on glycerol but should not be necessary for growth on other carbon sources like cholesterol. Indeed, GlpK had a total of 1968 reads in the cholesterol condition, and only 22 total reads in glycerol, achieving an adjusted p-value (or q-value) of 0.0 with the resampling method.

Among those genes identified as necessary for growth in cholesterol were several of the Mce-family of proteins, which is believed to be involved in lipid catabolism [26]. These included Mce4A, Mce4C, Mce4D, and Mce4F, which had 3,453, 1817, 4896, and 5168 reads in glycerol respectively, and 302, 61, 180 and 32 reads in cholesterol.

Several of the genes identified as differentially essential actually contain read-counts that are significantly suppressed in one condition compared to the other, indicating a selection against insertions, hence suggesting a fitness cost to the organism. An example of such a gene is Rv3200c, which contains 167 reads in glycerol and 1,755 in cholesterol, despite having a substantial number of TA sites with insertions in both conditions (7 out of 13 in glycerol and 9 out of 13 in cholesterol), thus showing the gene can tolerate insertions in both conditions. The relative suppression in read-counts alone is enough to achieve a q-value of 0, suggesting that it is conditionally essential.

To illustrate the comparative analysis on datasets from a different organism, TRANSIT was used to perform a comparative analysis of TnSeq datasets of *H. influenza*. Gawronski et al. [3] compared two datasets of *H. influenza* grown *in vitro* and one dataset derived from lung samples after passaging through mice. The libraries were relatively sparse, with 39% mean density in vitro and 26% for the lung dataset. Gawronski et al. identified a total of 136 genes necessary for growth in lung using a combination of the log ratio of read-counts between conditions and the insertion density *in vitro*. Using TRANSIT's comparative analysis, 342 genes are obtained using a $q < 0.05$ cutoff. Out of the 136 genes identified by Gawronski et al., 133 (98%) are identified by TRANSIT as differentially essential.

## Availability and Future Directions

TRANSIT standardizes many of the complex steps (workflow) in processing analysis of TnSeq datasets, and provides a user-friendly interface that facilitates analysis of TnSeq data, particularly for libraries generated using the Himar1 transposon. The current version provides three different methods for identifying essential genes, including a method for comparative analysis of conditional essentiality between different conditions.

TRANSIT is written in the Python programming language, and can run on Linux, Macs, or Windows PCs. TRANSIT requires several Python modules (like Scipy for scientific computation, and wxPython for the user-interface), and these dependencies must also be installed. Installation instructions are provided in the manual.

TRANSIT is an Open-Source software platform that can be extended in future releases to include other analysis methods as they are developed. Source code for TRANSIT is available is distributed under the GNU GPL v3 license, and available at the following GitHub repository: https://github.com/mad-lab/transit. The package includes the Python implementation of TRANSIT and TPP, the *M. tuberculosis* TnSeq data used in this article, and documentation.

## Supporting Information

**S1 Data. Source code and datasets.** Source Code for TRANSIT and TPP, and datasets used to obtain results. Please see the GitHub Repository https://github.com/mad-lab/transit to obtain the latest version of the software.
(GZ)

## Author Contributions

Conceived and designed the experiments: MAD CA RB CS TRI. Performed the experiments: MAD CA RB CS TRI. Analyzed the data: MAD CA RB CS TRI. Contributed reagents/materials/analysis tools: MAD CA RB CS TRI. Wrote the paper: MAD CA RB CS TRI.

