Qiime Community Profiling
University of Colorado at Boulder

Authors: Antonio González, Jose C. Clemente, Dan Knights, Rob Knight
Version: 1.0
Effective Date:

1 Abstract

2 Introduction

This SOP describes QIIME (Quantitative Insights Into Microbial Ecology) for community profiling using the Human Microbiome Project 16S data. The process takes users from their raw sequencing output through initial analyses such as OTU picking, taxonomic assignment, and construction of phylogenetic trees from representative sequences of OTUs, and through downstream statistical analysis, visualization, and production of publication-quality graphics.

3 Requirements

3.1 Data requirements

- Sequencing data (.sff files available at http://hmpdacc.org/HMR16S/)
- Metadata mapping files (.lmd files available at http://hmpdacc.org/HMR16S/)
- Greengenes OTUs (available at http://qiime.sourceforge.net/home_static/dataFiles.html)

3.2 Software requirements

- QIIME (available for download from http://qiime.org)
- Additional custom scripts available at **hmpdacc.org/HMQCP**

4 Procedure

4.1 Get files and split libraries

4.1.1 Get all files (sff and mapping files) from http://hmpdacc.org/HMR16S/

4.1.2 Run sffinfo on all the sff files with -notrim to get all the barcodes

- echo "sffinfo -notrim $FOLDER/$line > $F_OUT/$line.txt"
- echo "sffinfo -notrim -q $FOLDER/$line > $F_OUT/$line.qual"
- echo "sffinfo -notrim -s $FOLDER/$line > $F_OUT/$line.fna"

  Where $FOLDER is the name of the folder that hosts the sff files, $line is the name of each individual sff, $F_OUT is the name of the output folder.

4.1.3 clean_fasta.py from PrimerProspector (http://pprospector.sourceforge.net/) to make the fasta files compatible with QIIME

4.1.4 Use split_metadata.py, a custom script available at **hmpdacc.org/HMQCP** that relies on the mapping file to create ad-hoc split_libraries.py commands per sffs as the barcodes have variable length (-b option) and the start index should not overlap with other runs. The main

parameters for the split library commands are the default QIIME parameters for release 1.3.0:

-r, --remove_unassigned
DEPRECATED: pass --retain_unassigned_reads to keep unassigned reads [default: none]

-l MIN_SEQ_LEN, --min-seq-length=MIN_SEQ_LEN
minimum sequence length, in nucleotides [default: 200]

-L MAX_SEQ_LEN, --max-seq-length=MAX_SEQ_LEN
maximum sequence length, in nucleotides [default: 1000]

-s MIN_QUAL_SCORE, --min-qual-score=MIN_QUAL_SCORE
min average qual score allowed in read [default: 25]

-a MAX_AMBIG, --max-ambig=MAX_AMBIG
maximum number of ambiguous bases [default: 0]

-H MAX_HOMOPOLYMER, --max-homopolymer=MAX_HOMOPOLYMER
maximum length of homopolymer run [default: 6]

-M MAX_PRIMER_MM, --max-primer-mismatch=MAX_PRIMER_MM
maximum number of primer mismatches [default: 0]

-b BARCODE_TYPE, --barcode-type=BARCODE_TYPE
barcode type, hamming_8, golay_12, variable_length (will disable any barcode correction if variable_length set), or a number representing the length of the barcode, such as -b 4. [default: golay_12] ***** This was modified per sff

-e MAX_BC_ERRORS, --max-barcode-errors=MAX_BC_ERRORS
maximum number of errors in barcode [default: 1.5]

-n START_INDEX, --start-numbering-at=START_INDEX
seq id to use for the first sequence [default: 1]

-w QUAL_SCORE_WINDOW, --qual_score_window=QUAL_SCORE_WINDOW
Enable sliding window test of quality scores. If the average score of a continuous set of w nucleotides falls below the threshold (see -s for default), the sequence is discarded. A good value would be 50. 0 (zero) means no filtering. Must pass a .qual file (see -q parameter) if this functionality is enabled. Default behavior for this function is to truncate the sequence at the beginning of the poor quality windowmand test for minimal length (-l parameter) of the resulting sequence. [default: 0]

4.1.5 Run output from previous step.

4.1.6 Concatenate all the fasta files per region (V13 and V35) using cat:

- cat pp*_split/v13/*/seqs.fna > seqs_v13.fna
- cat pp*_split/v35/*/seqs.fna > seqs_v13.fna

Note: If you plan to combine results for V13-V35 you will need to rename some of the sample names in pp1_v13 & pp1_v35, and pps_v13 & pps_v35 because they have duplicated names.

