Robust statistical methods for differential abundance analysis of metagenomics data

Joseph N Paulson* Mihai Pop† Héctor Corrada Bravo‡

December 17, 2011

Abstract

This document outlines my 2011-2012 AMSC project for the 663/664 course series and in particular the mid-year progress. It is an ever evolving document. The project is to develop Metastats 2.0, a software package analyzing metagenomic data. We propose two major extensions and modifications to the Metastats software and the underlying statistical methods. The first extension of Metastats is a mixed-model zero-inflated Gaussian distribution that allows Metastats to account for a common characteristic of metagenomic data: the presence of many features with zero counts due to under sampling of the community. The number of 'missing' features (zero counts) is correlated to the amount of sequencing performed, thereby biasing abundance measurements and the differential abundance statistics derived from them. In the second extension we describe new approaches for data normalization that enable a more accurate assessment of differential abundance by reducing the covariance between individual features implicitly introduced by the traditionally used ratio-based normalization. Below I discuss an introduction and background to the problem followed by algorithms implemented and a few results obtained so far.

[1]Applied Mathematics and Scientific Computing (AMSC), Center for Bioinformatics and Computational Biology (CBCB), University of Maryland - College Park, jpaulson@umiacs.umd.edu

[2]Department of Computer Science, AMSC, CBCB, University of Maryland - College Park, mpop@umiacs.umd.edu

[3]Department of Computer Science, AMSC, CBCB, University of Maryland - College Park, hcorrada@umiacs.umd.edu

1 Introduction

1.1 Background

Metagenomics is the study of the genetic material recovered from an environmental sample. The DNA from a particular environmental sample is amplified through a process known as polymerase chain reaction (PCR). This process essentially doubles the DNA with each cycle of the process. Final DNA material is approximately $DNA_b \cdot 2^k$ where, DNA_b is the initial DNA quantity supplied, and k are the number of cycles [6].

This process is required for the next steps in the analysis pipeline. Following amplification, the DNA is sequenced, a process to determine the order of the nucleotides of a particular DNA strand. The end result are thousands of nucleotide sequences in a text file. For second generation sequencing technologies, each line in the text file consists of 30 - 400 base pairs representing a replication of a fragment of DNA. Each of these are known as a read. These reads are then annotated, a process of assigning the read to a particular organism based on a biological database. The number of reads assigned to a particular organism is an approximation of the abundance of that organism in the community. Typically the reads are first clustered according to similarity, given an arbitrary name and these clusters are annotated by their representative sequence. These clusters are known as Operational Taxanomic Units (OTUs).

In many studies, there is a goal to compare samples, as in to determine whether or not the abundance of one or more organisms is correlated with some characteristic of the sample, including health/disease status. In metagenomic data, there are many issues trying to compare samples as there is a large variation in the number of reads output by the sequencer for unknown reasons.

As there are an arbitrary number of reads output determined by the sequencing instrument, and one's ability to sample from potentially millions of bacteria in a particular environment, we are dealing with relative abundances (to a true population) where lower abundant bacteria are missed due to the sampling process. We hypothesize that many bacteria are also preferentially sampled at varying degrees. It should be noted that in many metagenomic studies, and the datasets we will use, a certain conserved / hypervariable region of a bacteria's genome is specifically sought out during the amplification and sequencing stage and used for annotation. The common region used is called 16S ribosomal DNA and refers to the $\approx$ 1,500 nucleotides that encode that region of the RNA. The 16S region is itself a subregion of the 30s subunit of a prokaryotic ribosome (unit of cells that help assemble proteins).

1.2 Previous approaches

Metagenomic studies originally focused on exploratory and validation projects, but are rapidly being applied in a clinical setting. In this setting, researchers are interested in finding characteristics of the microbiome that correlate with the clinical status of the corresponding sample [5]. Comparatively few computational/statistical tools have been developed that can assist in this process, rather most developments in the metagenomics community have focused on methods that compare samples as a whole. Specifically, the focus has been on developing robust methods for determining the level of similarity or difference between samples, rather than identifying the specific characteristics that distinguish different samples from each other.

Metastats [13] was the first statistical method developed specifically to address the questions asked in clinical studies. Metastats allows a comparison of metagenomic samples (represented as counts of individual features such as organisms, genes, functional groups, etc.) from two treatment

populations (e.g., healthy vs. disease) and identifies those features that statistically distinguish the two populations.

The underlying algorithm used by Metastats was to compute a t-statistic from the two groups for each particular feature/bacteria i: $t_i = \frac{\bar{X}_{i1} - \bar{X}_{i2}}{(s_{i1}^2/n_1 + s_{i2}^2/n_2)^{.5}}$. Following that initial observed t-statistic, an empiraclly obtained p-value would be obtained by permuting the samples B times, recalculating a t-statistic for each feature each time and taking the proportion of t-statistics greater than the originally observed value, ie. $p_i = \frac{\{|t_i^{ob}| \geq |t_i| b \in 1 \ldots B\}}{B}$.

Biomarker discovery is essential in all biological fields. In metagenomics differential abundance of taxanomic groups can elucidate key differences between one biological group from another. The goal is to discover what particular features explain the difference between healthy and pathogenic individuals for all applicable diseases or environmental differences.

