Bilbao estuary, 16S analysis protocol

Mikel Aguirre Rodrigo

Abstract

Citation: Mikel Aguirre Rodrigo Bilbao estuary, 16S analysis protocol. **protocols.io** dx.doi.org/10.17504/protocols.io.htmb6k6

Published: 17 May 2017

Protocol

Sample collection

Step 1.

Tributary samples were collected during the months of April, August and October 2014. A total of 12 samples (including two replicates) were collected in the last stretch of each tributary and from fixed points (primarily at bridges), avoiding areas affected by the tide. In the case of the Galindo River, the sampling station was 5 meters from the outlet of the WWTP.

For the estuary samples, collection was carried out monthly from August 2013 to October 2014. In total, 171 samples (including two replicates) were collected for the 14-month period. Sampling took place only on days of neap tide coefficient (30-50), always at high tide, and at approximately the same time of day (10:00 A.M.-12:00 P.M.) to eliminate confounding variables. Salinity gradient points of 30, 33 and 35 ppt were localized along the estuary each month. Once the water mass stabilized, samples were collected at a middle depth (>3 m), below the halocline (B30, B33, B35), and at the upper layer of each euhaline water mass (surface samples: IS, MS, OS, respectively).

Preprocesing of the samples

Step 2.

Samples were collected using an oceanographic Niskin bottle. The water (10 L approx.) was stored in opaque plastic jerry cans in the field. Once in the laboratory, the water was filtered (5 L approx.) through 20 µm Nylon net filters (Millipore, 90 mm diameter) and bacteria were collected with 0.22 µm Durapore® membrane filters (Millipore, 47 mm diameter). Filtration was performed in triplicate using a Kitasato Flask and a vacuum pump. The whole process, from sampling to storage, took less than 3 hours to perform. All filters were store at -80 ºC until DNA extraction.

DURATION

03:00:00

Physical and chemical variables measurement

Step 3.

At each sampling point vertical profiles (every 0.5 m) of salinity, temperature, pH, and dissolved oxygen (DO) saturation (%) were obtained in situ using a YSI 556 MPS Multiparameter Probe. Water transparency was measured with a Secchi Disk. Chl-a concentrations were calculated from spectrophotometric measurements on acetone extracts using a monochromatic method with acidification. In addition, precipitation data was obtained through the Hydrometeorology Service of the Regional Council of Bizkaia.

LINK:
http://www.bizkaia.eus/Ingurugiroa_Lurraldea/Hidrologia_Ac/Datos_meteo.asp?Idioma=CA&Tem_Codigo=2679

DNA extraction

Step 4.

Complete genomic DNA was extracted from the half of the 0.22 µm Durapore® membrane filters using PowerSoil DNA isolation kit (Mo Bio laboratories, Inc., Carlsbad, CA, USA) following the manufacturer protocol. The DNA quantity and quality of each sample was assessed by either a ND-1000 spectrophotometer (NanoDrop) or Qubit fluorimeter (Life technologies). To avoid cross-contamination all tools were flame-sterilized between samples and lab surfaces were decontaminated with DNA-ExitusPlus (Applychem) after each session. Finally, the DNA extractions were stored at -20 ºC until DNA sequencing.

REAGENTS

PowerSoil DNA Isolation Kit 12888-100 by Mobio

DNA-ExitusPlus A7089 by Applychem

16S rRNA gene amplification and sequencing

Step 5.

The 16S rRNA samples were amplified and sequenced by the Next Generation Sequencing Core at Argonne National Laboratory, Lemont, IL (USA) (http://www.earthmicrobiome.org/). Earth Microbiome Project's protocols were followed for the amplification and sequencing of the community 16S v4 region by using 515f and 806r primers that contained 12 bp barcodes for sequencing. The sequencing was carried out in two MiSeq runs (2x150 paired-end). The data is available in the QIITA portal (ID 10470) and on the ENA database (study: PRJEB14094).

LINK:
http://press.igsb.anl.gov/earthmicrobiome/protocols-and-standards/16s/

Sequence trimming

Step 6.

The raw sequences were trimmed using Sickle tool (v1.33) with default parameters (including Phred score ≥ 20).

SOFTWARE PACKAGE (Linux)

Sickle, 1.33

Joshi NA
https://github.com/najoshi/sickle

cmd COMMAND

```
sickle pe -f Undetermined_S0_L001_R1_001.fastq -r Undetermined_S0_L001_R2_001.fastq -
t sanger -o trim_16_1.fastq -p trim_16_2.fastq -s trim_16_f.fastq -q 20
```

Sequence trimming using sickle software. Link: https://github.com/najoshi/sickle

Merge Paired-end sequences

Step 7.

