

Mapping metagenomic reads to a reference of markers can detect novel eukaryotes

by Wojtek, Ann, Kathryn, Dan, possibly others

Introduction

Eukaryotes such as fungi and protists are frequently present in microbial communities. They are more difficult to detect in whole-genome sequencing reads than bacteria and archaea, so their presence is often not reliably reported by common tools.

One challenging aspect of identifying reads in an environmental sample is potential dissimilarity of a sequenced material to any previously known reference. If signal from mapped reads does not take incompleteness of the reference and potential for false positives into account, reported results can be absurd (R Marcelino, Holmes, and Sorrell 2020).

A recent method aimed at solving this problem is EukDetect (Lind and Pollard 2021), a tool based on read mapping which exploits the finding that using a specially prepared reference of sequences only typically present in eukaryotes can remove spuriously aligning bacterial reads. However, a possibility of an unknown eukaryote being present in the sample is not discussed in the original EukDetect publication. Only a small proportion of eukaryotic species has been named, let alone sequenced - the 1/23/2021 version of the EukDetect reference used in this publication contains sequences for 4023 taxa, and there are estimated a 2-3 million of just fungi (Hawksworth and Lücking 2017).

We use alignments of simulated reads to show that EukDetect is able to report approximately correct results when the sequenced material contains an unknown species or a non-reference strain of a species, but it is limited by its current approach to less confident alignments, specifically, its strategy to filter aligned reads on the MAPQ field, originally defined in the context of reference genomes like the human genome (Li et al. 2009). We exhibit strengths and limitations of filtering on MAPQ, and provide an interpretation of this field in the context of metagenomics. Finally, we suggest how programs like EukDetect could improve their result reporting such that an appearance of a species that differ from available references can be correctly interpreted.

Methodology

Methods of differentiating species through their sequences can be tested in silico through simulation (Hovhannisyan et al. 2020). We use **wgsim** (Li 2011) to simulate reads from EukDetect’s reference of BUSCOs from OrthoDB (Kriventseva et al. 2019), and use **bowtie2** (Langmead and Salzberg 2012) to align them back to the reference.

When using **wgsim** we set read length to 100, base error rate to 0, and other parameters set to their default values unless otherwise specified.

We approximate the possibility an unknown species being present in the sequenced material through a hold-out analysis. We remove sequences for 371 species from the reference (one tenth of the species) to form a hold-out set, and build a **bowtie2** index with the remaining nine-tenth of the species for searching the remaining set. Skipping any inputs where **wgsim** considers too fragmented yields 338 species to be sampled from.

First we perform an analysis of simulated samples by preparing 338 files each containing reads from one hold-out species at 0.1 coverage (Sims et al. 2014). This lets us establish how well EukDetect currently works when given unknown species and identify the MAPQ ≥ 30 filter as responsible for not passing enough information through. We also try a modification to EukDetect to instead filter on MAPQ ≥ 5 used by default in MetaPhlAn (Segata et al. 2012), a frequently used program for estimating taxonomic abundance also based on read mapping.

We follow with an analysis of simulated reads. We simulate a large number of reads and align them. For each aligned read, we record correctness of the match based on its source taxon with the taxon it aligned to as well as properties of the alignment (identity, MAPQ). Then lets us infer trends between reported properties of alignments and two measures of their correctness, namely precision: a proportion of correctly mapping reads among reads that map to any reference, and recall: a proportion of sampled reads that correctly map, similarly to how the values are used by the OPAL framework (Meyer et al. 2019). For reads that do not align back to the BUSCO they are sampled from, we annotate the mismatch with a level of the lowest common taxon containing source and match, with help of the ETE toolkit (Huerta-Cepas, Serra, and Bork 2016) using the NCBI database version dated 2020/1/14 packaged with EukDetect.

We do this in three contexts: reads sampled from the whole reference and then mapped back to it (an optimal case we might expect in real data), equivalently sampled reads which are then modified (a more realistic case where a sampled organism is of a different strain to the reference), and reads from a hold-out set mapped to the remaining set (a case of unknown species). In the first two cases, we consider the read to map correctly if it maps to the same taxon, and in the case of mapping species from a hold-out set, if it maps to another taxon of the same genus.

Similarity of reference sequences makes mapping reads more difficult, and the need for strain-level differentiation in fields like genomic epidemiology has inspired the creation of specialized aligners like KMA (Clausen, Aarestrup, and Lund 2018). To understand how `bowtie2` is affected by duplication of sequences, we prepare an index of 371 species from the reference where each sequence is included once as originally, and once after extending it by a single “A” base.

Results

EukDetect given unknown species

Running EukDetect on each of the samples produces an empty list of results for 219 samples, and some results for 119 samples. Of these, 76 are one taxon of the same genus as the source species in the hold-out set. 17 are one taxon of a different genus, and 26 are more than one taxon.

Running `bowtie2` and keeping samples which have any reads mapped - that is, applying no filtering - reports results for 301 of the samples. We investigate it by applying some of EukDetect’s filters in turn. The first filter based on query length has no effect as we are using simulated reads. The second filter applied by EukDetect, based on the MAPQ field, reduces the number of samples with any mapped reads to just 156, additionally requiring reads mapping to two different markers in a taxon narrows down the list further to 133, and also requiring four reads fully accounts for all missing results. Skipping the MAPQ ≥ 30 field and only requiring two markers and four reads in a taxon reports results for 206 samples. This shows that the choice of filtering rules is key to sensitive detection of organisms using mapped reads.

