miPathoscope: Deconvolving Ambiguous RNA-Seq Alignments for Microbiome Identification using Adapted Algorithm Logic from Pathoscope

Calvin Rhodes*

Department of Molecular Biochemistry & Biophysics, Yale University

E-mail: calvin.rhodes@yale.edu

Abstract

In studying the microbiome, the essential insight is to find the relative abundance of each organism and their transcripts. Transcript quantification remains a major problem because ambiguous read alignments confuse read assignment algorithms. After surveying the available algorithms, it was clear that statistical inference methods, which infer read assignment based on all available data, were the best approach. Pathoscope was a clear choice for logic adoption because of its novel read re-assignment algorithm using unique reads as indicators and computational subtraction methodology. Here I present miPathoscope, an optimized microbiome transcript quantification pipeline for metatranscriptomics data using algorithm logic from Pathoscope and a host-centric computational subtraction methodology.

(1) Introduction

The microbiome - all microorganisms found inside humans - has recently exploded as an area of research interest; scientists all around the world are finding that a healthy microbiome correlates heavily with healthy presentation and a strong immune system. A 2014 Nature paper, for example, describes the effect of diet on the human gut microbiome, establishing a link between Bilophila wadsworthias presence and inflammatory bowel disease.[1] The essential piece of microbiome analyses is being able to determine how much of each microbe is there - each strain and each relative abundance. In doing so, strain presence can be correlated with disease presentation, illuminating potentially causal relationships that can be explored further with functionality assays. These so called abundance studies have become a powerful way to discover the different ways that the microbes in our body affect our health.

While abundance studies usually involve studying DNA and data from the genome, a less studied - but perhaps more important - area involves studying the RNA and the transcriptional activities - the transcriptome - of the microbiome. The analysis of the transcriptome has slightly different implications versus genome analysis. Quantifying the genome can only help us understand how much base genetic information - DNA - from each microbe is present in a sample, while quantifying the transcriptome allows us to quantify how much of that base genetic information is actively transcribed into RNA. Causality is more strongly implied via gene expression activity rather than just presence alone. Thus, many disease studies have utilized RNA-seq technology to gain more data on the microbiota of disease. Understanding and profiling the microbiota community and the respective transcriptional activities of each microbe can prove useful in understanding which microbes contribute to disease progression.

The type of data that comes out of transcriptional studies can vary. Classical studies of the transcriptome have centered around incubating fluorescently labelled cDNA with microarrays to yield data on existing sequences. This technique relies on the LINE element, which converts mRNA into cDNA for reinsertion into the genome. In the late 2000s, a deep-

sequencing technology called RNA-seq was developed, in which a population of RNA from a sample is converted to a library of cDNA fragments, and then sequenced in a high-throughput manner to output short (30-400 base pair) reads.[2] In microbiome analysis, processing RNA-seq data involves aligning these reads to known reference bacterial and viral genomes, and assigning these reads to the best match organism. Sequence driven methodologies have the advantage in their rapid start-to-finish rate and mostly unbiased approach to species detection, making DNA and RNA-seq analysis the newest go-to method for disease studies.[3]

One of the diseases that research has recently targeted for RNA-seq studies is asthma. Asthma is a heterogeneous disease with a high economic burden, saddling patients with the threat of dangerous symptoms and affecting 7% of the U.S. population. The mechanisms behind this stratified presentation are largely unclear, but novel efforts by Professor Geoff Chupp have revealed that the complex patterns can be sub-typed into categories using microarray expression based on sputum transcriptome data.[4] This novel non-invasive analysis characterized gene expression of sputum samples and found three significant clusters of asthma using unsupervised clustering analysis developed from the Kyoto Encyclopedia of Genes and Genomes. As a continuation of this approach, Professor Mark Gerstein and Dr. Dan Spakowicz have embarked on bulk-cell RNA-seq processing on sputum samples, a technique that measures the average expression level for each gene across a large population of input cells. From this data, they hope to reveal causal relationships between severe asthma presentation and various transcriptionally active microbiota, like Streptococcus pneumoniae, which was implicated in bacterial infections developed during the asthmatic response.[5]

The data available to us is 235 samples of bulk-cell RNA-seq data, with corresponding asthma presentation statistics as our responding variable for each sample. The missing link is the conversion from bulk-cell RNA-seq data into transcript abundance data. This conversion is essentially read assignment - assigning the RNA-seq reads to the genomes that it aligns with the best. A significant complication in read assignment is that the reads often do not map uniquely to a single gene or genome. This problem is especially frustrating in

the microbiome, since there are a number of microbial organisms with very similar genomes and thus a major source of overlapping reads. Our problem statement is thus to build an algorithm that accurately assigns ambiguous RNA-seq reads to the correct organism, translating RNA-seq data to a transcript abundance calculation.