Pear software (v0.9.6) was used to merge Illumina paired-end reads, using a cut-off of 0.01 (P-value) for the observed expected alignment score.

SOFTWARE PACKAGE (Linux)

Pear, 0.9.6

Jiajie Zhang
https://github.com/xflouris/PEAR

cmd COMMAND

```
pear -f trim_16_1.fastq -r trim_16_2.fastq -o trim_out_pear -j 20 -v 15 -q 20
```

Merge the paired-end sequences using Pear software. link: https://github.com/xflouris/PEAR

Remove non-existent barcodes

Step 8.

Next, we utilized fastq-barcode.pl to remove non-existent barcodes from the fastq achieved by Pear.

SOFTWARE PACKAGE (Linux)

fastq-barcode.pl

Daniel Smith
https://gist.github.com/dansmith01/3920169

cmd COMMAND

```
./fastq-
barcode.pl Undetermined_S0_L001_I1_001.fastq trim_out_pear.assembled.fastq > out.barcodes.f
astq
```

Remove non-existent barcodes. Link: https://gist.github.com/dansmith01/3920169

Splitlibraries

Step 9.

Perform a demultiplexing of Fastq sequence data.

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

cmd COMMAND

```
split_libraries_fastq.py -m Bilbao_mapping_file.txt -i trim_out_pear.assembled.fastq -
b out.barcodes.fastq -o SPLIT_out
/ --rev_comp_mapping_barcodes
```

Demultiplexing of Fastq sequence data.

Chimeric sequences elimination

Step 10.

Before carrying out the taxonomic assignment, the chimera sequences were removed by identify_chimeric_seqs.py in QIIME using the usearch61 (v7.0.1090) de novo method. The resulting dataset was then analyzed by QIIME software (v1.9). We only included sequences that were 240-260 bp in length (average 253bp) to avoid background noise in the subsequent analyses.

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

cmd COMMAND

```
identify_chimeric_seqs.py -m usearch61 -i SPLIT_out/seqs.fna --suppress_usearch61_ref -
o chimera_usearch61_denovo/

filter_fasta.py -f SPLIT_out/seqs.fna -o SPLIT_out/filter_seqs_no_chim.fasta  -
s chimera_usearch61_denovo/chimeras.txt -n
```

Removing chimeric sequences using usearch61 algorithm. Usearch sowtware link: http://www.drive5.com/usearch/

pick de novo OTUs

Step 11.

An open reference OTU picking method was used in QIIME for clustering using a 97% similarity cut-off using UCLUST algorithm (v1.2.22q) and the taxonomy of the reference sequences was assigned based on Silva 119 database version (clustered at 97% identity). The OTUs with which representative sequences failed in PYNAST alignment were discarded.

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

cmd COMMAND

```
pick_open_reference_otus.py -i SPLIT_out/filter_seqs_no_chim.fasta -o OPEN_silva -
r Silva119_release/rep_set/97/Silva_119_rep_set97.fna -m uclust --min_otu_size 10
```

More information link: http://qiime.org/scripts/pick_open_reference_otus.html#index-0

Chloroplastial sequences elimination

Step 12.

After the taxonomical assignment, all chloroplast were removed from the BIOM file using filter_taxa_from_otu_table.py script in QIIME.

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

cmd COMMAND

```
filter_taxa_from_otu_table.py -i otu_table.biom -o otu_table_nocholo.biom -
n D_2__Chloroplast
```

Biom file preparation

Step 13.

Afterwards, samples with less than 5000 sequences were eliminated. Then, all OTUs with less than 10 sequences were removed. Finally, the BIOM file was normalized using metagenomeSeq's CSS algorithm, which normalized sequences using the cumulative sum scaling transformation.

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

cmd COMMAND

```
filter_samples_from_otu_table.py -i otu_table_nocholo.biom -o  -n 5000
filter_otus_from_otu_table.py -i otu_table_min5000.biom -o otu_table.biom -n 10
normalize_table.py -i otu_table_n10.biom -a CSS -o CSS_normalized_otu_table.biom -s
```

Taxonomic composition

Step 14.

To visualize the bacterial community composition of the samples, taxa_summary_through_plot.py command in QIIME v1.9 software was used.

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

cmd COMMAND

```
summarize_taxa_through_plots.py -i CSS_normalized_otu_table.biom -o taxa_summary_plot -
m Bilbao_mapping_file.txt
```

Alpha diversity

Step 15.