The question of differential abundance has been addressed in the microarray community and more recently the RNA-Seq community. In these fields the features are the abundance of gene(s) or gene expression. The methods used in those contexts are not directly applicable for metagenomics. The particular methods used to distinguish differential abundance in the those fields were developed in response to the biases introduced by the collection and technical aspects.

RNA-seq and microarray gene expression analyses, developed methodologies targeted at reducing obscuring or technical variation specific to the data-type biases, [4], [2], [1]. The data generated in these other fields each have their own unique obscuring variation. For metagenomic data the most obvious issues are the relative abundances and sparsity of counts.

XIPE [8], was the first approach used for biomarker discovery in metagenomic samples, but was used for comparing two samples. Xipe relied on bootstrapping as there was no evidence for the data to come from any particular distribution.

The method builds a null distribution for a given feature by drawing counts randomly with replacement from the set of all counts. Then two samples of M sequences are drawn from the pooled set and the difference is the test statistic. Bootstapping empirically created the null distribution, which is then used to compare features against.

Lefse [9] is a recent methodology for biomarker discovery that makes use of non-parametric tests, in particular the non-parametric factorial Kruskal-Wallis sum-rank test [12] followed by pairwise tests among subclasses with Wilcoxon rank-sum test [12] and finished with a linear discriminant analysis (LDA) [3] to estime the impact of features. Taking into account multi-class membership, Lefse is the most recent in the field that allows comparisons many samples. After comparing 2 or more classes, Lefse attempts to estimate the effect size and determine which organisms describe the major difference between the groups.

None of the methodologies previously described take into account a biologically relevant nuance - the depth of coverage for a particular sample. The library size of a particular sample impacts how much and what is observed. Currently people do ratio normalization, converting all the counts to fractions that can affect both variance and correlation. Using a large metagenomic dataset we also infer count data follows a log-normal distribution. Using that information we can develop tests maximizing power.

2 Objectives

The main objective for Metastats2.0 is to provide a simple to use R program that will allow users to manipulate metagenomic tables of data. After the user prepares biological data in the proper

format in tab-delimited format there are multiple scripts to load the data in to R, allow the user to remove samples or features from their understanding of the project, normalize, and calculate proper statistics seen in Figure 1.

The first main extensions to the program are the two normalization methods, one a method that scales sample counts to follow a similar distribution to that of the data's reference and another that scales counts by the sum of a sample's counts up to and including the specified (typically 95th) quantile.

The second extension is an Expectation-Maximization algorithm to take into account depth of coverage for samples in the dataset and provide probabilities that a zero is a "technical zero".

3 Extension I

Our first extension are methods for data normalization that enable a more accurate assessment of differential abundance by reducing the covariance between individual features implicitly introduced by the traditionally used ratio-based normalization. These normalization techniques are also of interest for time-series analyses or in the estimation of microbial networks.

When dealing with sequencing data, there is a need to normalize count data due to the extreme variance in sample coverage and remove the arbitrariness of the sampling process from the equation []. The hope is to cleary identify the biological differences, in particular differential abundance of the particular feature, whether it be gene, 16S, or read count. Unfortunately, obscuring variation can be induced due to sample preparation, sample site, etc. In short, there is interesting variation and obscuring variation, normalization hopes to diminish the effect of obscuring variation.

The usual normalization procedure for bacterial counts is dividing each count by the sample's total counts. This introduces false correlations between taxa resulting from dividing the numerator (count, c_{ij}), for a specific taxa by a denominator derived in part by the numerator, ie. $y_{ij} = c_{ij}/N_j$ where $N_j = \sum_i c_{ij}$ [7]. However, the need to normalize across samples with different sequencing yields certainly exists when analyzing metagenomic data.

In both of the following algorithms the input will be a matrix counts of size M x N for the M features and N samples. The output will be a matrix of normalized counts, ie. some sort of scaling of the original counts.

3.1 Cumulative Distribution Normalization

In 2002, quantile normalization for micro-array data was shown to be the ideal method for normalization [1]. The technique is a method meant to make two distributions identical in statistical properties and remove the variation of non-biological origin. The motivation is coupled by the fact that certain measurements are sampled preferentially.

Similarly to quantile normalization, the assumption follows that the rate of sampling a particular measurement is similar for those with similar proportions of identified taxa. We too show a technique for making two distributions identical in statistical properties with the additional metagenomic assumption that there is a finite capacity in a metagenomic community. As such, the cumulative summation of a samples' 16s or metagenomic count should follow a similar rate to that of other samples with similar proportions of zeros at an OTU level.

Our algorithm follows (wording is similar to [1]):

- bin samples into groups, G_m, of similar zeros proportions at the OTU level; (meant to account for Zeros)
    1. given n_i samples $\in G_m$ all of length p, form X_m of dimension p x n_i;
    2. sort each column of X_i to obtain $X_{m,sort}$;
    3. replace each column of $X_{m,sort}$ with the cumulative sum of that column;
    4. take the means across rows of $X_{m,sort}$ and assign the mean to each element in the row to get $X'_{m,sort}$ and take the inverse of the cumulative norm;
    5. get $X_{m,normalized}$ by rearranging each column of $X'_{m,sort}$ to have the same ordering of the original X_m
    6. force new-nonzero features, back to zero
- scale each group's normalized counts to the median of the groups.