There are a number of algorithms out there that deal with relative abundance quantifications, but none exactly fit the bill. Three broadly defined categories span the entirety of approaches to the read assignment problem: composition-based, similarity-based, and statistical inference algorithms. Composition-based relies on the intrinsic features of the reads, and tend to perform poorly on low-abundance populations. Similarity-based methods, using sequence alignment algorithms for homology search such as Bowtie, are widely considered the most sensitive methods for read classification.[6] These methods, however, are weak in that they assign each read one at a time, instead of incorporating all read information simultaneously. Methods focused on the statistical inference of organism transcript abundances and estimation of their relative proportions incorporate all read and genome alignment into one cohesive inference. The nuance lies in their handling of the multireads. Some algorithms simply discard these reads, preferring only to keep the uniquely mapping reads for gene expression estimation.[7] This is wasteful - throwing away valuable read data that can help eliminate species that dont exist in our data set. Allocating fractions of these multireads to genes has been shown to be a better approach, validated by microarray experiments.[8]

Rather than toss out the multireads, a few algorithms instead use a reduced set of reference genomes. MetaPhlAn (Metagenomic Phylogenetic Analysis) maps all reads against a reduced set of clade-specific marker sequences that uniquely identify microbial clades at the species level or higher taxonomic levels.[9] This approach, however, tosses away all parts of reference genomes that overlap - with lung microbiota, this feels wrong. It is expected that a huge portion of the reads from these samples will align to these sequences of the reference genomes that overlap, thus throwing away a large amount of our relevant data. Thus, MetaPhlAn is not sufficient for this experiment. Most algorithms out there are ideally

suited for one very specific sub-case that does not overlap with our needs. RSEM, for example, is tailored to comparing a small number of short reference genes / isoforms to the reads, something that wont suit alignment against the many large genomes in the bacterial taxonomy.[10] GRAMMy, a mixed model microbial assignment algorithm, is tailored to the gut microbiome and is fixed for genomic analysis, featuring a relatively inflexible pipeline that wouldnt work with our data.[11]

After extensive surveying of all the algorithms out there, the two that stand out as the best options for adaptation to our specific problem are RDPs Bayesian Classifier and Pathoscope, a Bayesian assignment algorithm. RDPs algorithm has been built to rapidly and accurately classify bacterial 16S rRNA sequences into the hierarchical bacterial taxonomy.[12] This algorithm works on a genus level, however, computing conditional probabilities that some sequence originates from that genus. Pathoscope, on the other hand, is tailored to pathogen identification from DNA sequencing data - not exactly our use case.[13] The underlying Bayesian assignment logic, however, is special - Pathoscope uses the reads that uniquely map to one genome to guide the reassignment of multireads. This logic, in the context of the highly similar species of the sputum microbiome, is nearly perfectly suited to our problem.

Upon examining the entirety of the field, it is clear that the best approach is that of Pathoscope, which utilizes all data available and has a novel and effective approach in using the unique reads to guide non-unique read assignment. However, due to Pathoscopes differing use case, the details - alignment algorithm used, filtering approaches, target libraries - surrounding the rock-solid logic of Pathoscope must be properly tailored. In this paper, I first state the statistical problem statement, discuss and implement Pathoscopes Bayesian mixture model approach as the skeletal framework, and optimize parameters to output a fully realized transcript abundance pipeline, called miPathoscope, for microbiome bulk-cell RNA-seq data.

(1.1) Problem Statement

Our goal is to estimate the transcriptome, the set of all expressed transcripts and relative frequencies in a sample at a given time. There is one strong measure of transcript quantification that we are interested in, θ_G, which defines the fraction of transcripts assigned to genome G. This estimation will follow from input RNA-seq data, with R sequence reads of length L, and input reference genome sequences available for G genomes.

(2) Methods

(2.1) Bayesian Mixture Model

The Pathoscope Bayesian mixture model attempts to compute the ML values of the parameters: π_j, which defines the proportion of reads mapping to the jth genome, and θ_j, which represents the reassignment proportion of non-unique reads that given to the jth genome.[13] In the reassignment process, the parameters are designed to penalize the value of non-unique reads in the presence of unique reads, and re-weight the non-unique reads based on overall mapping proportions when no reads map uniquely.