The alpha diversity (observed OTUs and Shannon) of the samples were calculated using phyloseq (v1.14) R package.

SOFTWARE PACKAGE (R -)

phyloseq, 1.14

Paul J. McMurdie
https://github.com/joey711/phyloseq/

COMMAND

```
table_otu = import_biom("CSS_normalized_otu_table.biom")
metadata = import_qiime_sample_data("Bilbao_mapping_file.txt")
tree = read.tree("rep_set.tre")
DOM = merge_phyloseq(table_otu, metadata, tree)
DOM
colnames(tax_table(DOM)) <-
 c(k = "Kingdom", p = "Phylum", c = "Class", o = "Order", f = "Family", g = "Genus", s = "Species")
ntaxa(DOM)
rank_names(DOM)

p <-
 plot_richness(DOM, x = "SampleType", color = "SampleType", measures = c("Chao1", "Observed"))
p + geom_boxplot()
print(p)
```

A R script using phyloseq R package to plot the alpha diversity for each water mass alogn the annual cycle.

Core-microbiome analysis

Step 16.

The core microbiome was analysed using compute_core_microbiome.py command on QIIME v1.9.

LINK:

http://bioinformatics.psb.ugent.be/webtools/Venn/

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

COMMAND

```
compute_core_microbiome.py -i CSS_normalized_otu_table.biom -o otu_table_core_IS --
mapping_fp Bilbao_mapping_file.txt --valid_states "SampleType:IS"
compute_core_microbiome.py -i CSS_normalized_otu_table.biom -o otu_table_core_MS --
mapping_fp Bilbao_mapping_file.txt --valid_states "SampleType:MS"
compute_core_microbiome.py -i CSS_normalized_otu_table.biom -o otu_table_core_OS --
mapping_fp Bilbao_mapping_file.txt --valid_states "SampleType:OS"
compute_core_microbiome.py -i CSS_normalized_otu_table.biom -o otu_table_core_B30 --
mapping_fp Bilbao_mapping_file.txt --valid_states "SampleType:B30"
compute_core_microbiome.py -i CSS_normalized_otu_table.biom -o otu_table_core_B33 --
mapping_fp Bilbao_mapping_file.txt --valid_states "SampleType:B33"
compute_core_microbiome.py -i CSS_normalized_otu_table.biom -o otu_table_core_B35 --
mapping_fp Bilbao_mapping_file.txt --valid_states "SampleType:B35"
```

The resulting list of core OTUs of each water mass was inclue in Venn diagram analysis of http://bioinformatics.psb.ugent.be/webtools/Venn/

Supervised machine learning

Step 17.

A supervised learning analysis was performed for estuarine water masses using the Random Forests classifier, ten-fold cross-validation models, and 1,000 trees. OTUs were considered “predictors” and sample type or water mass were the “class label”. This method determines the diagnostic power of bacterial profiles for predicting the characteristic community of the water masses by using a subset of samples to train a model that identifies unique features within data categories. The technique then determines the accuracy of the model by categorizing sample subsets that were not used to build the model. Through this method, we were able to evaluate not only the discriminative power in the microbial community to distinguish those groupings (sample type and water mass) but also the robustness of the groupings themselves.

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

cmd COMMAND

```
supervised_learning.py -i CSS_normalized_otu_table.biom -m Bilbao_mapping_file.txt -
c SampleType -o mlSampleType --ntree 1000 -f -e cv10
```

Beta diversity analysis

Step 18.

Principal coordinate analysis (PCoA) plots were used to examine community dissimilarity and determine the impact of environmentalexperimental factors (salinity, temperature, pH, DO concentration, precipitation, Chl-a) on microbial community structure. Result visualizations were made using EMPeror tool. Beta diversity was estimated using the unweighted UniFrac metric for 16S rRNA amplicon data. Also, an Unweighted Pair Group Method with Arithmetic mean (UPGMA) was used to construct a tree from the unweighted UniFrac beta diversity distance matrix. This analysis aimed to characterize the differences in phylogenetic community structure.

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

cmd COMMAND

```
beta_diversity_through_plots.py -i CSS_normalized_otu_table.biom -o Beta_div_plot/ -
m Bilbao_mapping_file.txt -t rep_set.tre
upgma_cluster.py -i Beta_div_plot/unweighted_unifrac_dm.txt -
o Beta_div_plot/beta_div_cluster.tre
```

Spearman's rank correlation analysis

Step 19.