Following this normalization method we assert that technical zeros have been accounted for and that one can calculate various statistics following the methods found in Metastats1.0.

3.2 Cumulative sum normalization

A recent proposal for normalization of RNA-seq data is to scale counts by the 75th quantile of each samples non-zero count distribution q_{75} ie. $y_{ij} = c_{ij}/q_{75j}$ [2]. This type of normalization is motivated by the observation that a few measurements, e.g., taxa or genes, are sampled preferentially as sequencing yield increases, and have an undue influence on normalized counts derived by the usual normalization procedure. In that case, the 75th quantile was a chosen as it behaved consistently across samples. In our data, we have analyzed the distribution of non-zero counts and have determined that the 95th quantile is more appropriate. Nonetheless, the usual normalization procedure in metagenomic data does assume there is a finite capacity in metagenomic communities, which is not necessarily true in RNA-seq samples. To address this we introduce another, simpler novel normalization method denoted S_{95}, which scales counts by dividing the sum of each samples counts up to and including the 95th quantile, ie. for all samples x_j, $S_{95j} = \sum_i c_{ij} \leq q_{95j}$. This procedure addresses both issues identified above, namely, it constraints communities with respect to a total capacity, but does not place undue influence on features that are preferentially sampled.

4 Extension II

As mentioned before, many low abundant features are not "found" in a particular sample, simply because of the large sample size and low total number of reads, ie. depth of coverage.

Here we propose two major improvements to the Metastats software and the underlying statistical methods. The first extension of Metastats is a mixed-model zero-inflated Gaussian distribution that allows Metastats to account for a common characteristic of metagenomic data: the presence of many features with zero counts due to under sampling of the community. The number of 'missing' features (zero counts) correlates with the amount of sequencing performed, thereby biasing abundance measurements and the differential abundance statistics derived from them.

4.1 Zero-Inflated Gaussian Model

The zero-inflated model is defined for the continuity-corrected log of the count data:

$$y_{ij} = \log_2(c_{ij} + 1)$$

as a mixture of point mass at zero $I_{\{0\}}(y)$ and a count distribution $f_{count}(y; \mu, \sigma^2) \sim N(\mu, \sigma^2)$. Given mixture parameters π_j, we have that the density of the zero-inflated gaussian distribution for feature i, in sample j with S_j total counts and values $\theta_{ij} = \{S_j, \beta_0, \beta_1, \mu_i, \sigma_i^2\}$:

$$f_{zig}(y_{ij}; \theta_{ij}) = \pi_j(S_j) \cdot f_{\{0\}}(y_{ij}) + (1 - \pi_j(S_j)) \cdot f_{count}(y_{ij}; \mu_i, \sigma_i^2)$$

The mean is specified as, given class membership k_j:

$$E(y_{ij}|k_j) = \pi_j \cdot 0 + (1 - \pi_j) \cdot (b_{i0} + b_{i1}k_j)\,.$$

.

Based on the observation that the number of zero-valued features on a sample depend on its' total number of count s, using a binomial model, we model the mixture parameters $\pi_j(S_j)$,

$$\log \frac{\pi_j}{1 - \pi_j} = \beta_0 + \beta_1 \cdot \log(S_j)$$

To estimate the parameters we will make use of the E-M algorithm.

The input data will be a matrix of normalized count values, samples along the columns and features (organisms) along the rows, total raw counts (ie. number of reads for a particular sample) S_j, and class indicator k_j.

We have decided that an OTU-specific normalization factor would be important. As such, we adjusted the above modelled mean to be:

$$E(y_{ij}|k(j)) = \pi_j \cdot 0 + (1 - \pi_j) \cdot (b_{i0} + \eta_i \log_2(s_{95j}) + b_{i1}k(j))\,.$$

In this case, as before, parameter b_{i1} is an estimate of fold-change in mean normalized counts between the two populations. The term including $\log_2$ captures OTU-specific normalization factors through parameter η_i.

Upon investigation of the two differently modelled means we observed better detection of technical probabilities for smaller library sizes with the OTU-specific normalization factor.

4.2 Expectation-Maximization algorithm:

We can get maximum-likelihood estimates using the expectation-maximization algorithm, where we treat mixture membership $\Delta_{ij} = 1$ if y_{ij} come s from the zero point mass as latent indicator variables. Denote the full set of estimates as $\theta_{ij} = \{\eta, \beta_0, \beta_1, \eta_i, b_{i0}, b_{i1}\}$. The log-likelihood in this extended model is then

$$l(\theta_{ij}; y_{ij}, S_j) = (1 - \Delta_{ij}) \log f_{count}(y; \mu_i, \sigma_i^2) + \Delta_{ij} \log \pi_j(s_j) + (1 - \Delta_{ij}) \log\{1 - \pi_j(s_j)\}.$$

E-Step: Estimates responsabilities $z_{ij} = Pr(\Delta_{ij} = 1|\hat{\theta}_{ij}, y_{ij}) = E(\Delta_{ij}|\hat{\theta}_{ij}, y_{ij})$ as:

$$\hat{z}_{ij} = \frac{\hat{\pi}_j \cdot I_{\{0\}}(y_{ij})}{\hat{\pi}_j \cdot I_{\{0\}}(y_{ij}) + (1 - \hat{\pi}_j) f_{count}(y_{ij}; \hat{\theta}_{ij})}$$

ie. the responsibility, or proportion of counts coming from the spike-mass distribution. Notice $\hat{z}_{ij} = 0 \; \forall \; y_{ij} > 0$.