Re-running EukDetect modified to filter on MAPQ ≥ 5 reports some results for 160 values: 78 are one taxon of the same genus, 13 are one taxon of a different genus, and 69 are more than one taxon.

Mapping simulated reads

Tools for read mapping like `bowtie` or `samtools` have been developed in the context of reference genomes like the human genome (Langmead et al. 2009), (Li et al. 2009). The SAM specification originating with `samtools`, which `bowtie` and `bowtie2` follow, defines the MAPQ field as a measure of certainty about position of the alignment. Eukdetect aligns metagenomic reads with `bowtie2` to a reference which is not like a single reference genome - it contains groups of similar sequences from many genomes - and one of the filters it applies to the alignments is to require $\text{MAPQ} \geq 30$.

Our basic experiment is to simulate reads from each taxon in the reference and then map it back to the whole reference. Most reads map correctly - overall, average precision and recall are both 95.1%, and adding the $\text{MAPQ} \geq 30$ filter increases precision to 99.7% while decreasing recall to 91.7%. To convey the same information with a differently calculated statistic: 8% of the reads map with $\text{MAPQ} < 30$ and 46.2% of those are incorrectly mapped, while among reads with $\text{MAPQ} \geq 30$, only 0.3% are incorrectly mapped. Most misses are near misses: 89% of reads that do not map back to a sequence of their species map within the correct genus. Average MAPQ value among correct mappings is 36.1, among near misses 5.2, and 4.4 among other misses. For $\text{MAPQ} \geq 5$, precision and recall are 99.0% and 96.0%.

The degree of improvement achieved in this way turns out to depend on the source taxon of the reads (figure A). Out of 3977 taxa whose reads map back to the reference, reads from 1908 map with 100% precision. After applying the $\text{MAPQ} \geq 30$ filter, 1105 more taxa map with 100% precision, but 146 taxa still have precision worse than the pre-filter average of 95.1%. In the case of five taxa (*Fusarium* cf. *fujikuroi* NRRL 66890, *Escovopsis* sp. Ae733, *Favella ehrenbergii*, *Leishmania peruviana*, *Mesodinium rubrum*) applying the filter decreases precision, and for one taxon (*Chlamydomonas* debaryana var. *cristata*) applying the filter removes all reads.

Within the hold-out set, most misses are still near misses - 73% of reads from a hold-out set that do not map back to their genus map to another genus in the same family. Average MAPQ value is 12.7 for reads that map to the same genus, 11.7 among reads that map to different genus within the same family, and 14.8 for reads that map to a different family. Same-genus precision of 82% only increases to 83.6%, while the same-genus recall of 30% decreases to below 7%. For $\text{MAPQ} \geq 5$, the precision and recall values are 87% and 18.2%.

The effect of applying the $\text{MAPQ} \geq 30$ filter continues to vary for different taxa. Among 322 taxa in the hold-out set whose reads map to the remaining set, 48 map only to the taxa of the same genus, while 117 do not map to taxa of the same genus at all. Applying the $\text{MAPQ} \geq 30$ filter only keeps reads mapping to correct genus for 94 more taxa, while removing all reads that map to the correct genus in 6 cases, and removes all reads for 126 taxa.

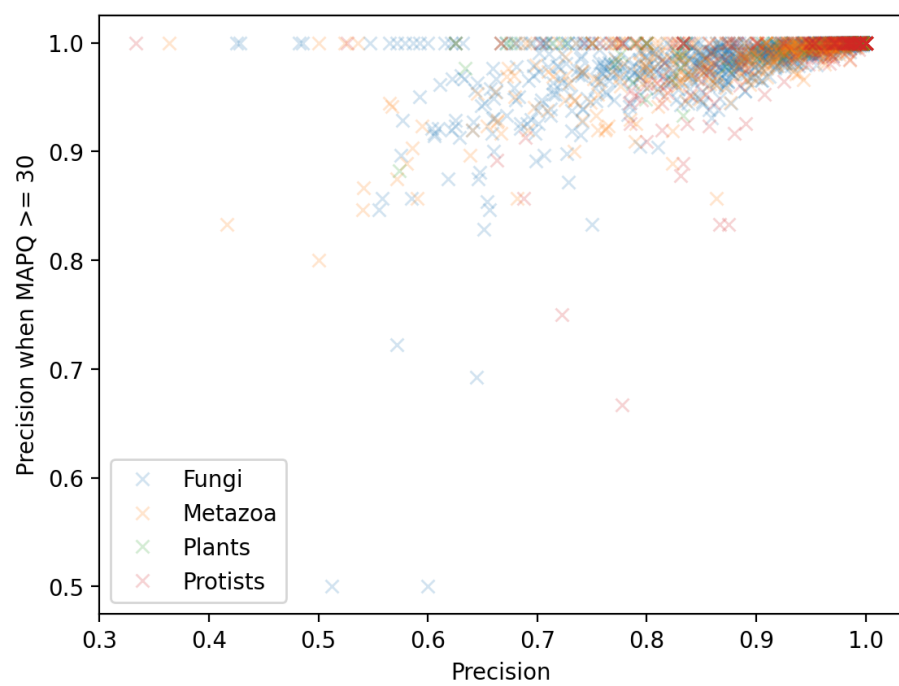


Figure 1: (A) Precision and fraction of reads with MAPQ ≥ 30 , each dot is source taxon

To understand an effect of smaller differences between reference and signal, we analyse reads that are mutated before alignment (Figure B). We gradually increase the `wgsim` mutation rate parameter until recall drops below 10%. Precision stays between 95% and 96% throughout the range of mutation rates, which is concordant with `bowtie2` preserving precision over recall as seen in e.g. (Peng et al. 2015). Keeping only reads with $\text{MAPQ} \geq 30$ improves precision to between 99.6% and 99%, similarly to the case of unmutated reads. However, fraction of reads with $\text{MAPQ} \geq 30$ declines more rapidly than recall does, and so the cost of improving precision in this way gets disproportionately large.

