

Box 2: Comparison of QIIME OTU Table collapsing code with native QIIME OTU table data structures (Panels A-D) and biom-format Table objects (Panel E). Given an OTU table and associated sample metadata, this code collapses sets of samples with the same value for a given metadata entry into a single sample. Here we illustrate the vastly reduced complexity of this operation using biom-format Table objects (in QIIME 1.4.0-dev svn revision 2770 and later; Panel E) versus native QIIME objects in QIIME (QIIME 1.4.0 and earlier, Panels A-D). The full version of each example can be found in the QIIME repository using the information in each panel caption.

Panel A: QIIME 1.4.0: Qiime/scripts/summarize_otu_by_cat.py
(prior to switch to biom-format Table objects).

```
62 mapping_f = open(mapping_fp, 'U')~
63 otu_table_f = open(otu_table_fp, 'U')~
64 ~
65 summarized_otu_table = \~
66     summarize_by_cat(mapping_f, otu_table_f, mapping_category, normalize)~
67 ~
68 f = open(output_fp, 'w')~
69 f.write(summarized_otu_table)~
70 f.close()~
```

Panel B: QIIME 1.4.0: Qiime/qiime/summarize_otu_by_cat.py
(prior to switch to biom-format Table objects).

```
36 def get_sample_cat_info(lines, category):~
37     cat_by_sample = {}~
38     sample_by_cat = defaultdict(list)~
39     meta_dict = {}~
40     num_samples_by_cat = defaultdict(int)~
41     label_lists_dict = defaultdict(list)~
42     mapping_data, header, comments = parse_mapping_file(lines)~
43     ~
44     category_labels = header~
45     index = category_labels.index(category)~
46 ~
47     for line in mapping_data:~
48         categories = line[0:len(category_labels)+1]~
49         sample = categories[0].strip()~
50         meta_dict[sample] = [(categories[index], 0)]~
51 ~
52         cat_by_sample[sample] = [(l.strip(), c.strip()) \~
53                                 for l, c in zip(category_labels, categories)]~
54 ~
55         cat_list = []~
56         for i, (l, c) in enumerate(zip(category_labels, categories)):~
57             if c not in label_lists_dict[l]:~
58                 label_lists_dict[l].append(c)~
59             l = l.strip()~
60             c = c.strip()~
61             cat_list.append((l, c))~
62             sample_by_cat[(l, c)].append(sample)~
63             num_samples_by_cat[(l, c)] += 1~
64 ~
65         cat_by_sample[sample] = cat_list~
66 ~
67     return cat_by_sample, sample_by_cat, len(category_labels),~
68     meta_dict, label_lists_dict, num_samples_by_cat~
```

Panel C: QIIME 1.4.0: Qiime/qiime/summarize_otu_by_cat.py
(continued; prior to switch to biom-format Table objects).

```
70 def get_counts_by_cat(lines, num_meta, meta_dict, cat_list, category, num_samples_by_cat,
71                       normalize):
72     ~
73     samples_from_mapping = meta_dict.keys()
74     norm_otu_table = []
75     sample_counts = defaultdict(int)
76     cat_otu_table = []
77     otus = []
78     taxonomy = []
79     sample_ids, otu_ids, otu_table, lineages = parse_otu_table(lines)
80     ~
81     label_list = sample_ids
82     if lineages == []:
83         is_con = False
84     else:
85         is_con = True
86     for idx, line in enumerate(otu_table):
87         new_line = []
88         label_dict = defaultdict(int)
89         data = line
90         to_otu = otu_ids[idx]
91         otus.append(to_otu)
92         con = ''
93         if is_con:
94             con = '; '.join(lineages[idx])
95             counts = data
96         else:
97             counts = data
98         taxonomy.append(con)
99         if not normalize:
100             for i, c in zip(label_list, counts):
101                 if i in samples_from_mapping:
102                     label_dict[meta_dict[i][0][0]] += c
103             for i in cat_list:
104                 new_line.append(str(label_dict[i]))
105             cat_otu_table.append(new_line)
106         ~
107         else:
108             new_line.extend(counts)
109             norm_otu_table.append(new_line)
110             for i, c in zip(label_list, counts):
111                 sample_counts[i] += c
112     total = 0
113     if normalize:
114         for l in norm_otu_table:
115             counts = l
116             new_line = []
117             label_dict = defaultdict(float)
118             getcontext().prec = 28
119             for i, c in zip(label_list, counts):
120                 if i in samples_from_mapping:
121                     label_dict[meta_dict[i][0][0]] += float(c)/(sample_counts[i])
122             for i in cat_list:
123                 new_line.append(round((label_dict[i]/ num_samples_by_cat[(category,i)]),5))
124             cat_otu_table.append(new_line)
125     return cat_otu_table, otus, taxonomy
```

Panel D: QIIME 1.4.0: Qiime/qiime/summarize_otu_by_cat.py
(continued; prior to switch to biom-format Table objects).

```
128 def summarize_by_cat(map_lines,otu_sample_lines,category,norm):-
129     """creates the category otu table"""-
130     cat_by_sample, sample_by_cat, num_meta, meta_dict, label_lists_dict, \-
131         num_samples_by_cat = get_sample_cat_info(map_lines,category)-
132     -
133     lines, otus, taxonomy = get_counts_by_cat(otu_sample_lines, num_meta, \-
134         meta_dict,label_lists_dict[category],category,num_samples_by_cat,\-
135         norm)-
136     -
137     #This for loop was added to remove columns that sum to 0, since you may -
138     #pass a mapping file that has more samples than in the OTU table, hence resulting-
139     #in columns with no counts-
140     new_labels=[]-
141     new_lines=[]-
142     for i,line in enumerate(zip(*lines)):-
143         total_col=sum([float(x) for x in line])-
144         if total_col>0:-
145             new_lines.append(line)-
146             new_labels.append(label_lists_dict[category][i])-
147     new_lines=zip(*new_lines)-
148     -
149     lines = format_otu_table(new_labels, otus, array(new_lines), \-
150         taxonomy=taxonomy,-
151         comment='Category OTU Counts-%s'% category)-
152     return lines-
```

Panel E: QIIME 1.4.0-dev, revision 2770: Qiime/scripts/summarize_otu_by_cat.py
Replacement for all code in Panels A-D after switch to biom-format Table objects from native QIIME
OTU table data structures.

```
64     # define a function that returns the bin a sample should be placed into-
65     bin_function = lambda sample_metadata: sample_metadata[mapping_category]-
66     # parse the sample metadata and add it to the OTU table (we assume that-
67     # sample metadata is not already present in the table)-
68     sample_metadata = parse_mapping_file_to_dict(open(mapping_fp,'U'))[0]-
69     table = parse_biom_table(open(otu_table_fp,'U'))-
70     table.addSampleMetadata(sample_metadata)-
71     # create a new OTU table where samples are binned based on their return-
72     # value from bin_function -
73     result = table.collapseSamplesByMetadata(bin_function,norm=False,min_group_size=1)-
74     -
75     # normalize the result if requested by the user-
76     if normalize:-
77         result = result.normObservationBySample()-
78     -
79     # write a new BIOM file-
80     f = open(output_fp,'w')-
81     f.write(format_biom_table(result))-
82     f.close()-
```