M-Step: Estimates parameters $\hat{\theta}_{ij} = \{\hat{\eta}, \hat{\beta}_0, \hat{\beta}_1, \hat{\eta}_i \hat{b}_{0i}, \hat{b}_{1i}\}$ given current estimates $\hat{z}_{ij}$:

Current mixture parameters are estimated as: $\hat{\pi}_j = \sum_{i=1}^{M} \frac{1}{M} \hat{z}_{ij}$ from which we estimate β, using least squares on the logit model as

$$\log \frac{\hat{\pi}_j}{1 - \hat{\pi}_j} = \beta_0 + \beta_1 \log (s_j)$$

.

Parameters for the count distribution are estimated using weighted least squares where the weights are $1 - \hat{z}_{ij}$. Note only samples with $y_{ij} = 0$ potentially have weights less than 1.

For up to ten iterations, at each iteration we will calculate the negative log-likelihood for each feature and determine if the estimates reached convergence for a particular feature.

4.3 P-values

From the estimated fold-change (b_{1i}) and it's standard error, we construct a t-statistic. We use Empirical Bayes method [10] to construct a moderated t-statistic and use a parametric t-distribution to obtain p-values. We found that by using a log-normal distribution, the moderated t-test was appropriate and thus substitute the permutation method used to obtain p-values in the original Metastats software. As in the previous Metastats version, we use the q-value method to correct for multiple testing.

4.4 Possible issues

There are several issues that one could potentially encounter. The biological data needs to be processed and as we have very large datasets we need to preprocess the data and remove select features. Continuing the data structure used in the original Metastats, a function that will load data much more quickly was implemented making use of R being column-oriented and its internal class structure. Two other functions were also written, one to remove features based off of low variances if the user wished, and one to remove samples that have abnormal total counts.

5 Implementation

5.1 Software

Code was developed using the R language. R is useful for the various statistical R functions and R packages available. Given time, C code that will be wrapped to in R will be implemented. The C code would include use of the OpenMP library (a parallel programming C library).

5.2 Hardware

Development on my Macbook Air, 1.6 core duo, 4 GB of ram.
Code will be run on UMIACS's computer Ginkgo
8 x Quad-core AMD Opteron Processor 8365 (2300MHz) (32 cores), 256 GB Ram, RHEL5 x86_64

5.3 Database

5.3.1 Mouse diet data

To illustrate the effects of normalization and transforming of count data we analyzed germ-free mice that were gavaged with a human fecal microbiota from a healthy donor and fed a low-fat, plant-polysaccharide-rich (LF/PP) diet for four weeks. Subsequently, half of the mice were switched to a high-fat/high-sugar Western diet. For each mouse, pyrosequencing of amplicons generated from variable region 2 of bacterial 16S rRNA genes was performed using fecal samples collected over the course of eight weeks. Sequences were assigned to taxa using the RDP Classifier (minimum confidence level = 0.8). The counts of the microbial community for each mouse tended to cluster the mice by their diet. The data is further described in [11]

5.3.2 Dysentery dataset

This data is from an ongoing project to discover novel pathogens in stool samples from children under the age of five and in third world countries. Samples were collected from four countries,Mali, Bangladesh, Kenya and Gambia. Samples were sequenced using amplification of 16S rDNA using universal primers on a 454 FLX sequencing platform. The entire set of trimmed 16S sequences totaled 3,680,225. When analyzing the large dataset we trimmed 15 samples due to low abundances.

6 Results

6.1 Normalization

We illustrate the effect of data normalization by using a metagenomic dataset that tracked the microbial community in the guts of gnotobiotic mice [11]. The longitudinal study analyzed the gut composition of $n = 6$ mice whose diet was shifted from a low-fat, plant-polysaccharide-rich (LF/PP) diet for four weeks to a high-fat/high-sugar Western diet. Another $n = 6$ mice were kept on the same diet for the same time periods. In all, the dataset comprises 54 "western" and 85 "normal" diet samplesfrom 12 mice.

We plot a heatmap in Figure 2A the tradionally normalized (ratio normalization) counts for this dataset.

In this case, and representative of many metagenomics datasets, most counts were near zero, and the overall normalized count disitrbution is heavy-tailed. Cluster analysis was unable to correctly identify the difference in microbial communities of the two mouse diets.

This is consistent with the observation that the usual normalization procedure introduce spurious correlation between features resulting from dividing the numerator (count, c_{ij}), for a specific taxa) by a denominator derived in part by the numerator, ie. $y_{ij} = c_{ij}/N_j$ where $N_j = \sum_i c_{ij}$ [7]. We observed that the majority of pairwise correlations between OTUs for this dataset when data was normalized in the usual way are non-zero and negative (Figure 2B).

2C and 2D show the improvement of our normalization method described in our paper. We used euclidean distance and hiearachical clustering, the default parameters on R's heatmap. The diets are separated and the correlations are now centered around zero.

