Explaining the fraction mapped bug

Kelly L. Sovacool

16 Dec. 2020

The Bug

For each seed, the dataset was split into a reference and query fraction. Then, reference sequences were clustered *de novo* into OTUs with OptiClust. The seed which produced the best *de novo* OTUs was selected and used as the set of reference OTUs for all downstream OptiFit runs. For each seed, the query sequences were fit to the reference OTUs from the best clustering seed with OptiFit. This was a problem because the same seed variable was used for the dataset split, OptiClust, and OptiFit steps. Whenever the current OptiFit seed corresponded to the best OptiClust seed, the query and reference sequences were from the same data split and the results were correct; in all other cases they were from different splits and the results were unexpected.

With open-reference clustering, any query sequences which can't initially be mapped to the reference OTUs are then clustered *de novo*, so that all query sequences end up in the final OptiFit OTUs. Thus, the fraction of query sequences which mapped to the reference should always be 1 with open-reference clustering. However, with this bug, the fraction mapped was less than 1 whenever the OptiFit seed was different from the best OptiClust seed.

To fix the bug, I removed the step that picked the best seed from OptiClust for all downstream OptiFit runs. This step caused query sets to be fit to reference sets from different data splits. It was not necessary anyway since OptiClust results were remarkably similar across different seeds. If we wanted to keep this step, an alternate fix would be to have multiple seed variables (i.e. one each for the data split and clustering steps) to prevent query sequences from one data split being fit to reference sequences from a different data split.

Why fraction mapped = query fraction

- q: fraction of sequences in the query.
- $1 - q$: fraction of sequences in the reference.
- m: the mapped sequences as a fraction of the total sequences.
- $\frac{m}{q}$: fraction of query sequences that mapped.
- u: the unmapped sequences as a fraction of the total.
- $\frac{u}{q}$: fraction of query sequences that did not map.

Interestingly, with open-reference clustering and a simple random sampling method for splitting the dataset into reference and query fractions, **the fraction of query sequences which mapped to the reference was always** q.

$$\frac{m}{q} = q$$

$$m = q^2$$

:

$$\frac{u}{q} = 1 - \frac{m}{q}$$

$$\frac{u}{q} = 1 - \frac{q^2}{q}$$

$$u = q(1 - q)$$

So we expect the unmapped sequences as a fraction of the total to be $q(1 - q)$ based on the observation that the fraction of query sequences which mapped to the reference was always q.

Let A be the event that sequence X is selected for the query. Let B be the event that sequence X is selected for the reference. A and B are independent events. When a simple random sample is used, the probability that sequence X is selected for the query with seed 1 and sequence X is selected for the reference with seed 2 is:

$$\mathbb{P}(A)\mathbb{P}(B) = q(1 - q)$$

When running OptiFit with the printref parameter set to false, the reference sequences are not written to the output list file. It seems that this includes any reference sequences that are also in the query. Therefore, the sequences that are in both the query and reference are counted as unmapped, i.e.

$$u = q(1 - q) = \mathbb{P}(A)\mathbb{P}(B)$$