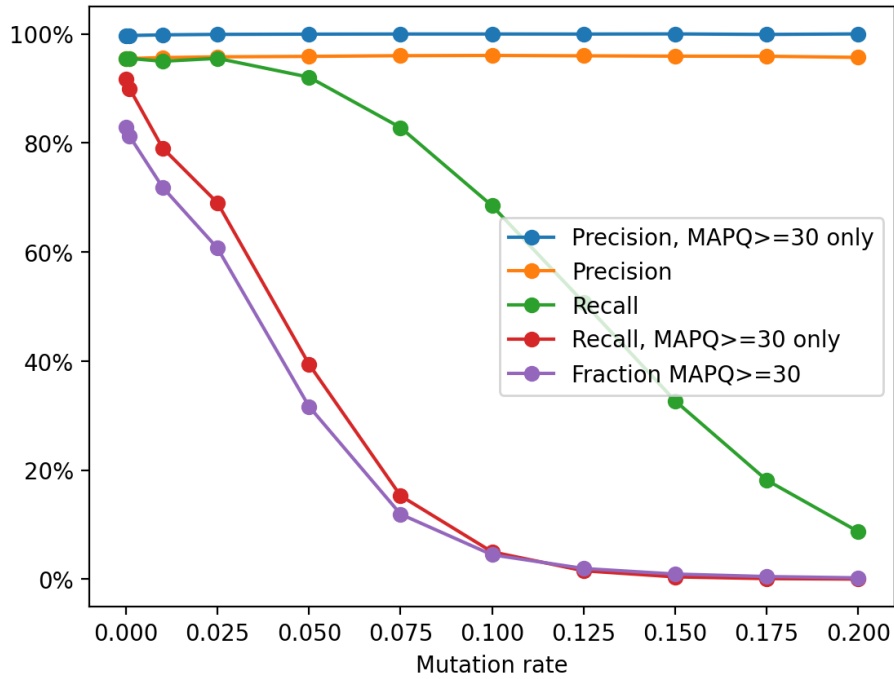


Figure 2: (B) Alignments of mutated reads, known species

Simulating reads and mapping them to an variant of the reference with 371 sequences included twice shows that a small amount of redundancy in reference sequences does not compromise same-species precision or same-species recall (both are 99.6%). The reported MAPQ values are almost uniformly 0 or 1.

Discussion

As shown above, precision with which metagenomic reads can be mapped varies based on their source taxon, and the effects of the MAPQ ≥ 30 filter vary based on the source of the reads. We can better explain this variability if we re-cast the task of identifying a source of reads given metagenomic reads as a reference as a nearest neighbour search in a space of sequences. A reference consisting of cDNA sequences from multiple species is different from a reference genome like the human genome, because naturally occurring proteins form isolated clusters of varying size and in-cluster similarity (Smith 1970). Since MAPQ is a measure of certainty about alignment position (Li et al. 2009), we can expect it to be low for reads whose nearest neighbour is either ambiguous or distant.

Reads map less correctly as well as with low MAPQ in particularly congested areas of sequence space, because a read coming from a segment shared between sequences does not contain enough information to assign it to the source. Our data suggests this might be the right model for a large fraction of errors, because most misses are near misses. In our opinion, rejecting ambiguously mapping reads can be a sensible strategy when there are plenty of other reads to use, but making use of ambiguous mappings can improve detection ability of taxonomic classifiers that focus on eukaryotes.

Meanwhile, when the source of reads is quite distant from its nearest reference which is nevertheless the best match for each read, reads either do not map or they correctly map with low MAPQ. Rejecting low MAPQ reads can not improve sensitivity in this case, and our data shows downsides of applying the MAPQ ≥ 30 filter to unknown species or non-reference strains.

Conclusion

We have shown that the MAPQ ≥ 30 filter used by EukDetect decreases the tool’s sensitivity at detecting unknown species. The trade-offs offered by the filter are very attractive when it is applied to reads from source taxa that are highly similar to exactly one taxon in the reference, but it does not universally improve results. Changing MAPQ ≥ 30 to a more permissive value like MAPQ ≥ 5 offers slightly different trade-offs - higher precision and recall for unknown species, but also an increased number of off-target hits.

We conclude that while the value of MAPQ is linked to correctness of results in general, it is not a reliable measurement of uncertainty per read when mapping metagenomic reads to a reference of markers. Instead of using the MAPQ value to filter alignments before binning them, tools like EukDetect could filter on average MAPQ per detected taxon, or communicate the value in conjunction with other measures of uncertainty of results.

Not in the paper

I've written these sections when the paper was going to be bigger. Perhaps there's a whole paper in there ! :)

Our method and software

We have developed a software tool, `marker_alignments`, for producing taxonomic profiles from alignments of marker genes, with a design goal of producing good guesses from a small amount of alignments to a potentially incomplete reference.

The tool is implemented in Python. It uses the `pysam` module to read alignment files, a `sqlite3` database to store alignments and build reports, and the MCL algorithm to convert pairwise similarities of markers and taxa into cluster assignments (Van Dongen and Abreu-Goodger 2012).