Files available at hmpdacc.org/HMQCP:

1. split_metadata.py
2. Sequences:

seqs_v13.fna
seqs_v35.fna

4.2 Pick OTU using OTUpipe

Note: This description is mostly for v13 (v35 is similar).

4.2.1 Input sequences obtained from step 4.1.6.

seqs_v13.fna
seqs_v35.fna

4.2.2 Chimera check, ref-based

4.2.2.1 Chimera check ref-based in each subdirectory

*Directory: v13/*chimera_ref_split

```
usearch --uchime seqs_v13.fna --db $UCHIME_REFDB --rev --uchimeout
refdb.uchime --chimeras chimeras_ref.fa --nonchimeras
non_chimera_refdb.fa --log uchimedb.log
```

4.2.2.2 Concatenate outputs of chimera checking

Output from v13/chimera checking

Directory: chimera_ref_split

```
md5sum non_chimera_refdb.fa
                    1b015bcf145d6ce56e06b90cc1b6b97b non_chimera_refdb.fa
```

For v35:

```
md5sum non_chimera_refdb.fa
                     6f0ac625d99067465267147dd40f7fdc non_chimera_refdb.fa
```

Files available at hmpdacc.org/HMQCP:

1. Chimeric sequences (all chimeric reads that are excluded from further analysis, as a resource for those who want to do trash/treasure hunting)
   chimeric_v13.fna
   chimeric_v35.fna
2. Non-chimeric sequences (non_chimera_refdb.fa)
   non_chimeric_v13.fna
   non_chimeric_v35.fna

4.2.3 OTU picking

4.2.3.1 Sort non-chimeric sequences by abundance

Directory: v13/chimera_ref_split

```
usearch --sortsize non_chimera_refdb.fa --out put tmp2.fas --
minsize 4 --log sort2.log
```

4.2.3.2 Cluster (OTU picking) at 97%

Directory: v13/chimera_ref_split

```
usearch --cluster tmp2.fas --id 0.97 --uc otu1.uc --seedsout
tmp3.fa --w 20 --slots 16769023 --log cluster2.log --sizein --
sizeout --maxrejects 64 –usersort
```

4.2.3.3 Assign sequential numbers to OTUs

Directory: v13/chimera_ref_split

```
fasta_number.py tmp3.fa OTU_97 > otus.fa
```

4.2.3.4 Assign reads to OTUs

Directory: v13/chimera_ref_split

```
usearch --query seqs.fna --db otus.fa --logbin.log --id 0.97 --
global --ucread2otu.uc
```

4.2.3.5 Make read map file

Directory: v13/chimera_ref_split

```
python uc2qiime.pyread2otu.uc > otu2reads.txt
```

Output from OTU picking

Directory: v13/chimera_ref_split

```
md5sum otu2reads.txt
```

43471acc802c95dfa0807e002ac5e10f otu2reads.txt

For v35:

```
md5sum otu2reads.txt
```

d017f081e680fb5a7041888a26d8bc22 otu2reads.txt

Files available at hmpdacc.org/HMQCP:

1. Map file between OTU clusters and sequences
   otu2reads_v13.txt
   otu2reads_v35.txt

4.3 Construct OTU table and Assign Taxonomy

4.3.1 Modify otupipe OTU map file to remove everything except the seq id

Directory: v13/chimera_ref_split

perl clean_otu2reads.pl otu2reads.txt > otu2reads.clean.txt

4.3.2 Pick the rep set

Directory: v13/chimera_ref_split

```
pick_rep_set.py –i otu2reads.clean.txt –f seqs_v13.fna –o
seqs_rep_set.fna
```

4.3.3 Assign taxonomy with RDP 2.2 (re-trained with greengenes)

Directory: v13/chimera_ref_split

```
assign_taxonomy.py -i seqs_rep_set.fna -r
gg_97_otus_4feb2011.fasta -t  greengenes_tax_rdp_train.txt -o
rdp_assigned_taxonomy/
```

4.3.4 Create OTU table

Directory: v13/chimera_ref_split

```
make_otu_table.py -i otu2reads.clean.txt –t
rdp_assigned_taxonomy/seqs_rep_set_tax_assignments.txt –o
otupipe.chimeraref.otutable.txt
```