To formally describe the model, let $i = 1, ..., R$ be the index of the reads, and let $j = 1, ..., G$ be the index of the genomes. Define a set of genome indicators, $x_i = (x_{i1}, x_{i2}, ..., x_{iG}) = \{x_{ij}\}$, where $x_{ij} = 1$ if the read originated from the jth genome and $x_{ij} = 0$ if the read did not originate from genome j. Only one element in the entire vector x_i can be equal to 1 - each read should come from only one reference genome. We assume that x_i follows a multinomial distribution, with probability of success $\pi = (\pi_1, \pi_2, ..., \pi_G) = \{\pi_j\}$, where π_j is the proportion of reads that originated from the jth genome.

With reads that uniquely map to one genome, our observations are straightforward, with the x_i vector as 1 for x_{ik} where the kth genome is the unique genome aligned to i read. With the non-unique reads, the indicator x_i is treated as missing data. Our observations

here are partial mapping values for a subset of genomes, provided as posterior probabilities by the alignment program. We denote these mapping scores as $q_i = (q_{i1}, q_{i2}, ..., q_{iG}) = \{q_{ij}\}$ for the ith read. For unique reads, $q_{ij} = x_{ij}$. For non-unique reads, these values represent uncertainty in mapping, and indicate reads that need to be reassigned to the correct template genome. We thus define another parameter, $\theta = (\theta_1, \theta_2, ..., \theta_G) = \{\theta_j\}$, where θ_j is defined as the reassignment proportion to the jth genome. For ease of writing our likelihood function, we define y_i as a unique read indicator variable for read i, with $y_i = 1$ if the read maps uniquely to just one genome. Thus, $\Sigma_{i=1}^{R}(1 - y_i)$ should give the number of non-unique reads.

Given the observed data (reads, y_i, unique x_i), missing data (non-unique x_i), we can define our likelihood of our parameters (θ, π) as:

$$L(\pi, \theta | x_i, q_i, y) \propto \Pi_{i=1}^{R} \Pi_{j=1}^{G} [\pi_j \theta_j^{(1-y_i)} q_{ij}]^{x_{ij}} \tag{1}$$

(2.2) EM Algorithm

In order to estimate the model parameters, we utilize an expectation-maximization (EM) algorithm.[14] In our E step, non-unique reads are reassigned to its expected genome based on current mapping quality scores (q_i). The important step here is the re-scaling of mapping quality scores - calculated as follows:

$$E(x_{ij}) = \frac{\pi_j \theta_j^{(1-y_i)} q_{ij}}{\Sigma_{k=1}^{G} \pi_k \theta_k^{(1-y_i)} q_{ik}} \tag{2}$$

These expected values are then brought to the M-step, which calculates new estimates of π and θ given q_i, y and newly calculated expected values. The π and θ values are calculated as follows:

$$\hat{\pi_j} = \frac{\Sigma_{i=1}^{R} E(x_{ij})}{\Sigma_{k=1}^{G} \Sigma_{i=1}^{R} E(x_{ij})} \tag{3}$$

$$\hat{\theta_j} = \frac{\Sigma_{i=1}^{R}(1-y_i)E(x_{ij})}{\Sigma_{i=1}^{R}(1-y_i)} \tag{4}$$

This algorithm repeats, calculating new E-steps using new M-steps and vice versa until the estimates of π and θ converge to stable values.

(2.3) miPathoscope Parameter Optimization

With the logic of Pathoscope in place, various parameters must be optimized in order to fit our microbiome bulk-cell RNA data. First, the optimal alignment program must be chosen. Fortunately, in the updated Pathoscope paper, this comparison has already been done, with four aligners tested across mapping reads to the three genome libraries in question.[15] This result is discussed in the Results section.

Next, the reference genome libraries must be put in order. In order to reduce computational burden, Pathoscope denotes three reference libraries of genomes - the target, host, and other filtration libraries. In the optimized Clinical Pathoscope pipeline, the experimenters denoted the viral NCBI genome library as the target, with host as the human NCBI genome and other filtration as the bacterial NCBI genome library. In our situation, miPathoscope is set up to align to the bacterial genome library as the target, and filter out reads that align to the human or viral libraries.