We have also developed a Nextflow workflow, `wbazant/marker-alignments-nextflow`, which bundles file download, a run of `bowtie2` set up to produce multiple alignments per read, and processing the alignments. For processing MicrobiomeDB data, we have settled on a set of parameters equivalent to the following procedure:

Step 1. (Filter alignments on length)

Keep alignments that are at least 60 bases long regardless of their MAPQ.

Step 2. (Marker clusters)

For each taxon, classify each marker as “at least average” or “below average” based on identity in its marker cluster. Reject taxa for which half or more of the markers are below average.

Step 3. (Taxon clusters)

Assign clusters to remaining taxa based on shared queries.

Step 4. (Unambiguous hits)

Return taxa which have at least two different reads aligned to at least two markers, and average alignment identity of at least 97%.

Step 5. (Strong ambiguous hits)

For each taxon cluster with no unambiguous hits that has least four markers and eight reads, join names of taxa in the cluster, prepend “?”, and report as one result.

Our data

Using the above workflow, we add eukaryotic detection analysis to our standard treatment of whole genome sequencing data on MicrobiomeDB (Oliveira et al. 2018), starting with Release 25 (2 Dec 2021). Below is a descri

metagenomic samples of gut microbiome samples of developing children in DIABIMMUNE (Vatanen et al. 2016), as well as healthy humans from the Human Microbiome Project (Turnbaugh et al. 2007).

Our results from processing Human Microbiome Project data (Turnbaugh et al. 2007) can be characterized broadly sensible but patchy overall. The most common taxa found are *Malassezia* and *Candida* fungi. Only 15 / 146 of stool samples report *S. cerevisiae*, an ubiquitous yeast that is likely to be present in the majority of the samples. This is in line with (Nash et al. 2017) who processed the same data through whole-genome alignments and report the sequencing depth use as insufficient to match sensitivity achieved by amplicon sequencing. Findings of eukaryotes that can potentially cause disease include a protozoan *Trichomonas tenax* in two oral samples, and 4/10 of samples from the subject ID 246515023 containing a fungus *Aspergillus fumigatus*.

Data from the DIABIMMUNE study (Vatanen et al. 2016) lets us cross-check our method with results reported in the original EukDetect publication. Having processed the same 1154 samples, we agree on 122 data points, EukDetect has additional 14, and we have additional 97. Some of the additional results can be explained as higher sensitivity of our method - for example, a common taxon *S. cerevisiae* is reported 67 times by us and 31 by EukDetect. The additional complexity of our method when handling unknown species produces a few disagreements. In sample G78909, EukDetect reports *Penicillium nordicum*, while our method reports a novel *Penicillium*, and in sample G78500, EukDetect's reported *Kazachstania unispora* is rejected by our method as an off-target hit. In sample G80329, our method agrees with EukDetect about *Candida parapsilosis*, and reports an additional *C. albicans* - plausibly, because the reads assigned to the two species align to entirely different markers.

Processing infant stool samples from the Preterm Infant Resistome study (Gibson et al. 2016) uncovered that a sample corresponding SRA run ID SRR3132435 has been colonised by organisms from the *Mucor* genus, possibly between sample collection and sequencing. This is not reported by EukDetect despite over three thousand reads from the sample mapping to *Mucor* markers, because they all get filtered out by the MAPQ ≥ 30 filter. Our method reports a presence of a species of *Mucor* most similar to *M. velutinosus*, *M. circinelloides*, and *M. ambiguus*. A second source for this result is STAT (Katz et al. 2021), assigning 0.34% of the reads assigned to order Mucorales on the basis of k-mer similarity.

Our interpretation of this result is that the sample has gone mouldy between sample collection and sequencing.

All results are publicly viewable and downloadable on MicrobiomeDB. In addition, results and scripts used for the DIABIMMUNE comparison can be found in supplemental material.

All in all, we are satisfied with the above workflow and consider it a robust incremental improvement.

Information about mismatches

Assigning a read to the sequence it is most similar to is generally possible with **bowtie2**, and frequently very successful: reads simulated from 1908 / 3977 taxa map back with 100% precision as quoted before, and same-genus precision of mapping reads from a hold-out set is as high as 82%.

EukDetect introduces an element of anticipating potential mismatches - for each genus, a taxon with the most matching reads and greatest coverage is considered the primary for its genus, and a more stringent burden of evidence is placed on any other results in the same genus. It is shown that together with a filter of minimum read count, the method can distinguish a true mixture from off-target hits when simulating reads from two closely related species of *Entamoeba* at a wide ranges of coverage.

Our simulations show that the difficulty of distinguishing species from each other might be highly variable.

For each taxon, we count reads sampled from the sequences of that taxon that align back to it, compare it to the count of sequences sampling to each other taxon, and sort by the ratio of the two. *E. dispar* is mismatched as *E. histolytica* only once compared to 75 reads, and in general simulated *Entamoeba* reads turn out to not be particularly difficult to map back to their source - among 7172 pairs, the the *E. nuttali* reads being aligned to *E. dispar* at a ratio of 0.08 are the highest on the list at position 940, a far cry from a 0.86 ratio for two brown algae *Ectocarpus* sp. Ec32 and *Ectocarpus siliculosus*.

Also, coming back to the MAPQ stuff - the mechanism by which the MAPQ ≥ 30 filter improves precision is revealed when we look at precision vs. MAPQ for each source species, for reads sampled without added mutations.