We addressed these two issues by applying a log transform as a variance-controlling data transformation that explicitly models the mulitplicative effect of PCR amplification on count data, and by using a novel normalization technique (termed cumulative sum scaling) to control for biases in OTU PCR amplification (Materials and Methods). We plot transformed and normalized data in Figure 2C. In this case cluster analysis is able to distinguish diet. More importantly, the distribution of pairwise correlations are centered around zero, indicating that this transformation is able to control spurious correlations.

6.2 Zero-inflated mixture model for metagenomic data

Metagenomics experiments for clinical or comparative purposes have been limited to small number of samples [] []. However, experiments involving large numbers of samples are now becoming the norm due to the rapidly declining cost of high-throughput sequencing. Statistical methods for the analysis of data from experiments of this size may need to address technical biases and issues that are not observed in smaller experiments.

We developed Metastats2.0 with precisely these types of datasets in mind. As a motivating example, we analyze the largest metagenomic 16S dataset to date; a comparative metagenomics experiment that has 1007 samples of healthy and sick children from four different countries, roughly half of whom had contracted diarrhea. Total community DNA was extracted from cases and controls of children under 5 years of age from Gambia, Mali, Kenya and Bangladesh. DNA was amplified and sequenced using primers for the 16S rRNA gene.

Not surprisingly, the number of OTUs detected in a sample depends strongly on its library size (Figure 3A). This relationship between the number of OTUs detected and library size differs between experiment sites (Figure 3B). Although the former observation is not surprising, this is the basis for the ubiquitous rarefaction curves. The impact of this technical bias on comparative analysis, in particular, differential abundance has not been methodically studied.

We developed a zero-inflated mixture model for metagenomic data to address this issue. We model log-transformed count data as the mixture of a point density at zero, which models technical zeros in the data due to sampling effects, and a normal distribution [4]. This allows estimates for the count distribution to not be biased by zero counts, providing robust statistics for differential abundance analysis (see Materials and Methods).

A by-product of our mixture model is a posterior probability that an observed zero-count is due to technical under-sampling. Using these posterior probabilities we were able to elucidate information about experiment design by quantifying required sampling depth to control for sampling biases for OTUs of diverse abundance (Figure 4). We believe that by providing robust estimates that are informative to the experimental process, the mixture model developed for Metastats2.0 will increase the usability of our software in clinical settings.

7 Validation

7.1 Normalization

Trivial datasets were tested and withstood the test. During this procedure it should be noted that the cumulative sum algorithm substracts machine ϵ from counts to account for some numerical issues discovered.

7.2 Expectation Maximization Algorithm

The first method of validation of the code ensured that results on a matrix of non-zero counts, the model's results and fit should coincide with a log model - $E(y_{ij}|k_j) = (b_{i0}+b_{i1}\cdot k_j+\eta_i log2(s95))$. The results should be identical - this is a byproduct of the weights coming being the relative proportion of values coming from the spike-mass distribution (for which in this case there are none). After running multiple matrices with solely positive counts all model fits were identical.

The second approach that is intended to validate the model with is to generate data using the model. We will generate OTU level datasets with 1000 features. Each feature will get .1% of the total sample counts. For the two groups, for each feature, there will be a base mean and one group will have a siginificant mean difference. That particular group will have a very low variance. Sparsity will be randomly introduced. The resulting data will be plugged into the algorithm and should show that π_j for the first group (no large k_j effect) will be approximately 1. However, for the second group, we would expect π_j to be closer to zero. The fits should show this. This is in development and we will expand on this later on.

8 Testing

8.1 Normalization

To test and compare normalization methods there is a need to quantitatively compare the normalization techniques. To compare the normalization methods we estimate false discovery rates.

Selecting a number of features f, and a number of permutations B, we compute T_{ij}^{obs} which are the pairwise feature correlation statistics (valued between -1, 1). As such, we will obtain (f choose 2) = p values.

We permute each sample's feature's counts randomly and calculate T_{ij}^{b} for $b \in \{1, ..., B\}$. Following those two steps we calculate:

1. $\hat{V}(c) = \frac{1}{B}\sum_{i=1}^{p}\sum_{b=1}^{B} I\{|T_i^b| > c\}$

2. $\hat{R}(c) = \sum_{i=1}^{p} I\{|T_i^*| > c\}$.

3. $\hat{\pi}_0 = \frac{2}{p}\sum_{i=1}^{p} I\{|T_i^*| \leq q\}$ and $q :=$ median of all permuted values, $|T_j^b|$.

From which we can calculate $F\hat{D}R(c) = \pi_0 \frac{\hat{V}(c)}{\hat{R}(c)}$

We vary c from -1 to 1.

8.2 Zero-Inflated Gaussian model

We will generate OTU level datasets with 1000 features, 50 will be "significant". One of the groups will be different from the second group for those 50 and the model will be run on this data, as well as Metastats 1.0.

9 Project Schedule + Milestones

- November 30th
  Finish code that will preprocess data, including the normalization of a dataset. I will implement as a function routine in R that will load a tab-delimited file of counts, annotation, OTU, and sample names in a convenient manner and quickly. The original Metastats has a version that runs very slowly.

  Completed:

  Several routines were written to cleanup, load, and normalize the data. Loading function now performs over 200% faster when compared to the original loading function on the 1007 sample dataset. Both normalization routines were written up. Cumulative sum distribution was written to only return the cumulative sum up to the user's requested quantile. In this manner counts could either be scaled appropriately or modeled with the E-M extensions discussed above. Cumulative distribution normalization code was implemented with two variants. The first follows the algorithm above, the second assigns to the reference the counts from samples within the bin according to the rank in the sample and the rank of the reference.