## References


1. Barquist L, Boinett CJ, Cain AK. Approaches to querying bacterial genomes with transposon-insertion sequencing. RNA Biol. 2013 Jul; 10(7):1161–1169. doi: 10.4161/rna.24765 PMID: 23635712
2. Goodman AL, McNulty NP, Zhao Y, Leip D, Mitra RD, Lozupone CA, et al. Identifying genetic determinants needed to establish a human gut symbiont in its habitat. Cell Host Microbe. 2009 Sep; 6(3):279–289. doi: 10.1016/j.chom.2009.08.003 PMID: 19748469
3. Gawronski JD, Wong SMS, Giannoukos G, Ward DV, Akerley BJ. Tracking insertion mutants within libraries by deep sequencing and a genome-wide screen for Haemophilus genes required in the lung. PNAS. 2009; 106(38):16422–16427. doi: 10.1073/pnas.0906627106 PMID: 19805314
4. van Opijnen T, Bodi KL, Camilli A. Tn-seq: high-throughput parallel sequencing for fitness and genetic interaction studies in microorganisms. Nat Methods. 2009 Oct; 6(10):767–772. doi: 10.1038/nmeth.1377 PMID: 19767758
5. Langridge GC, Phan MD, Turner DJ, Perkins TT, Parts L, Haase J, et al. Simultaneous assay of every Salmonella Typhi gene using one million transposon mutants. Genome Research. 2009; 19(12):2308–2316. Available from: http://www.ncbi.nlm.nih.gov/pubmed/19826075 doi: 10.1101/gr.097097.109 PMID: 19826075
6. van Opijnen T, Camilli A. Transposon insertion sequencing: a new tool for systems-level analysis of microorganisms. Nat Rev Microbiol. 2013 Jul; 11(7):435–442. doi: 10.1038/nrmicro3033 PMID: 23712350
7. Lampe DJ, Churchill ME, Robertson HM. A purified mariner transposase is sufficient to mediate transposition in vitro. the The European Molecular Biology Organization Journal. 1996; 15(19):5470–5479.
8. Zomer A, Burghout P, Bootsma HJ, Hermans PW, van Hijum SA. ESSENTIALS: software for rapid analysis of high throughput transposon insertion sequencing data. PLoS ONE. 2012; 7(8):e43012. doi: 10.1371/journal.pone.0043012 PMID: 22900082
9. Deng J, Su S, Lin X, Hassett DJ, Lu LJ. A statistical framework for improving genomic annotations of prokaryotic essential genes. PLoS ONE. 2013; 8(3):e58178. doi: 10.1371/journal.pone.0058178 PMID: 23520492
10. Zhang YJ, Ioerger TR, Huttenhower C, Long JE, Sassetti CM, Sacchettini JC, et al. Global assessment of genomic regions required for growth in Mycobacterium tuberculosis. PLoS Pathog. 2012 Sep; 8(9):e1002946. doi: 10.1371/journal.ppat.1002946 PMID: 23028335
11. DeJesus MA, Zhang YJ, Sassetti CM, Rubin EJ, Sacchettini JC, Ioerger TR. Bayesian analysis of gene essentiality based on sequencing of transposon insertion libraries. Bioinformatics. 2013 Mar; 29(6):695–703. doi: 10.1093/bioinformatics/btt043 PMID: 23361328
12. Pritchard JR, Chao MC, Abel S, Davis BM, Baranowski C, Zhang YJ, et al. ARTIST: high-resolution genome-wide assessment of fitness using transposon-insertion sequencing. PLoS Genet. 2014 Nov; 10(11):e1004782. doi: 10.1371/journal.pgen.1004782 PMID: 25375795
13. DeJesus MA, Ioerger TR. A Hidden Markov Model for identifying essential and growth-defect regions in bacterial genomes from transposon insertion sequencing data. BMC Bioinformatics. 2013; 14:303. doi: 10.1186/1471-2105-14-303 PMID: 24103077
14. Long JE, DeJesus M, Ward D, Baker RE, Ioerger TR, Sassetti CM. Identifying essential genes in *Mycobacterium tuberculosis* by global phenotypic profiling. In: Lu LJ, editor. Methods in Molecular Biology: Gene Essentiality. vol. 1279. Springer; 2015.

15. Blades NJ, Broman KW. Estimating the Number of Essential Genes in a Genome by Random Transposon Mutagenesis. Dept. of Biostatistics Working Papers, Johns Hopkins University; 2002. MSU-CSE-00-2. Available from: http://biostats.bepress.com/jhubiostat/paper15

16. Solaimanpour S, Sarmiento F, Mrazek J. Tn-seq explorer: a tool for analysis of high-throughput sequencing data of transposon mutant libraries. PLoS ONE. 2015; 10(5):e0126070. doi: 10.1371/journal.pone.0126070 PMID: 25938432

17. Muller P, Parmigiani G, Rice K. FDR and Bayesian Multiple Comparisons Rules. In: Proceedings of the ISBA 8th World Meeting on Bayesian Statistics. Benidorm, Spain; 2006.

18. Rabiner LR. A tutorial on hidden markov models and selected applications in speech recognition. In: Proceedings of the IEEE; 1989. p. 257–286.

19. Koonin EV. Comparative genomics, minimal gene-sets and the last universal common ancestor. Nature Reviews Microbiology. 2003; 1(2):127–36. Available from: http://www.ncbi.nlm.nih.gov/pubmed/15035042 doi: 10.1038/nrmicro751 PMID: 15035042

20. DeJesus MA, Ioerger TR. Reducing type I errors in Tn-Seq experiments by correcting the skew in read count distributions. In: 7th International Conference on Bioinformatics and Computational Biology (BICoB 2015); 2015.

21. Li H, Durbin R. Fast and accurate short read alignment with Burrows-Wheeler transform. Bioinformatics. 2009 Jul; 25(14):1754–1760. doi: 10.1093/bioinformatics/btp324 PMID: 19451168

22. Griffin JE, Gawronski JD, DeJesus MA, Ioerger TR, Akerley BJ, Sassetti CM. High-Resolution Phenotypic Profiling Defines Genes Essential for Mycobacterial Growth and Cholesterol Catabolism. PLoS Pathog. 2011 09; 7(9):e1002251. doi: 10.1371/journal.ppat.1002251 PMID: 21980284

23. Glass JI, Assad-Garcia N, Alperovich N, Yooseph S, Lewis MR, Maruf M, et al. PNAS. 2006; 103 (2):425–430.

24. Gee CL, Papavinasasundaram KG, Blair SR, Baer CE, Falick AM, King DS, et al. A phosphorylated pseudokinase complex controls cell wall synthesis in mycobacteria. Sci Signal. 2012; 5:ra7. doi: 10.1126/scisignal.2002525 PMID: 22275220

25. Sassetti CM, Boyd DH, Rubin EJ. Genes required for mycobacterial growth defined by high density mutagenesis. Molecular Microbiology. 2003; 48(1):77–84. Available from: http://dx.doi.org/10.1046/j.1365-2958.2003.03425.x doi: 10.1046/j.1365-2958.2003.03425.x PMID: 12657046

26. Kendall SL, Withers M, Soffair CN, Moreland NJ, Gurcha S, Sidders B, et al. A highly conserved transcriptional repressor controls a large regulon involved in lipid degradation in Mycobacterium smegmatis and Mycobacterium tuberculosis. Mol Microbiol. 2007 Aug; 65(3):684–699. doi: 10.1111/j.1365-2958.2007.05827.x PMID: 17635188