Basically, different species can be more or difficult to tell from other, similar ones. MAPQ is a proxy measure for that - except it's not a reliable way to tell that a match is wrong, MAPQ is frequently low for good matches and when a source species is not in the reference there are plenty of high MAPQ matches to incorrect results. The task of mapping a 100bp read back to its source is in general very easy for a modern aligner, requiring MAPQ ≥ 30 dings everything that has a chance of being wrong - which for some species means throwing away most of the reads - and that's the secret of 99.5%+ precision on simulated reads.

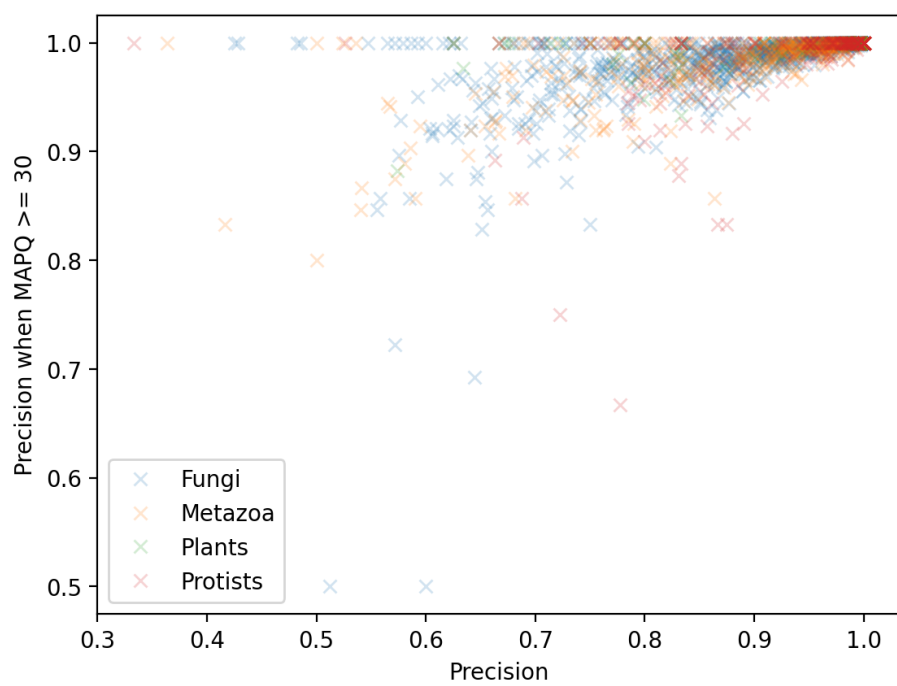


Figure 3: Precision and fraction of reads with MAPQ ≥ 30 , each dot is source taxon

Sequence space stuff

Maynard Smith, John (7 February 1970). “Natural Selection and the Concept of a Protein Space”. *Nature*. 225 (5232): 563–564. Bibcode:1970Natur.225..563M. doi:10.1038/225563a0. PMID 5411867. S2CID 204994726.

- the landscape of existing proteins is made up of islands of related sequences
- mapping reads corresponds to a problem of finding the nearest neighbour
- explains variability of success:
- suggests link between correctness of match and identity?

Filtering on the MAPQ field does not originate with EukDetect - it is also present in MetaPhlAn (Segata et al. 2012).

Tools for read mapping like `bowtie` or `samtools` have been developed in the context of reference genomes like the human genome (Langmead et al. 2009), (Li et al. 2009). The SAM specification originating with `samtools`, which `bowtie` and `bowtie2` follow, defines the MAPQ field as a measure of certainty about position of the alignment.

The variance in trade-offs offered by the MAPQ ≥ 30 filter can be In the context of nearest

MAPQ is a measure of ambiguity, and decreases when there are several nearby proteins or when there

Tools for read mapping like `bowtie` or `samtools` have been developed in the context of reference genomes like the human genome (Langmead et al. 2009), (Li et al. 2009). The SAM specification originating with `samtools`, which `bowtie` and `bowtie2` follow, defines the MAPQ field as a measure of certainty about position of the alignment. Eukdetect aligns metagenomic reads with `bowtie2` to a reference which is not like a single reference genome - it contains groups of similar sequences from many genomes - and one of the filters it applies to the alignments is to require MAPQ ≥ 30 .

Bornberg-Bauer, E (Nov 1997). “How are model protein structures distributed in sequence space?”. *Biophysical Journal*. 73 (5): 2393–403. Bibcode:1997BpJ...73.2393B. doi:10.1016/S0006-3495(97)78268-7. PMC 1181141. PMID 9370433.

Network stuff

todo this is not yet coherent

To make use of alignments which are only mostly correct, we need a source of information on what markers and taxa are likely to be confused with each other.

We investigate an alternative approach of using secondary alignments.

EukDetect introduces a concept of primary and secondary hits: a taxon with the most matching reads is considered a primary taxon for its genus, and a more stringent burden of evidence is placed on any other (secondary) results in the same genus.

It works well when simulating mixtures of species at even low coverage, but fails to account for an unknown species present at a relatively high coverage - see the quick cross-validation result at the top.

We believe that off-target alignments need a more nuanced treatment.

Diagram

We have a theoretical model presented by these two diagrams:

Diagram 1a. With X' and X'' differing from X only by aggregated single-nucleotide changes, reads preferentially map to the closer X'. MAPQ and percentage identity get lower, but the information in the reads is sufficient to report the nearest taxon.

```

    --- X''  AACTAACAATATAA
   /
  \----- X'  AACAAACAAAAATA
   \----- X  AAAAAAAAAAAAAA

```

Diagram 1b. If X' misses some of the sequence, some reads map to X'' instead.

```

    --- X''  AACTAACAATATAA
   /
  \----- X'  AACAA_____AAATA
   \----- X  AAAAAAAAAAAAAA

```

Measuring network structure

Some markers get confused with each other more often than others, specifically, an off-target hit is likely to be a hit to a related protein from a closely related organism. This is a basis of EukDetect's `eukdetect/count_primary_and_secondary.py` program, which uses a partition of markers based on their taxonomic origin.