Output from OTU table construction and taxonomy assignment

Directory: v13/chimera_ref_split

```
md5sum otupipe.chimeraref.otutable.txt
```

fbfe3931860c7647729b61aa7170ac26 otupipe.chimeraref.otutable.txt

Directory: v35/chimera_ref_split

```
md5sum otupipe.chimeraref.otutable.txt
```

7ff1fc09c8240023205493244c127954 otupipe.chimeraref.otutable.txt

Files available at hmpdacc.org/HMQCP:

1. Representative sequence sets (seqs_rep_set.fna)
   - rep_set_v13.fna
   - rep_set_v35.fna
2. OTU table per sample (Un-merged OTU-table w/ tax, otupipe.chimeraref.otutable.txt)
   - otu_table_v13.txt
   - otu_table_v35.txt

4.4 Detect mislabeling and contamination

4.4.1 Add new 'Env' column to mapping file with four categories: Oral (includes throat), Skin (includes nares), Stool, Vagina. Starting data:

Original OTU tables:
- otupipe.chimeraref.otutable.v13.txt
- otupipe.chimeraref.otutable.v35.txt

Mapping files:
- v13_map_modified_4sl.txt
- v35_map_modified_4sl.txt

4.4.2 Rarefy OTU tables at depth 100.

```
single_rarefaction.py –I otupipe.chimeraref.otutable.v13.txt -d 100
-o otupipe.chimeraref.otutable.v13_rare100.txt
single_rarefaction.py -i otupipe.chimeraref.otutable.v35.txt -d 100
-o otupipe.chimeraref.otutable.v35_rare100.txt
```

4.4.3 Drop all OTUs present in less than 1% of the samples:

```
filter_otu_table.py -i otupipe.chimeraref.otutable.v13_rare100.txt
-s 37 -o otupipe.chimeraref.otutable.v13_rare100_s37.txt

filter_otu_table.py -i otupipe.chimeraref.otutable.v35_rare100.txt
-s 60 -o otupipe.chimeraref.otutable.v35_rare100_s60.txt
```

4.4.4 Estimate probability of mislabeling using random forests classifier

```
supervised_learning.py -v -i
otupipe.chimeraref.otutable.v13_rare100_s37.txt -m
v13_map_modified_4sl.txt -c Env -o sup_learning_s37 -f

supervised_learning.py -v -i
otupipe.chimeraref.otutable.v35_rare100_s60.txt -m
v35_map_modified_4sl.txt -c Env -o sup_learning_s60 -f
```

4.4.5 Create copy of mapping file with mislabeled samples (those with probability of the alleged label < 25%) removed:

```
remove_mislabels.py  v13/random_forest/mislabeling.txt
v13_map_modified_4sl.txt .25 >
v13_map_modified_4sl_nomislabels_s37.txt

remove_mislabels.py  v35/random_forest/mislabeling.txt
v35_map_modified_4sl.txt .25 >
v35_map_modified_4sl_nomislabels_s60.txt
```

4.4.6 Reduce number of features further before running SourceTracker (to reduce run-time): Remove OTUs present in <1% of the samples, then rarefy at depth 100, then again remove OTUs present in <1% of the remaining samples:

```
filter_otu_table.py -i otupipe.chimeraref.otutable.v13.txt -s 41 -o
otupipe.chimeraref.otutable.v13_s41.txt

filter_otu_table.py -i otupipe.chimeraref.otutable.v35.txt -s 69 -o
otupipe.chimeraref.otutable.v35_s69.txt

single_rarefaction.py -i otupipe.chimeraref.otutable.v13_s41.txt -d
100 -o otupipe.chimeraref.otutable.v13_s41_rare100.txt

single_rarefaction.py -i otupipe.chimeraref.otutable.v35_s69.txt -d
100 -o otupipe.chimeraref.otutable.v35_s69_rare100.txt

filter_otu_table.py -i
otupipe.chimeraref.otutable.v13_s41_rare100.txt -s 37 -o
otupipe.chimeraref.otutable.v13_s41_rare100_s37.txt
```

```
filter_otu_table.py -i
otupipe.chimeraref.otutable.v35_s60_rare100.txt -s 60 -o
otupipe.chimeraref.otutable.v35_s69_rare100_s60.txt
```