Pathoscope also employs a novel computational subtraction methodology, in which the reads are sequentially aligned against target, host, and other filtration libraries to reduce the computational burden. In the original DNA-seq optimized pipeline, the experimenters found that the computational load was reduced if they first filtered out reads that didnt map well enough to the target library, followed by filtration via unique mapping to the host / other libraries, finally followed by mapping again to the target library with read reassignment immediately afterwards. To optimize for microbiome performance, I considered three methodologies of subtraction: nave, target centric, and host centric. Nave features no

subtraction, while target centric first filters out reads that don't align significantly to the target library, and host centric first filters out reads that uniquely align to the host and non-target libraries. The Clinical Pathoscope program, as discussed above, features a target centric approach, but it makes sense to test out the different approaches on microbiome specific test data sets. Figure 1 shows the Clinical Pathoscope pipeline.

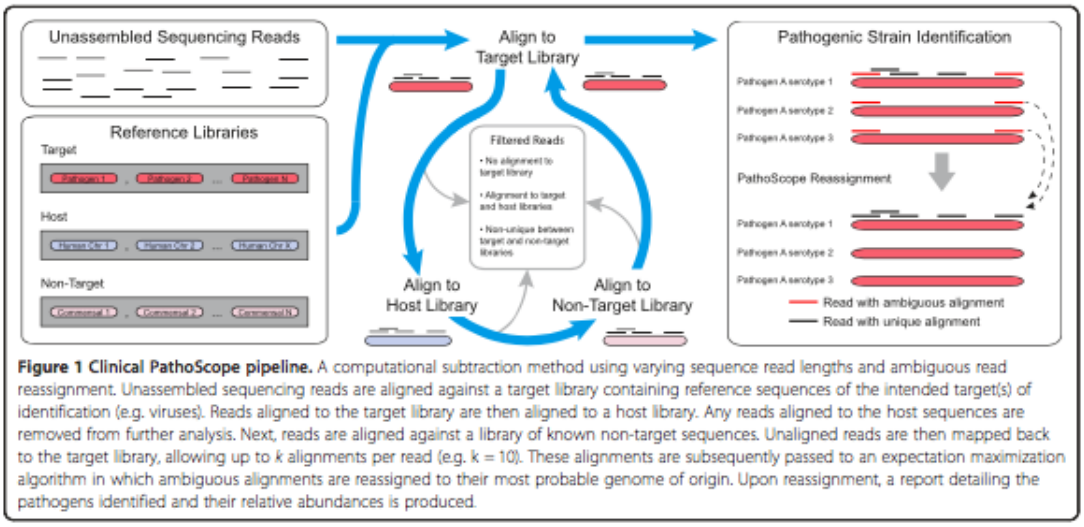

Figure 1 Clinical PathoScope pipeline. A computational subtraction method using varying sequence read lengths and ambiguous read reassignment. Unassembled sequencing reads are aligned against a target library containing reference sequences of the intended target(s) of identification (e.g. viruses). Reads aligned to the target library are then aligned to a host library. Any reads aligned to the host sequences are removed from further analysis. Next, reads are aligned against a library of known non-target sequences. Unaligned reads are then mapped back to the target library, allowing up to *k* alignments per read (e.g. k = 10). These alignments are subsequently passed to an expectation maximization algorithm in which ambiguous alignments are reassigned to their most probable genome of origin. Upon reassignment, a report detailing the pathogens identified and their relative abundances is produced.

Scheme 1: The target-centric computational subtraction method employed by Clinical Pathoscope.[15] This approach actually proved to be less effective than the host library on our miPathoscope pipeline (see results section!).

These parameters are tested on our test data sets for sensitivity, specificity, and runtime - since the original source of each read is known, true positive, false positive, true negative, and false negative amounts can be calculated per run. Run time is measured as cpu minutes on the Grace cluster of the Yale High Performance Computing center. Sensitivity and specificity are measured as follows:

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{TN + FP}$$

(2.4) Dataset Simulation

There were two sets of test data generated to validate this miPathoscope pipeline and to test the different parameters. The first data set, called S1, to test proof of concept, features two randomly generated 800-bp genomes, with 400bp of overlap between the two. In order to generate reads from these genomes, I used ArtificalFastQGenerator, a tool that can take a reference genome sequence as input and outputs artificial paired-end FASTQ files containing Phred quality scores.[16] 76-bp artificial reads were generated from three sections: (i) the 400-bp unique section of genome 1, (ii) the 400-bp unique section of genome 2, and (iii) the overlapping sequence of genomes 1 and 2. Reads from these three parts were then combined in defined quantities to output the fully realized S1. This test data set was merely included to show proof of concept, and since it is not a realistic set of data, will not be used to test miPathoscope parameters.

