Reproducing and extending the analysis of Lloyd et al (2013) with the `Lloyd.et.al.Cell.abundance.metaanalysis` package

Andrew D. Steen[1,2], Megan K. May[2,3,4], and Karen G. Lloyd[2]

[1]andrew.decker.steen@gmail.com
[2]Department of Microbiology, University of Tennesse, Knoxville
[3]DePauw University
[4]Present address: Woods Hole Oceanographic Institution

11 November 2013

1 Introduction

This package allows users to reproduce, and more importantly to extend, the data analysis from the paper by Karen G. Lloyd, Megan K. May, Richard T. Kevorkian, and Andrew D. Steen (2013), "Meta-analysis of quantification methods shows archaea and bacteria to be similarly abundant in the subseafloor", Applied and Environmental Microbiology, doi: 10.1128/AEM.02090-13.

2 Reproducing the analysis

The simplest way to reproduce the entire analysis is to use

```
> invisible(reproduce_research,
+           print_plots=TRUE,
+           save_plots=TRUE,
+           fast_calc=FALSE)
```

This will create all of the analysis in the paper, print and save plots and certain tables, and display some tables in the console output.

This is a fine way to check our work, and we encourage it. However, the purpose of writing this package is to encourage **extensions** of our analysis, either by adding data to the database as it becomes avaialable, or by applying new analyses to the existing data set or to new data sets.

3 Extending the analysis

In order to extend the analysis, it is useful to be able to:

- Add data to our database
- Run functions on our database (or an extended database) in isolation

3.1 Working with our database

This package is built on two central databases, containing data from the water column and from sediments. These databases are similar, but not identical. These are included as two Microsoft Excel (`.xlsx`) files included as supplemental files to the journal article. Somewhat confusingly, they are encoded as three R objects (each included with the package as an `.RData` file). These files are:

- `corrected_sw` containing seawater data, and
- `corrected_seds` containing sediments data.
- `all_data`, containing concatenated seawater and sediments data, with a somewhat reduced set of columns.

For most of the analysis, the more limited data frame is used when possible; the `all_data` data frame is only used when necessary.

The original versions of these data sets are loaded (via lazy loading) when the package is loaded. Thus, to access the original dataset you may call `all_data`, `corrected_sw`, or `corrected_seds` once the package is loaded.

You may create versions of these data frames based on new underlying databases. To do this, use the function `read_data()`. By default, this function will load the database that the paper is based on. Depending on the function arguments, it can be used to read data from an Excel file (which should have the same columns as the Excel file supplied with the paper), or from a different `.RData` file, if you have extended the databse. For instance, it might be easiest to add data to our database by extending the Microsoft Excel file supplied in the paper's supplemental, and saving the excel file with a new filename. Then, you would read that file into R using

```
> # Read in the modified database
> new_data_list <- read_data(reload.from.xlsx=TRUE,
+                            seds_fn="myPath/mySeds.xlsx",
+                            seds_sheet_name="Sheet1",
+                            sw_fn="myPath/mySw.xlsx",
+                            sw_sheet_name="Sheet1")
```

Note that `read_data()` does some processing of the Excel files in addition to reading them in, so you don't want to load the data simply using `read.xlsx()`.

Ideally you should only read from Excel files when you have to, because reading Excel files into R is slow (ca. 4 minutes on my machine for both Excel files.)

Therefore, once you have read in your modified database, save `corrected_sw`, `corrected_seds`, and `all_data` as `.RData` files:

```
> save(all_data, file="myPath/all_data_modified.RData")
> save(corrected_sw, file="myPath/corrected_sw_modified.RData")
> save(corrected_seds, file="myPath/corrected_seds_modified.RData")
```

From now on, you can load your modified database using

```
> new_data_list <- read_data(reload.from.xlsx=FALSE,
+                            all_data_fn="myPath/all_data_modified.RData",
+                            corrected_sw_fn="myPath/ccorrected_sw_modified.RData",
+                            corrected_seds_fn="myPath/corrected_seds_modified")
```

3.2 Performing one analysis at a time

Once the three central data frames are loaded using `read_data()`, you can use or modify the functions included with the package to run one analysis at a time, or to write/extend your own analyses. In many cases, it may be simplest to call thes functions by stepping through `reproduce_research()`; i.e. by opening the file and running it line-by-line as a script.

The full list of functions included in the package follows. Each is documented separately. Access the documentation for each function using `?`.