- December 15
  I will have finished the E-M algorithm, I will have a script function to pass data that was loaded in the previous step and normalized/processed and send it to the E-M code. and present a mid-year report claiming I have finished all of the zero-inflated Gaussian model and normalization codes and beginning ruminations on validation and testing.

  Completed:

  The E-M algorithm was implemented and successfully passed the first round of validation. The E-M algorithm was also run on the dysentery dataset and we show that our statistical model provides specific insight into experiment design by quantifying the effect of under-sampling on feature detection, thereby providing an answer to the question how much sequencing is enough? using the by-product of the zero-inflated mixture model, namely the posterior probability that an observed zero-count is due to technical under-sampling. The mid-year report is this file.

- January 15-February 15
  I will continue reading and work on validation of the methods, I also hope to compare the normalization methods on real datasets. I will also throughout this time begin packaging the code, commenting, etc.

  Partially finished:

  Began comparing normalization methods on the time-series data. Also, the packaging is being done. It is organized to a satisfactory level currently.

- March 15
  I will finish analyzing various datasets (those mentioned previously) and if the schedule is not delayed, I will parallelize the code (if necessary and time permitting). I also will find datasets other than those, whether from NCBI or other sources to analyze.

- May 15
  I plan to deliver the final report.

10 Deliverables

The deliverables include submission of the the R code package for Bioconductor, a final-year report, and submission of the datasets into the NCBI databases if collaborators accept. Also, an archive of results for datasets made public and published will be included if there are any at that time.

11 Figures

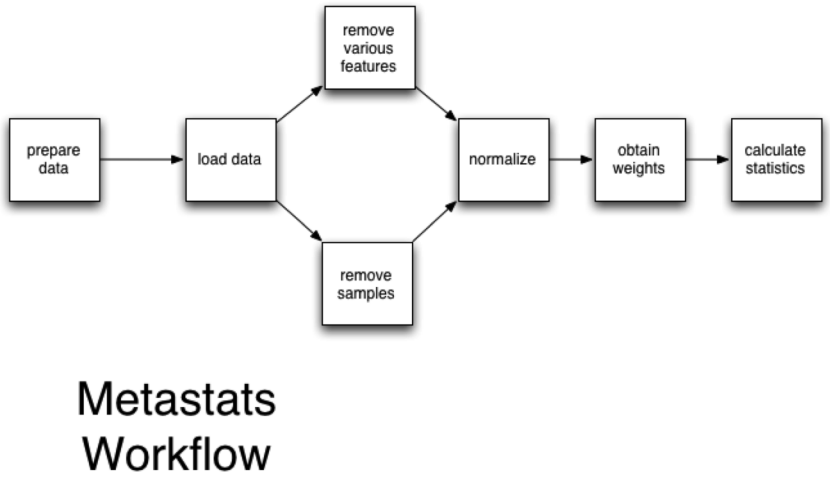
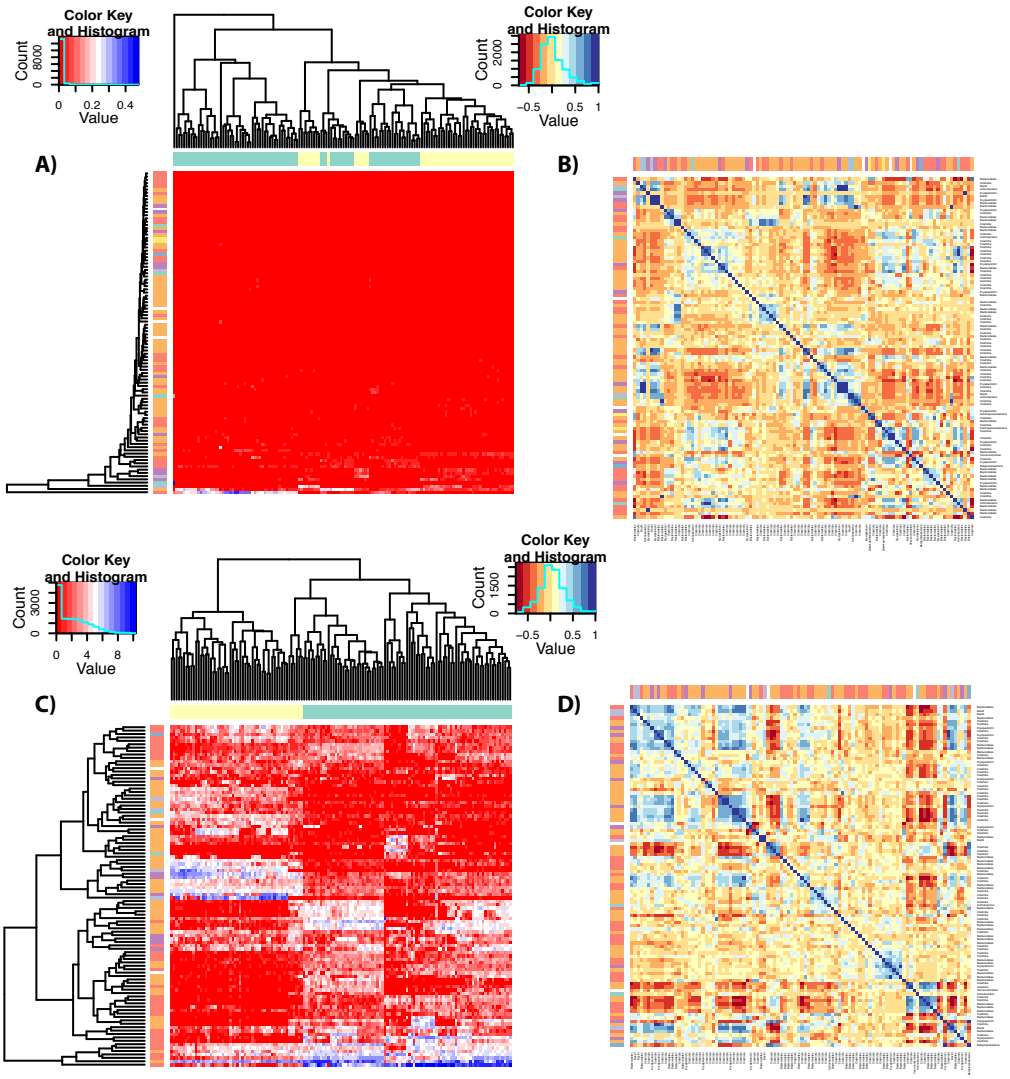


