

Supplementary Information

for

GMHI-webtool: a user-friendly tool for
assessing health through metagenomic gut
microbiome profiling

Daniel Chang *et al.*

Contents

Contents	2
1 Webtool Information	3
1.1 Implementation	3
1.2 User Input	3
1.3 User Interaction and Exports	3
2 Pipeline	4
2.1 Setup	4
2.2 Repair fastq files using bmap	5
2.3 Quality check and identification of overrepresented sequences	5
2.4 Extract probable adapter sequences from FastQC outputs	5
2.5 Remove human contaminants	6
2.6 Remove adapter sequences and low quality reads	6
2.7 Profile the metagenome	6
3 α-diversity Indices	7
3.1 Richness	7
3.2 Shannon Diversity	7
3.3 Evenness	7
3.4 Simpson Diversity	7
3.5 Inverse Simpson	7
4 References	8

1 Webtool Information

1.1 Implementation

Python and the Numpy (Harris et al., 2020) library were used to pre-compute GMHI, α -diversities, and taxonomic distributions of the pooled dataset. The scikit-learn (Pedregosa et al., 2011) library was used for Principal Component Analysis (PCA) of the pooled dataset. Access the data [here](#). Regarding the front-end, GMHI-webtool is a client-side application written using JavaScript and the D3.js (Bostock, 2012) library. The Math.js (de Jong and Mansfield, 2018) library is used to project the input sample onto the first two principal components of the pooled dataset. α -diversities of the input sample, along with text parsing and validation, are implemented using JavaScript functions.

1.2 User Input

Users need to first upload (Supplementary Fig. S1A) the taxonomy profile (see Section 2). If the file has multiple samples, users can select the sample for analysis (Supplementary Fig. S1B). Users may choose to compare the input sample with healthy samples, nonhealthy samples, or all samples in the pooled gut microbiome dataset (Supplementary Fig. S1C). This selection pertains to the first two plots only. After making the proper selections, users can press the “display results” button (Supplementary Fig. S1D) to compute and visualize the analyses.

1.3 User Interaction and Exports

Users can change plot options (e.g., index, taxonomic level) by using the tabs above plots (Supplementary Fig. S2A). Additionally, users can hover their mouse over legend text (Supplementary Fig. S2B) to highlight information.

Users can export index data by clicking the “export as CSV” button (Supplementary Fig. S1E), and export plots by clicking links below them (Supplementary Fig. S2C).

Input Data

Please upload or paste your MetaPhlAn2 output file (.txt) ?

Choose File No file chosen

ID	healthy_poop	unhealthy_poop
k__Bacteria	100.0	100.0
k__Bacteria p__Actinobacteria	10.87438	1.27414
k__Bacteria p__Actinobacteria c__Actinobacteria	10.87438	1.27414

Example data

Clear

Please select sample ?

healthy_poop

Please select health status for analyses ?

All

Display results

Export as CSV

Supplementary Fig. S1. The user input panel.



Supplementary Fig. S2. Stacked bar plots of taxonomic distributions.

2 Pipeline

Prior to using GMHI-webtool, users need to profile their metagenome .fastq files using MetaPhlAn2 ([Truong et al., 2015](#)). The following is the pipeline used to preprocess and profile the stool metagenomes used to compute GMHI. Users should follow this pipeline closely to ensure GMHI is computed correctly. For brevity, this pipeline assumes that paired end metagenomes are used.

2.1 Setup

Install/update softwares:

- bmap (repair.sh) v38.93 ([Bushnell, 2014](#))
- fastqc v0.11.9 [[Source](#)]
- bowtie2 v2.4.4 ([Langmead and Salzberg, 2012](#))
- samtools v0.1.19 ([Li et al., 2009](#))
- bedtools v2.30.0 ([Quinlan and Hall, 2010](#))
- trimmomatic v0.39 ([Bolger et al., 2014](#))
- MetaPhlAn2 v2.x.x ([Truong et al., 2015](#))

```
# make sure the paired end metagenome files are available
ls
.
├── in1.fastq
└── in2.fastq

# set this var to directory containing human reference genome (use GRCh38/hg38)
# example:
$HUMAN_REFERENCE_GENOME=/users/mynamejeff/human_genomes/GRCh38_noalt_as/

# set this var to desired number of parallel processes
# example:
$N_JOBS=16

# set this var to directory containing metaphlan2 clade markers
# example:
$CLADE_MARKERS=/users/mynamejeff/metaphlan2_data/clade_markers

# Prepare adapter sequence file
```

```

echo ">PrefixPE/1
TACTACTCTTTCCCTACACGACGCTCTTCCGATCT
>PrefixPE/2
GTGACTGGAGTTCAGACGTGTGCTCTTCCGATCT" > TruSeq3-PE.fa

```

2.2 Repair fastq files using bbmap

```

repair.sh in1=in1.fastq in2=in2.fastq out1=repaired1.fastq \
out2=repaired2.fastq outs=gabage

```

2.3 Quality check and identification of overrepresented sequences

```

fastqc repaired1.fastq
fastqc repaired2.fastq
unzip repaired1_fastq.zip
unzip repaired2_fastq.zip