- `AIC_lik()`: Calculates log-liklihoods using the output of `AIC`
- `aov_perm_test()`: Performs permutation test using Analysis of Variance
- `boxplots_by_perm()`: Creates the plots in Fig 3, and calculates relevant statistics
- `intertidal_yield_fig()`: Makes a figure of yields of FISH/CARD-FISH for intertidal sediments, analogous to Fig 2
- `lm_stats()`: A particularly useful function which returns a single-row data frame containing slope, intercept, standard errors, p-values, and more for a linear model of the form `yvar~xvar`
- `make_qPCR_plots()`: Creates figure 4, about qPCR methods
- `make_sed_yield_boxplots()`: Makes boxplots of yield for sediments data (included in the supplemental), split by various methodological factors
- `make_sw_yield_boxplots()`: Makes boxplots of yield for seawater data (included in the supplemental), split by various methodological factors
- `plot_cell_vs_fish()`: Makes Figure 1
- `qPCR_516_evaluation()`: Makes plot and calculates summary stats for qPCR using or not using 516 as a primer

- `read_data()`: Reads and formats cell abundance database
- `reproduce_research()`: Master function to reproduce all analysis in the paper
- `sed_bac_and_arc_v_depth()`: Makes plots and models of sediments bac and arc concentrations vs depth
- `sed_percent_arc_v_depth()`: Make depth profiles and calculate linear models for sediment percent Archaea vs depth
- `significance_labeller()`: Create 'significance code' based on p values
- `single_sw_yield_boxplot()`: Function to make generic boxplots as in the supplemental figures
- `sw_depth_profiles()`: Creates depth profiles in figures 5 a-c: Bacteria, Archaea, and Also performs breakpoint analysis
- `yield_by_core()`: Create a boxplot of yield (total cells by *-FISH relative to total cells by direct count) for each core in the database

4 Appendix: Columns in each data frame

Table 1: Columns and data types for the **all_data** data frame.

column name	data type	notes
paper	factor	abbreviated text name of the reference
depth	numeric	depth, m
totalcells	numeric	cells per cm^3 [1]
qPCRtotal	numeric	16s copies per cm^3
Cell.stain	factor	
CARDFISH.Bac.per.cc	numeric	cells per cm^3
FISH.yield	numeric	*.FISH-counted cells / general-stain-counted cells
Fraction.Arc.CARDFISH	numeric	Archaea/(Archaea + Bacteria) by CARDFISH
Fish.or.CARDFISH	factor	also includes `"Polyribonucleotide FISH"`
Fixative	factor	
Bac.permeabilization	factor	
Arc.permeabilization	factor	
environment	factor	`"seawater"` or `"sediments"`
qPCRbac	numeric	
qPCRarc	numeric	
Cell.stain.1	factor	
Bac.probe	factor	

[1] When reported in other units (e.g. per gram sediment dry weight), these were adjusted to per cc.

Arc.probe	factor	
core	character	sediment core label
qPCR.Bac.per.cc	numeric	copy number of Bacteria per cc by qPCR
qPCR.Arc.per.cc	numeric	copy number of Archaea per cc by qPCR
qPCRuniversal	numeric	copy number by qPCR using 'universal' primers
percentqPCR	numeric	fraction (not percent) of Archaea by qPCR
SYBR.vs.Taqman	factor	qPCR probe
DNA.extraction.procedure	factor	

Table 2: Columns and data types for the **corrected_sw** data frame - only listing those columns that are not present in `all_data`

column name	data type	notes
Sample	factor	identifier for samples within the paper
Date	factor	mixed formats due to bad Excel entry
qPCR.MCG..copies.mL.water.	numeric	
Counting.method	factor	`"Microscopeeye"` or `"Microscopyautomated"`
depth_log10	numeric	log 10-transformed depth

Table 3: Columns and data types for the **corrected_seds** data frame - only listing those columns that are not present in in `all_data`

column name	data type	notes
sulfate	factor	mostly numeric in mM, some text
CARDFISH.Arc.nd	logical	were Archaea detected by CARDFISH?
Bac.formamide	numeric	concentration formamide for Bacteria
Arc.formamide	factor	concentration formamide for Archaea
Sonication	factor	sonication protocol
Average.cells.per.field	factor	includes text and numbers
Filter	factor	filter type
Bac.Probe	factor	Bacterial *-FISH probe
Arc.Probe	factor	Archaeal *-FISH probe
Arc.forward	factor	
Arc.reverse	factor	
TaqMan.Arc	factor	TaqMan probe for Archaea
Bac.forward	factor	Bac.forward
Bac.reverse	factor	
Taqman.Bac	factor	
Universal.forward	factor	
Universal.reverse	factor	
Taqman.Universal	factor	
Arc.standard	factor	qPCR standard, Archaea
Bac.standard	factor	qPCR standard, Bacteria

Mud.volcano.or.seep.	logical	samples come from a mud volcano or seep
Water.depth	numeric	water depth at sediment surface
Environment.Type	character	
Non.intertidal.marine...	logical	full name is much longer
DNA.extraction.efficiency...	factor	full name is much longer
Template.DNA.dilution.factor	factor	
Uses.516.for.Arc	logical	
Uses.349.for.Arc	logical	
Uses.806.for.Arc	logical	
Uses.331.for.Bac	logical	
Uses.340.for.Bac	logical	
Uses.515.for.Bac	logical	
paperNumber	numeric	numeric label for each paper