Figure 1. Metastats workflow chart. After the user prepares biological data in the proper format in tab-delimited format there are multiple scripts to load the data in to R, allow the user to remove samples or features based on their understanding of the project, normalize, and calculate proper statistics seen in Figure 1.


Color Key and Histogram
Count
Value
A)
B)
C)
D)

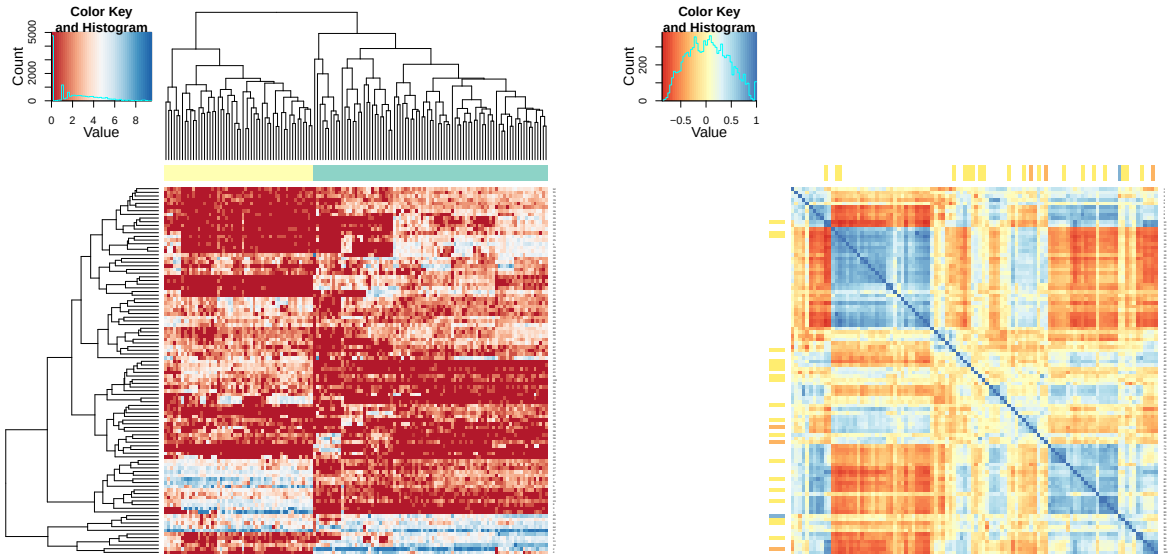


Figure 2. Effect of log-transformation and normalization on metagenomic counts. (A) Heatmap and hierarchical clustering of normalized OTU counts for the 100 OTUs with the largest overall variance in mouse diet dataset [11]. Red values indicate counts close to zero. Colors along rows indicate OTU taxonomic class, colors along the columns indicate mouse diet. Normalization uses the usual procedure of dividing each sample's OTU count by the sample's total number of reads. (B) Correlation matrix for the same OTUs from the samples on the LF-PP diet. (C,D) Heatmap of log2-transformed, cumulative sum scale normalized OT U counts andcorresponding correlation matrix. (E,F) Heatmap of log2-transformed, cumulative distribution scale normalized OT U counts andcorresponding correlation matrix. Cluster analysis was unable to correctly extract the difference in mouse diet from data normalized with the usual procedure. Furthermore, the majority of pairwise correlations between OTUs for this dataset when data was normalized in the usual way are non-zero and negative. In contrast, using log-transformed and cumulative-sum scale normalized data and cumulative distribution scale normalization, cluster analysis is able to distinguish diet differences between mice, and the distribution of pair-wise correlations is centered at zero.