The second test data set, named S2, was created using artificial reads generated from 3 bacterial genomes: streptococcus pyogenes, streptococcus mitis, and staphylococcus aureus. The use of both pyogenes and mitis was for the significant overlap in their genomes. Full genomes were downloaded from NCBI, and the Muscle alignment program for multiple sequence alignment was used to align the pyogenes and mitis genomes and find their longest overlapping sequence. 76-bp reads were then generated using ArtificialFastQGenerator from four sequences: (i) the longest overlapping sequence, (ii) the unique mitis genome, (iii) the unique pyogenes genome, and (iv) the aureus genome. The reads from these four were combined in defined quantities, giving S2.

The miPathoscope parameters are tested on S2. The value of these synthetic data sets lie in the parameter optimization - since the data sets resemble closely related bacterial metatranscriptomic data, the optimized parameters should perform well on similar data moving forward.

(3) Results

(3.1) Comparison of Alignment Algorithms

As shown in Table 1 below, Byrd et al ran 5 simulated sequencing samples of 10 million 100-bp reads against the three genome libraries of interest. Since microbiome is the name of the game here, we are most interested in the bacteria results. It's clear perusing run-time, specificity, and sensitivity, that Bowtie2 performs the best across all metrics. Thus, Bowtie2 will be the alignment program of choice for miPathoscope.

Table 1 Simulation study alignment statistics using optimal model parameters

	Human		Virus		Bacteria	
	Time (m)	Sensitivity Specificity	Time (m)	Sensitivity Specificity	Time (m)	Sensitivity Specificity
Bowtie2	8.2 ± 0.0	90.2 ± 0.0 100.0 ± 0.0	3.3 ± 0.6	98 .1 ± 0.6 99.8 ± 0.2	15.8 ± 1.6	79.8 ± 0.1 100.0 ± 0.0
BWA	22.8 ± 3.2	89.9 ± 0.0 100.0 ± 0.0	6.5 ± 1.4	76.8 ± 5.4 99.8 ± 0.2	-	- -
SOAP2	5.7 ± 1.6	76.7 ± 0.0 100.0 ± 0.0	3.9 ± 0.8	50.3 ± 5.4 99.9 ± 0.1	23.3 ± 2.2	27.7 ± 0.0 100 ± 0.0
PBLAT	61.2 ± 6.8	78.2 ± 0.0 100.0 ± 0.0	16.7 ± 1.3	99.8 ± 0.1 99.6 ± 0.2	306.3 ± 23.3	98.9 ± 0.0 52.7 ± 0.0

Each aligner was used to align the first set of five simulated sequencing samples (10 million 100 base-pair reads) against each of the three genome libraries using optimal parameters. The average run time, sensitivity, and specificity as well as confidence intervals for each alignment are reported. BWA failed to run to completion with the bacterial library.

Scheme 2: The target-centric computational subtraction method employed by Clinical Pathoscope.[15] This approach, while most time efficient for metagenomics studies, actually proved slower and less effective on our simulated datasets.

(3.2) Results on Simulated Datasets

S1: Since this dataset was generated to be more proof of concept than anything, only the naive subtractional approach was used. The target databases were specified to be the two 800-bp genomes. The S1 dataset featured 25 reads from unique genome 1, 5 reads from unique genome 2, and 76 reads from the overlap. miPathoscope perfectly reported this, achieving 100% specificity and sensitivity, with a runtime of only 0.571s. All in all, a resounding proof of concept!

	S1	S2		
Statistics	Naïve	Naïve2	Host	Target
Runtime (s)	0.571	0.769	1.49	1.558
Specificity	100	42.4	98.2	97.5
Sensitivity	100	27.4	85.6	81.5

Scheme 3: Runtime, Sensitivity, and Specificity on Naive, Host-centric, and Target-centric approaches for S1, S2. This data provides evidence that show the host-centric approach might be slightly more effective in our miPathoscope pipeline.

S2: This dataset featured 10000 76-bp reads: 1000 from unique pyogenes sequence (10%), 500 from unique mitis (5%), 500 from overlap of pyogenes and mitis, and 8000 from staph (80%). The target databases were specified to be the pyogenes and mitis sequences, with filtration specified as the staph aureus dataset. As seen in Table 2 above, the host-centric method actually proved to be faster, more sensitive, and more specific. When looking into the data, it's clear that many reads were filtered out because of unique mapping to the staphylococcus aureus genome.