To calculate correlations between OTUs abundances and environmental parameters, Spearman's rank correlation coefficient (rho) was carried out, by which it was possible to identify which OTUs were related to different environmental variables - salinity, temperature, pH, DO concentration, water turbidity, precipitation and chlorophyll. The impact of these environmental factors on bacterial communities was analyzed using the bio-env method of vegan (v. 2.3-4) R package.

cmd COMMAND

```
compare_categories.py --method bioenv -i Beta_div_plot/unweighted_unifrac_dm.txt -
m Bilbao_mapping_file.txt -
c precipitation,WaterTurbidity,Salinity,Temperature,DO,pH,Chlorophyll -o Spearman_bio_env -
n 999
```

Anosim analysis

Step 20.

Analysis of Similarity (ANOSIM) statistics (999 permutations) were carried out with the ANOSIM function and were used to test whether grouping samples by water mass was significant.

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

cmd COMMAND

```
compare_categories.py --method anosim -i Beta_div_plot/unweighted_unifrac_dm.txt -
m Bilbao_mapping_file.txt -c SampleTyple -o anosim_analysis -n 999
```

Adonis analysis

Step 21.

In order to calculate the percentage of beta diversity variation in each water mass explained by precipitation, an analysis of Adonis was performed.

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

cmd COMMAND

```
compare_categories.py --method adonis -i Beta_div_plot/unweighted_unifrac_dm.txt -
m Bilbao_mapping_file.txt -c Precipitation -o adonis_precipitation -n 999
```

This command was conducted for each water mass. No for the whole dataset.

Kruskal-Wallis analysis

Step 22.

To identify the differences in OTU composition between water masses, a Kruskal-Wallis non-parametric test was carried out between tributaries and estuarine water masses. In this way, the OTUs whose abundances significantly differed between water masses were identified.

SOFTWARE PACKAGE (Linux)

QIIME, 1.9.1

J Gregory Caporaso
https://github.com/biocore/qiime

cmd COMMAND

```
group_significance.py -i CSS_normalized_otu_table.biom -m Bilbao_mapping_file.txt -
c SampleType -o KW_mass.txt
```

DCA analysis

Step 23.

The community dissimilarity within the estuary and its tributaries were determined using a Detrended Correspondence Analysis (DCA) carried out by phyloseq (v. 1.14) R package.

SOFTWARE PACKAGE (R -)

phyloseq, 1.14

Paul J. McMurdie
https://github.com/joey711/phyloseq/

cmd COMMAND

```
table_otu = import_biom("CSS_normalized_otu_table.biom")
metadata = import_qiime_sample_data("Bilbao_mapping_file.txt")
tree = read.tree("rep_set.tre")
DOM = merge_phyloseq(table_otu, metadata, tree)
DOM
```

```
colnames(tax_table(DOM)) <-
 c(k = "Kingdom", p = "Phylum", c = "Class", o = "Order", f = "Family", g = "Genus", s = "S
pecies")
ntaxa(DOM)
rank_names(DOM)
p2 = plot_ordination(DOM, ordinate(DOM, "DCA", "bray"), color = "SampleType",
                     type = "sample", label = "SampleType", shape = "Season")
print(p2)
```

Extended Local Similarity analysis (eLSA)

Step 24.

To understand the bacterial dynamics in the inner euhaline zone of the Estuary of Bilbao, where the low DO concentrations and high values of temperature and chlorophyll concentrations dominate in summer, we used extended Local Similarity Analysis (eLSA) software. The analysis was performed using OTUs with highest abundance values at B30 samples. Following eLSA software guidelines, a total of 85 OTUs were included in the analysis. To adapt to the algorithm limitations and minimize computational cost, eLSA was used to reveal statistically significant local and potentially time-delayed association patterns between OTUs and environmental factors. Normalization of variables was performed by ‘robustZ’ method, including 14 time spots for the total number of sampling months. The rest of the analysis settings were set to default. Lastly, q-values were calculated to determine false-discovery rates. Correlations with q<0.01 were visualized in Cytoscape v3.2.1, creating a continuous mapping-based network.

SOFTWARE PACKAGE (Linux)

eLSA, 1.0.6

Charlie Xia
https://bitbucket.org/charade/elsa/src

COMMAND

```
lsa_compute table_B30_elsa.txt B30_eLSA.lsa -r 1 -s 14 -d 1 -q scipy
```

table_B30_elsa.txt (OTU table in eLSA admited format) B30_eLSA.lsa (output) -r (replica for each time-spot) -s (number of time-spot) -d (delay) -q (qvalue calculation method)