EukDetect's method corresponds to an error model where all equivalently annotated BUSCOs in genus level are considered confusable enough to require the more stringent cutoffs, and off-target hits on differently annotated BUSCOs or in different families are considered insignificant enough to be removed by other filters.

We can compare how well this partition captures markers in each simulated result by using the modularity formula:

$$Q = \text{fraction of edges within clusters} / \text{fraction of edges between clusters}$$

applied to a graph where each node is a marker and each edge weight is a count of mis-classifications.

We can also compute partitions using the MCL algorithm frequently deployed in bioinformatics in the context of sequence match, as well as the Leiden algorithm, a general-purpose tool for community detection.

Here are the results: TODO :).

As we see (TODO) the performance of the per-genus partition varies based on error rate introduced to the reads, and is not as good as an optimal partition calculated using the ground truth information.

We do not need an a-priori list of markers that might get confused with each other: it can be observed by setting the aligner to report all alignments per query, and using counts of shared alignments as edge weights between markers, then running the same algorithms as before.

It depends on coverage, but here are some results: TODO.

Summary of our method

This is the method

It is unfortunately really ad-hoc, how do I make it less ad-hoc?

TODO say it's research software :) We show using identity and all alignments might be better, we have a PYthon package for exploring this kind of thing.

bowtie2

We set **bowtie2** to report multiple alignments per query. This lets us add structure on the level of markers: - count reads aligned multiply for each pair of markers as proxy for similarity - use the MCL algorithm to produce clusters - use average alignment identity as proxy of distance from “true” sequence - use count of reads as evidence threshold and similarly on the level of taxa.

This lets us set up a sequence of filters similar to EukDetect without relying on the MAPQ field.

KMA

We set KMA to report all assembled fragments of reads for each template.

We process the fragment with an all-to-all nucleotide BLAST at 97% identity. For each pair of fragments that match, we compute their similarity as a fraction of query length to template length. Then we run MCL.

We have found KMA helpful in investigating samples that are dominated by a novel unknown taxon, similarly to the tool’s intended use in genomic epidemiology. Its additional capabilities are of little use when detecting rare species - say a eukaryote that only contributes a small number of reads to the metagenomic sequence - and the resource costs are formidable, so we have not further investigated its use.

Software implementation

to do:

This section is about the software

Figure out what should actually be here

We implement our way of interpreting results of alignments to markers as a Python package, **marker_alignments**. It uses a module **pysam** to read alignments into a following SQLite table:

column	type	description
query	text	read identifier
taxon	text	name of matched taxon
marker	text	name of matched taxon
coverage	number	fraction of marker covered by match
identity	number	fraction of bases agreeing between query and reference

Counts of entries and the coverage field are used for quantification, and identity is mainly used for clustering and filtering.

Filtering and reporting is implemented with SQL queries and Python code.

Clustering is based on a `markov_clustering` package, a Python implementation of the MCL algorithm.

In addition we also provide a Nextflow workflow, `marker-alignments-nextflow`. All our software is freely available on GitHub under an MIT license.

Examples

What would make a good example?

A good example would be specific to the improvement in the method: something about better specificity, or reporting variation in species beyond what SNPs can produce.

Maybe reporting those missing pieces of protein from diagram 2?

Interesting cases I came along

Example 1 - off-target hits in more than just a genus

SRR6262267 is a run from a sample dominated by *Trichosporon asahii* - according to SRA Traces, 23.68% of the reads in the sample can be attributed to this organism (source).

TODO an example - perhaps a heatmap of counts for BUSCOs in each *Trichosporon*?

Actually, EukDetect reports only *T. asahii*, because the other off-target hits are for the same genus.

TODO this will require some visualisation tools.

Example 2 - sticking to the reference too closely brings up nothing

Mucor example, demonstrate EukDetect returns nothing which it really should.

Example 3 - BUSCOs vs clusters

Maybe a drawing or a visualisation: a graph with BUSCOs corresponding to a shading of each node, added edges, and a shading or a line around clusters that end up together?

Potential application 1 - host blood meal

Identify host blood meal in metagenomic studies of mosquitos.

Potential application 2 - AMR

Detection/quantification of antimicrobial resistance (AMR) genes in metagenomic samples

MicrobiomeDB studies

to do:

This is about MicrobiomeDB results.

Decide if they should be a part of the paper and how.

We analyzed all data on MicrobiomeDB.

study	num samples	num samples with reported taxa	num reported taxa
HMP	xxx	yyy	zzz

DIABIMMUNE

to do:

This is an expanded part of the above that does a comparison for DIABIMMUNE.

Write a program to report the numbers, for transparency and because you will need to redo it

Make a point that our method improves sensitivity, and to have a comparison, maybe independent

Here's a not yet formal comparison to DIABIMMUNE. DIABIMMUNE reports

Just the paper DIABIMMUNE reported 8 samples. I think it's mostly the difference between references - new one has a bit more. We're more courageous in reporting species, and report 119 extra results. It shows up for the most common *S. cerevisiae* - we call it 42 more times - and *M. restricta* - 9 more times.