```

2.4 Extract probable adapter sequences from FastQC outputs

```

for f in repaired1_fastqc/fastqc_data.txt; do
    echo $f `grep -A100 ">>Overrepresented sequences" $f | \
    grep -m1 -B100 ">>END_MODULE" | \
    grep -P "Adapter|PCR" | \
    awk '{print ">overrepresented_sequences" "_" ++c "/1" $1}'` | \
    awk '{gsub(/\//1/, "/1\n")}'1' | \
    awk '{gsub(>/, "\n>")}'1' | \
    awk '{gsub(/fastqc_data.txt/, "")}'1' | \
    awk 'NF > 0';
done > adapter1.txt

for f in repaired2_fastqc/fastqc_data.txt; do
    echo $f `grep -A100 ">>Overrepresented sequences" $f | \
    grep -m1 -B100 ">>END_MODULE" | \
    grep -P "Adapter|PCR" | \
    awk '{print ">overrepresented_sequences" "_" ++c "/1" $1}'` | \
    awk '{gsub(/\//1/, "/1\n")}'1' | \
    awk '{gsub(>/, "\n>")}'1' | \
    awk '{gsub(/fastqc_data.txt/, "")}'1' | \
    awk 'NF > 0';
done > adapter2.txt

```

2.5 Remove human contaminants

```
bowtie2 -p $N_JOBS -x $HUMAN_REFERENCE_GENOME -1 repaired1.fastq \
-2 repaired2.fastq -S mapped.sam

samtools view -bS mapped.sam > mapped.bam

samtools view -b -f 12 -F 256 mapped.bam > human.bam

samtools sort -n human.bam human_sorted -@ $N_JOBS

bedtools bamtofastq -i human_sorted.bam -fq human1.fastq -fq2 human2.fastq
```

2.6 Remove adapter sequences and low quality reads

```
cat adapter1.txt adapter2.txt TruSeq3-PE.fa > adapters.txt

trimmomatic PE -threads $N_JOBS human1.fastq human2.fastq -baseout QC.fastq.gz \
ILLUMINACLIP:adapters.txt:2:30:10:2:keepBothReads LEADING:3 TRAILING:3 MINLEN:60
```

2.7 Profile the metagenome

```
metaphlan2.py QC_1P.fastq.gz,QC_2P.fastq.gz --bowtie2db $CLADE_MARKERS \
--bowtie2out --index mpa_v20_m200 --nproc $N_JOBS --input_type fastq \
-o profiled_metagenome.txt
```

After running the pipeline, users can upload the taxonomic profile “profiled_metagenome.txt” to GMHI-webtool. Users can also use MetaPhlAn2 to merge multiple taxonomic profiles:

```
ls
.
├── profiled_metagenome1.txt
├── profiled_metagenome2.txt
└── profiled_metagenome3.txt

merge_metaphlan_tables.py profiled_metagenome*.txt > merged_abundance_table.txt
```

And upload the merged file “merged_abundance_table.txt” to GMHI-webtool.

3 α -diversity Indices

GMHI-webtool computes a number of α -diversities from stool metagenome samples. Let p_i be the relative abundance of the i th species in the sample. For consistency with the original GMHI work, only the 313 species considered during the computation of GMHI were considered ([Gupta et al., 2020](#)). Let $S = 313$ be the maximum number of species in a single sample. Let $c = 0.00001$ be the presence threshold.

3.1 Richness

Richness R is the number of species with relative abundance greater than the presence threshold.

$$R = |\{ i \mid p_i > c \}|$$

3.2 Shannon Diversity

Shannon Diversity H' is derived from Shannon entropy, and is a measure of the uncertainty associated with predicting the species of any microbe in the sample.

$$H' = - \sum_{\forall i[p_i > 0]} p_i \ln(p_i)$$

3.3 Evenness

Evenness E is a measure of close in number (or relative abundance) different species are.

$$E = \frac{H'}{\ln(S)}$$

3.4 Simpson Diversity

Simpson diversity is equivalent to the probability that two randomly selected microbes are of the same species ([SIMPSON, 1949](#)).

$$\lambda = \sum_{\forall i[p_i > 0]} \ln(p_i)$$

3.5 Inverse Simpson

Inverse Simpson diversity is the reciprocal of Simpson diversity.

$$I = \frac{1}{\lambda}$$

4 References

- Bolger, A. M., Lohse, M., and Usadel, B. (2014). Trimmomatic: a flexible trimmer for illumina sequence data. *Bioinformatics*, 30(15):2114–2120.
- Bostock, M. (2012). D3.js - data-driven documents.
- Bushnell, B. (2014). Bbmap: A fast, accurate, splice-aware aligner. 0(0).
- de Jong, J. and Mansfield, E. (2018). Math.js: An advanced mathematics library for JavaScript. *Computing in Science & Engineering*, 20(1):20–32.
- Gupta, V. K., Kim, M., Bakshi, U., Cunningham, K. Y., Davis, J. M., Lazaridis, K. N., Nelson, H., Chia, N., and Sung, J. (2020). A predictive index for health status using species-level gut microbiome profiling. *Nature Communications*, 11(1).
- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825):357–362.
- Langmead, B. and Salzberg, S. L. (2012). Fast gapped-read alignment with bowtie 2. *Nature Methods*, 9(4):357–359.
- Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., and and, R. D. (2009). The sequence alignment/map format and SAMtools. *Bioinformatics*, 25(16):2078–2079.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- Quinlan, A. R. and Hall, I. M. (2010). BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, 26(6):841–842.
- SIMPSON, E. H. (1949). Measurement of diversity. *Nature*, 163(4148):688–688.
- Truong, D. T., Franzosa, E. A., Tickle, T. L., Scholz, M., Weingart, G., Pasolli, E., Tett, A., Huttenhower, C., and Segata, N. (2015). MetaPhlAn2 for enhanced metagenomic taxonomic profiling. *Nature Methods*, 12(10):902–903.