4.4.7 Estimate contamination levels for all remaining samples using SourceTracker

```
R --slave --vanilla --args  -i
otupipe.chimeraref.otutable.v35_s69_rare100_s60.txt -m
v35_map_modified_4sl_nomislabels_s60.txt -o sourcetracker_s69_s60 -
s < sourcetracker_for_QIIME.r

R --slave --vanilla --args  -i
otupipe.chimeraref.otutable.v13_s41_rare100_s37.txt -m
v13_map_modified_4sl_nomislabels_s37.txt -o sourcetracker_s41_s37 -
s < sourcetracker_for_QIIME.r
```

4.4.8 Merge sourcetracker output with original mapping file, then add mislabeling estimates:

```
R --slave --args v13_map_modified_4sl.txt map.txt
source_predictions.txt source_predictions_stdev.txt <
merge_sourcetracker_output_with_mapping_file.r

R --slave --args v35_map_modified_4sl.txt map.txt
source_predictions.txt source_predictions_stdev.txt <
merge_sourcetracker_output_with_mapping_file.r
```

4.4.9 Add 'Mislabeled' column to final mapping file, with 'NA' where mislabeling was not estimated (samples with < 100 sequences), 'TRUE' when the estimated probability of the alleged label was < .25, 'FALSE' otherwise. Add 'Contaminated' column with 'NA' when 'Mislabeled' is 'TRUE' or 'NA', 'TRUE' when 'Mislabeled' is FALSE and 'Max_Proportion_this_Env' < .6, and 'FALSE' otherwise.

```
R --slave --args v13_map_modified_4sl_st_ml_s41_s37.txt
v13_map_modified_4sl_st_ml_s41_s37_mislabeled_contaminated_.25_.6.t
xt .25 .6 < choose_mislabeling_contamination_thresholds.r

R --slave --args v35_map_modified_4sl_st_ml_s69_s60.txt
v35_map_modified_4sl_st_ml_s69_s60_mislabeled_contaminated_.25_.6.t
xt .25 .6 < choose_mislabeling_contamination_thresholds.r
```

4.4.10 Filter original OTU table using final mapping file to exclude mislabeled and contaminated samples, but to retain those samples for which estimation was not performed (samples with < 100 sequences):

```
filter_by_metadata.py -i otupipe.chimeraref.otutable.v13.txt -m
v13_map_modified_4sl_st_ml_s41_s37_mislabeled_contaminated_.25_.6.t
xt -s 'Mislabeled:FALSE,NA;Contaminated:FALSE,NA' -o
otupipe.chimeraref.otutable.v13_s41_no_mislabeled_contaminated_.25_
.6.txt -p tmp.txt

filter_by_metadata.py -i otupipe.chimeraref.otutable.v35.txt -m
v35_map_modified_4sl_st_ml_s69_s60_mislabeled_contaminated_.25_.6.t
xt -s 'Mislabeled:FALSE,NA;Contaminated:FALSE,NA' -o
otupipe.chimeraref.otutable.v35_s69_no_mislabeled_contaminated_.25_
.6.txt -p tmp.txt
```

Files available at hmpdacc.org/HMQCP:

1. Mislabeled or contaminated samples
   mislabeled_contaminated_list_v13.txt
   mislabeled_contaminated_list_v35.txt
2. Final OTU table (filtered & merged, by PSN#)
   otu_table_psn_v13.txt
   otu_table_psn_v35.txt

4.5 Summarize and Conduct Beta Diversity Analysis

4.5.1 Summarize the OTU by PSN to concatenate multiple runs of the same samples.

```
a.summarize_otu_by_cat.py -i v13_map.txt -c otu_table[from
  last step] -m PSN -o v13_psn_otu.txt
b.summarize_otu_by_cat.py -i v35_map.txt -c otu_table[from
  last step] -m PSN -o v35_psn_otu.txt
```

4.5.2 Beta diversity pipeline using 150 processors

a. beta_diversity_through_plots.py -i v13_psn_otu.txt -m v13_map_uniquebyPSN.txt --suppress_2d_plots --suppress_distance_histograms -t rep_phylo_v13.tre -o beta13 -e 1000 -a -O 150 -p qiime_parameters.txt
b. beta_diversity_through_plots.py -i v35_psn_otu.txt -m v35_map_uniquebyPSN.txt --suppress_2d_plots --suppress_distance_histograms -t rep_phylo_v35.tre -o beta35 -e 1000 -a -O 150 -p qiime_parameters.txt

Files available at hmpdacc.org/HMQCP:

1. Mappings files by Parent Sample Number (PSN)
2. Beta diversity analysis
   beta_diversity_v13/
   beta_diversity_v35/

4.6 Procrustes WGS/16S

4.6.1 Format KEGG module abundance files available for download at http://www.hmpdacc.org/HMMRC/ to look like OTU tables, using custom cleaning scripts available at **hmpdacc.org/HMQCP**:

a. clean_bugs.py
b. clean_ecs.py
c. clean_kegg.py

4.6.2 Summarize the resulting "OTU tables" by PSN

a. summarize_otu_by_cat.py -i bugs_map.txt -c bugs_fixed.txt -m PSN -o bugs_psn_otu_table.txt
b. summarize_otu_by_cat.py -i ecs_map.txt -c ecs_fixed.txt -m PSN -o ecs_psn_otu_table.txt
c. summarize_otu_by_cat.py -i kegg_map.txt -c kegg_fixed.txt -m PSN -o kegg_psn_otu_table.txt

4.6.3 Calculate beta diversity using Bray Curtis for each file:

a. beta_diversity.py -i bugs_psn_otu_table.txt -m bray_curtis -o dist_mtx
b. beta_diversity.py -i ecs_psn_otu_table.txt -m bray_curtis -o dist_mtx
c. beta_diversity.py -i kegg_psn_otu_table.txt -m bray_curtis -o dist_mtx

4.6.4 PCoA for each file

a. principal_coordinates.py -i dist_mtx/bray_curtis_bugs_psn_otu_table.txt -o dist_mtx/bugs_bray_curtis_pc.txt
b. principal_coordinates.py -i dist_mtx/bray_curtis_ecs_psn_otu_table.txt -o dist_mtx/ecs_bray_curtis_pc.txt
c. principal_coordinates.py -i dist_mtx/bray_curtis_kegg_modules_abundance_fixed.txt -

4.6.5 Transform PCoAs for WGS and 16S and do 1000 Monte Carlo iteration for statistical value:

a. transform_coordinate_matrices.py -i ../beta13/unweighted_unifrac_pc.txt,dist_mtx/bugs_bray_curtis_pc.txt -o procrustes_mc_v13_bugs/ -r 1000
b. transform_coordinate_matrices.py -i ../beta13/unweighted_unifrac_pc.txt,dist_mtx/ecs_bray_curtis_pc.txt -o procrustes_mc_v13_ecs/ -r 1000
c. transform_coordinate_matrices.py -i ../beta13/unweighted_unifrac_pc.txt,dist_mtx/kegg_modules_abundance_pc.txt -o procrustes_mc_v13_kegg/ -r 1000
d. transform_coordinate_matrices.py -i ../beta13/unweighted_unifrac_pc.txt,dist_mtx/bugs_bray_curtis_pc.txt -o procrustes_mc_v35_bugs/ -r 1000
e. transform_coordinate_matrices.py -i ../beta13/unweighted_unifrac_pc.txt,dist_mtx/ecs_bray_curtis_pc.txt -o procrustes_mc_v35_ecs/ -r 1000
f. transform_coordinate_matrices.py -i ../beta13/unweighted_unifrac_pc.txt,dist_mtx/kegg_modules_abundance_pc.txt -o procrustes_mc_v35_kegg/ -r 1000

4.6.6 Compare and create plots:

a. compare_3d_plots.py -i procrustes_mc_v13_bugs/pc1_transformed.txt,procrustes_mc_v13_bugs/pc2_transformed.txt -m ../v13_map_uniquebyPSN.txt -o procrustes_mc_v13_bugs/plots/ -b sex,HMPbodysubsite,HMPbodysupersite
b. compare_3d_plots.py -i procrustes_mc_v13_ecs/pc1_transformed.txt,procrustes_mc_v13_ecs/pc2_transformed.txt -m ../v13_map_uniquebyPSN.txt -o procrustes_mc_v13_ecs/plots/ -b sex,HMPbodysubsite,HMPbodysupersite
c. compare_3d_plots.py -i procrustes_mc_v35_bugs/pc1_transformed.txt,procrustes_mc_v35_bugs/pc2_transformed.txt -m ../v35_map_uniquebyPSN.txt -o procrustes_mc_v35_bugs/plots/ -b sex,HMPbodysubsite,HMPbodysupersite
d. compare_3d_plots.py -i procrustes_mc_v35_ecs/pc1_transformed.txt,procrustes_mc_v35_ecs/pc2_transformed.txt -m ../v35_map_uniquebyPSN.txt -o procrustes_mc_v35_ecs/plots/ -b sex,HMPbodysubsite,HMPbodysupersite
e. compare_3d_plots.py -i procrustes_mc_v13_kegg/pc1_transformed.txt,procrustes_mc_v13_kegg/pc2_transformed.txt -m ../v13_map_uniquebyPSN.txt -o procrustes_mc_v13_kegg/plots
f. compare_3d_plots.py -i procrustes_mc_v35_kegg/pc1_transformed.txt,procrustes_mc_v35_kegg/pc2_transformed.txt -m ../v35_map_uniquebyPSN.txt -o procrustes_mc_v35_kegg/plots