3100266

Here's a case where we do something weird. EukDetect reports *P. nordicum*, but we realise it's a cluster:

id	taxon	num_markers	num_reads	avg_identity
1	1429867 <i>Penicillium_camemberti</i> _FM_013	34	73	0.960183806201549

1	1439350	Penicillium_fuscoglaucum_FM041	33	74	0.962397902985074
1	1931374	Penicillium_sp_BW_MB	101	243	0.958158950450451
1	1931375	Penicillium_sp_BW_12	10	31	0.959146928571428
1	229535	Penicillium_nordicum	45	102	0.954239219101123
1	2488753	Penicillium_sp_SPG-F1	10	29	0.964647588235294
1	2593313	Penicillium	124	313	0.967739331578951
1	48697	Penicillium_freii	117	288	0.959325344168261
1	5073	Penicillium	306	359	0.962057904903424
1	60169	Penicillium_polonicum	41	78	0.959593049295774
1	60171	Penicillium_verrucosum	40	70	0.955100714285714
1	60172	Penicillium_solitum	39	81	0.959004702702702
2	5480	Candida_parapsilosis	8	10	0.996333894736842

and because they're all above 95%, we return them all. What I would prefer is to return the whole cluster, but how should I know?

3104340

Here's a case where we do something good.

G80329

diabimmune-paper.tsv.csv:G80329,Candida parapsilosis,5480,27,78,15.70%,8.04%,99.64%

diabimmune-paper.tsv.csv:G80329,Rhodotorula,5533,3,5,3.53%,3.76%,100.00%

diabimmune-paper.tsv.csv:G80329,Chaetomium globosum CBS 148.51,306901,3,5,1.69%,4.14%,99.79%

diabimmune.tsv:3104340 G80329 237561 Candida_albicans_SC5314 8

diabimmune.tsv:3104340 G80329 306901 Chaetomium_globosum_CBS_14851 3

diabimmune.tsv:3104340 G80329 5480 Candida_parapsilosis 19

diabimmune.tsv:3104340 G80329 5533 Rhodotorula 3

The above does not show that, but we are able to show that the presence of additional *C. albicans* is very convincing - the hits are in entirely different marker clusters.

Not in the paper - old

I've written these sections but they're not really useful. They might be good for talking about the stuff internally.

Same-genus precision is very high, same-family even higher

Our main result is confirmation of general validity of read mapping when applied to sequences that might exhibit this kind of difference. We see that mutated reads, even as they get aligned less frequently, overwhelmingly map to the taxonomic unit they were sampled from - and same-genus precision is between 99.5% and 99.6%.

Average match identity of reported alignments decreases as reflecting the mutations introduced, and average MAPQ drops to zero.

In the SAM specification (Li et al. 2009) the MAPQ field was defined as a measure of mapping quality - certainty of where the read should be positioned in the reference genome. EukDetect uses this field as a filter on evidence about presence of different species in a metagenomic sample.

We see in Figures 1 and 2 that adding the MAPQ filter improves the precision to between 99.6% and 99.9%, at the cost of making recall much less robust.

- The main result is that read mapping mostly works when mapping sequences from an unknown species: same-genus precision is 82%, same-family precision is 95%, same-genus recall is 30%, and same-family recall is 35%. Adding the MAPQ filter doesn't improve precision, but vastly decreases recall.

Introduction

to do:

This claims that reference bias is a problem in read mapping, and particularly for eukaryotes

It's a working theory at best

add citations / context or get rid of it

Methods of quantifying taxa in metagenomic samples that based on read mapping suffer from reference bias.

Mechanisms of variation are different between prokaryotes and eukaryotes. Bacteria lose whole genes and functions, but also gain them through recombination. Eukaryotic genes have introns, so the tweaking is more gradual, and small differences in sequence lead to very different phenotypes.

In eukaryotes, mapping reads for an organism that is only broadly characterized in the reference (a different strain or species) results in reference bias. The organism might be skipped, or reported as a mixture of a closely related species.

We show reference bias can be mitigated by incorporating secondary alignments as evidence against some taxa being present. We show this increases sensitivity