(3.3) Optimized miPathoscope Pipeline

The optimized miPathoscope pipeline features the bowtie2 alignment program, a target library of the NCBI bacterial library, and a host-centric computational subtraction methodology. This pipeline was validated on synthetic datasets S1, S2.

The finalized miPathoscope pipeline, plus all data supporting this paper, is hosted on Grace, the high performance computing cluster at Yale.

(4) Discussion

The optimized miPathoscope pipeline still features a couple of major question marks. Firstly, since the host-centric computational methodology was only tested on 1 synthetic dataset, there must be more validation before miPathoscope can be applied to the asthma samples with confidence that the optimal pipeline is being utilized. The next steps would be to buildout multiple samples of S2 while controlling the number of reads varying the proportion

of reads per sample, giving us average values across multiple iterations of the dataset and also giving us standard error. This would allow more confidence in our test statistics.

Additionally, a more realistic dataset should be tested. There are a couple of meta-transcriptomics datasets online, from which each read source is known and can be verified, like this 1.75million read dataset from a 2016 paper.[17] This dataset has already been tested in a publication before, meaning that any test we run on the data can be verified against this set.[18] Testing the pipeline on a dataset with a high volume of reads on a large number of genomes will be the next step towards validating miPathoscope.

In addition, another major question is how miPathoscope will handle the computational load of aligning to the entire human, viral, and bacterial genome libraries - in these studies, due to memory concerns, target and host libraries had to be tailored narrowly. The hope is that the results from S2 would scale to larger computational libraries, but anything is possible when the target library is changing from 500 MB to 25 GB of memory. In the original Clinical Pathoscope paper, each of the runtime time-scales were on the order of 5- 300 minutes, which greatly eclipses the max value of 1s performed by miPathoscope. Additional computer resources would be needed for aligning to the entirety of the bacterial / viral / human genome libraries.

With these concerns noted, it is clear that miPathoscope will perform best on the asthma samples when the target library is narrowly tailored to specific genomes of interest. This is very possible - Dr. Spakowicz has identified a literature basis for the presence of various strep strains, and along with strains identified via a pilot processing study of these RNA-seq, a more streamlined bacterial library is the next step towards identifying transcript abundances in the sputum samples. A tailored library would allow miPathoscope to perform as in the S2 dataset.

All in all, I greatly enjoyed my work on miPathoscope this semester. My favorite segment was diving into the scope of the field, and understanding each abundance program's unique EM algorithm and how these informed their effectivity and results. I have confidence

that the Pathoscope logic is the strongest I have seen, and I very much enjoyed applying it to our narrow problem!

Acknowledgement

I gratefully acknowledge allocation of supercomputing resources from the Gerstein lab at the Yale Center for Research Computing on the Grace cluster. I'd also like to thank the Gerstein lab, and in particularly Dr. Dan Spakowicz, for his guidance in picking my project, choosing the right algorithm, and structuring the actual research. Couldn't have done it without you Dan! Other thanks go to Joel Rozowsky of the Gerstein lab, my roommates for putting up with loud late night music, and Yale HPC!

References

(1) David, e. a. *Genome Biology* **2014**, *17*.

(2) Wang, e. a. *Nat Rev Genet* **2009**, *10*.

(3) Quail, e. a. *BMC Genomics* **2012**, *13*.

(4) Yan, e. a. *American Journal of Respiratory and Critical Care Medicine* **2015**, *191*.

(5) Otero, e. a. *Clin Exp Immunol* **2013**, *173*.

(6) Brady, e. a. *Nat Meth* **2009**, *6*.

(7) Marioni, e. a. *Genome Res* **2008**, *18*.

(8) Mortazavi, e. a. *Nat Methods* **2008**, *5*.

(9) Segata, e. a. *Nat Meth* **2012**, *9*.

(10) Li, B.; Dewey, C. *BMC Bioinformatics* **2011**, *12*.

(11) Xia, e. a. *PLOS ONE* **2011**, *6*.

(12) Wang, e. a. *Applied and Environmental Microbiology* **2007**, *73*.

(13) Francis, e. a. *Genome Res* **2013**, *23*.

(14) Dempster, e. a. *Journal of the Royal Stiatistical Soceity* **1977**, *39*.

(15) Byrd, e. a. *BMC Bioinformatics* **2014**, *15*.

(16) Frampton, M.; Houlston, R. *PLOS ONE* **2012**, *7*.

(17) Celaj, e. a. *Microbiome* **2014**, *2*.

(18) Narayanasamy, e. a. *Genome Biology* **2016**, *17*.