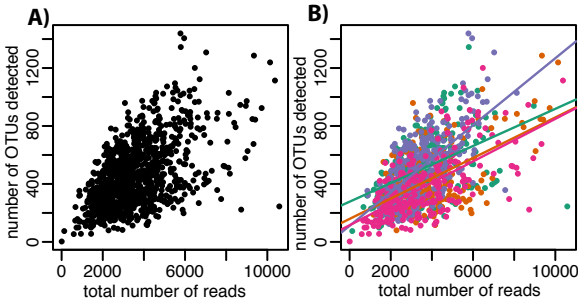


Figure 3. Effect of library size on the number of OTUs detected. (A) We plot the number of detected OTUs in a sample as a function of library size. There is a strong dependency

between library size and number of detected OTUs. (B) This relationship differs among samples collected in four different countries.

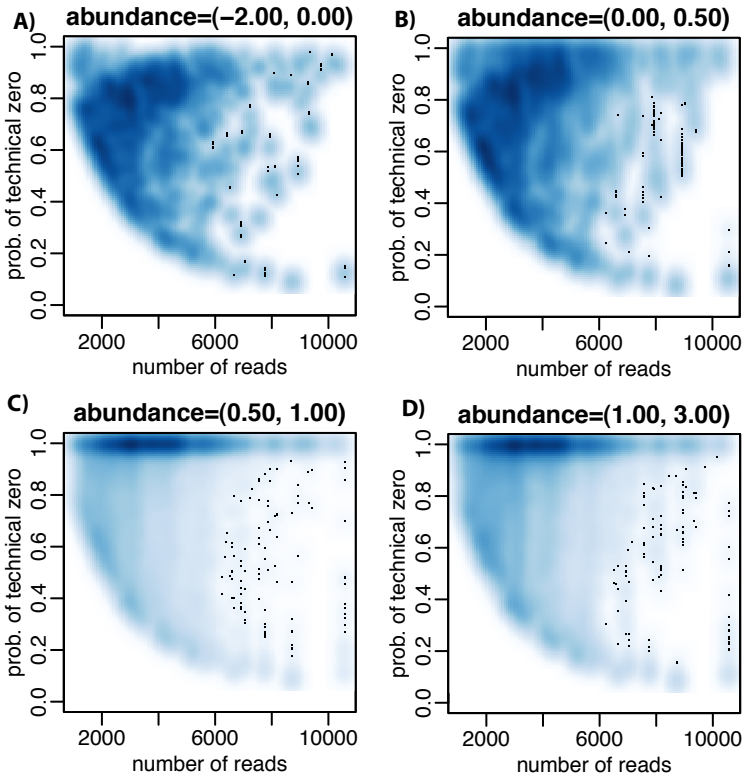


Figure 4. Using the zero-inflated model for experimental design. A by-product of our the zero-inflated mixture model is a posterior probability that an observed zero-count is due to technical under-sampling. Here we plot the estimated posterior probability as a function of library size. Each panel plots OTUs at different overall log-abundance. For low-abundance OTUs (A), it is difficult to properly estimate zeros with certainty with less than 6000 reads. On the other hand, for moderately abundant OTUs (C) and highly abundant OTUs (D), it is possible to estimate estimate zeros with certainty with libraries of size smaller that 4000 reads.

References

B M Bolstad, R A Irizarry, M Astrand, and T P Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193, 2003.

James H Bullard, Elizabeth Purdom, Kasper D Hansen, and Sandrine Dudoit. Evaluation of statistical methods for normalization and differential expression in mrna-seq experiments. *BMC Bioinformatics*, 11:94, 2010.

T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, corrected edition, July 2003.

Ben Langmead, Kasper Hansen, and Jeffrey Leek. Cloud-scale rna-sequencing differential expression analysis with myrna. *Genome Biology*, 11(8):R83, 2010.

National Academy of Science Committee on Metagenomics. The new science of metagenomics: Revealing the secrets of our microbial planet. *National Academy of Sciences*, 2007.

O. Paliy and Foy B. Mathematical modeling of 16s ribosomal dna amplification reveals optimal conditions for the interrogation of complex microbial communities with phylogenetic microarrays. *Bioinformatics*, 2011.

Karl Pearson. Mathematical contributions to the theory of evolution.– on a form of spurious correlation which may arise when indices are used in the measurement of organs. *Society*, 60:489–498, 1896.

Beltran Rodriguez-Brito, Forest Rohwer, and Robert Edwards. An application of statistics to comparative metagenomics. *BMC Bioinformatics*, 7(1):162, 2006.

Nicola Segata, Jacques Izard, Levi Waldron, Dirk Gevers, Larisa Miropolsky, Wendy S Garrett, and Curtis Huttenhower. Metagenomic biomarker discovery and explanation. *Genome biology*, 12(6):R60, June 2011.

Gordon K Smyth. *Limma: linear models for microarray data*, pages 397–420. Number October. Springer, 2005.

P.J. Turnbaugh, V.K. Ridaura, J.J. Faith, F.E. Rey, R. Knight, and J.I. Gordon. The effect of diet on the human gut microbiome: a metagenomic analysis in humanized gnotobiotic mice. *Sci. Transl. Med.*

Larry Wasserman. *All of Statistics: A Concise Course in Statistical Inference (Springer Texts in Statistics)*. Springer, December 2003.

James White, Niranjan Nagaranjan, and Mihai Pop. Statistical methods for detecting differentially abundant features in clinical metagenomic samples. *PLOS Comp Bio*, 11, 2009.