Files available at hmpdacc.org/HMQCP:

1. Cleaning scripts
   cleaning_wgs/
3. WGS distance matrices and Procrustes plots
   procrustes_wgs_16s/

4.7 Procrustes per sub site

4.7.1 Added a new column to the mapping file that is the combination from visitno & RSID (visitno_RSID), then use a custom script, available a **hmpdacc.org/HMQCP**, to split the mapping file into 3 files:
(1) a list of the combination of subsites to compare, where the first column is the sampleID and second is visitno_RSID, i.e. Anterior_nares.Attached_Keratinized_gingiva_map.txt ;
(2) a list of the samples to keep per subsite, i.e Anterior_nares_ids.txt; and
(3) a mapping file for make_3d_plots.py of the first member of the combination, i.e for Anterior_nares. Attached_Keratinized_gingiva I only create the mapping file for Anterior_nares (mapping_Anterior_nares.txt)

a. ./split.py -m v13_map_uniquebyPSN.txt -f HMPbodysubsite -c visitno_RSID -o files4split_v13
b. ./split.py -m v35_map_uniquebyPSN.txt -f HMPbodysubsite -c visitno_RSID -o files4split_v35

4.7.2 Filter the original PCoA file from unweighted UniFrac to get individual PCoA files per subsite

a. filter_distance_matrix.py -i unweighted_unifrac_dm_v13.txt -s files4split_v13/[subsite]_ids.txt -o dm_v13/[subsite].txt
b. filter_distance_matrix.py -i unweighted_unifrac_dm_v35.txt -s files4split_v35/[subsite]_ids.txt -o dm_v13/[subsite].txt

4.7.3 Transform the PCoA per combination and do 1000 Monte Carlo iterations

a. transform_coordinate_matrices.py -o trans_v35/[subsite1].[subsite2] -r 1000 -i pcoa_v35/pcoa_[subsite1].txt,pcoa_v35/pcoa_[subsite2].txt -s files4split_v35/[subsite1].[subsite2]_map.txt
b. transform_coordinate_matrices.py -o trans_v35/[subsite1].[subsite2] -r 1000 -i pcoa_v35/pcoa_[subsite1].txt,pcoa_v35/pcoa_[subsite2].txt -s files4split_v35/[subsite1].[subsite2]_map.txt

4.7.4 Create plots for the previous results

a. compare_3d_plots.py -i trans_v13/[subsite1].[subsite2]/pc1_transformed.txt,trans_v13/[subsite1].[subsite2]/pc2_transformed.txt -m files4split_v13_test/mapping_[subsite1].txt -o plots_v13/[subsite1].[subsite2]/
b. compare_3d_plots.py -i trans_v35/[subsite1].[subsite2]/pc1_transformed.txt,trans_v35/[subsite1].[subsite2]/pc2_transformed.txt -m files4split_v35_test/mapping_[subsite1].txt -o plots_v35/[subsite1].[subsite2]/

Files available at hmpdacc.org/HMQCP:

1. split.py
2. filter_distance_matrix.py
3. Procrustes results per body site with plots
   procrustes_per_site/

5 Implementation

6 Discussion

7 Related Documents & References

Caporaso JS, Kuczynski J, Stombaugh J, Bittinger K, Bushman FD, Costello EK *et al*. QIIME allows integration and analysis of high-throughput community sequencing data. *Nature Methods **7**, 335-336 (2010).*

Authors: Antonio González, Jose C. Clemente, Dan Knights, Rob Knight
Version: 1.0
Effective Date:

8 Revision History

Version	Author/Reviewer	Date	Change Made
1.0		11/1/2011	Establish SOP