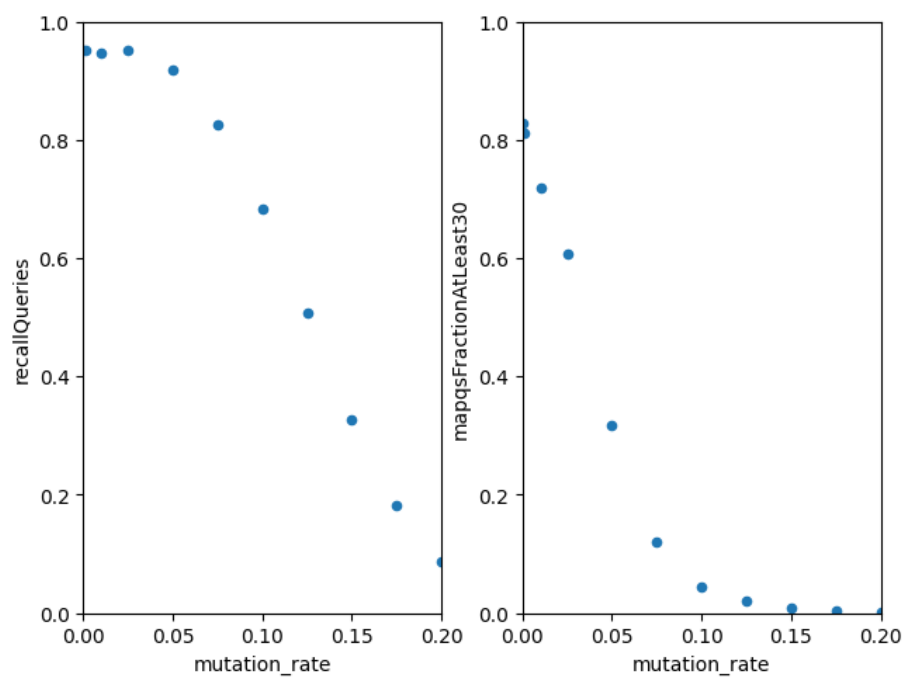


Figure 4: wgsim mutation rate - mapq drops before recall

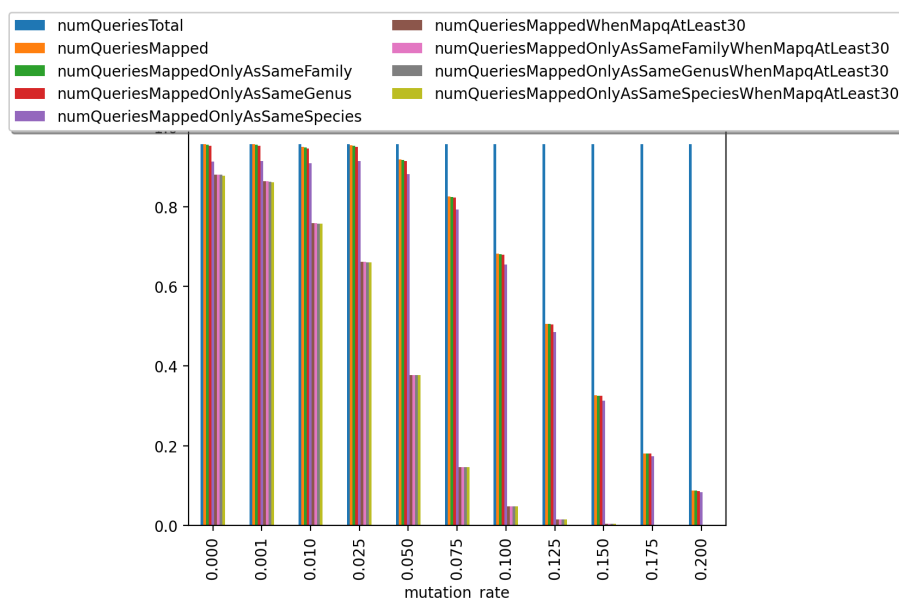


Figure 5: wgsim mutation rate - the precision is high, MAPQ ≥ 30 improves precision yet more at a large cost to recall

when mapping to a EukDetect reference of BUSCOs and replacing the MAPQ filter. We also show we can identify off-target hits in the output of CCMetagen.

Background

to do:

these are loose notes about things I've heard about

explain how our new thing is new in relation to those things

Reference bias

Quantifying Eukaryotes is hard

1. Euk genomes are widely contaminated (Lind and Pollard 2021) so bacterial reads match to them spontaneously.
2. The level of noise is high enough that reads to e.g. fungal sequences completely fail. The use of taxon-specific reference databases compromises metagenomic classification

3. The Euk genomes are larger and differ from each other by less
4. The references are quite spotty: there's like a thousand assemblies, 148,000 described, 2-3 mln estimated <https://en.wikipedia.org/wiki/Fungus>.
5. Telling apart multiple species in the sample: an inexact match to a reference looks like a mixture of related species

Current tools

Finding taxa in a metagenomic sample can be achieved by mapping reads to a reference database. K-mer methods (Kraken) and de novo assembly (anvio) are possible alternatives to read mapping.

Out of K-mer methods, CCMetagen(Marcelino et al. 2020) stands out as an alternative new tool developed for genomic epidemiology use. It is based on a specialised aligner KMA(Clausen, Aarestrup, and Lund 2018) that handles redundant databases, so it can use all possible genomes as its reference. Further it does reference-guided assembly on the reads, and assigns each sequence a separate match. Unfortunately it does not attempt to provide a list of species in the sample, but rather, a summary of naturally redundant KMA results into taxonomic units.

Metaphlan is the most established tool for running `bowtie2` on a reference of marker genes. Massive reference, ~1.1M unique clade-specific marker genes identified from ~100k reference genomes (~99,500 bacterial and archaeal and ~500 eukaryotic).

Kaiju uses protein alignments, trying to get higher sensitivity.

EukDetect achieves sensitivity through aligning to a reference of Eukaryotic BUSCOs and heavy filtering. The possibly-ambiguous reads are assigned low MAPQ scores.

Conclusion

to do:

this tries to explain why what I did is good and clever

use it to structure further work

Our contributions are:

- an explanation of how off-target hits happen, which helps authors of tools that interpret read mapping as counts of taxa

- software that runs and interprets alignments to EukDetect’s reference database, and possibly to other reference databases, which helps people who have metagenomic data and need to analyse in taxa present
- the analysis of MicrobiomeDB data, which helps people who want to know what eukaryotes are present in human microbiomes

We’ve not yet contributed, but could contribute:

- a better way of integrating multiple results for a taxon
- software for building references that will work well with our method, which would help bioinformaticians in setting up analyses like ours
- a reference for traces of animal DNA, which would help people who study samples where they might be present
- an analysis of host blood meals of a mosquito dataset, which would help epidemiologists and people who want to know what different mosquitoes feed on

We’ve had ideas about:

- using this work to detect anti-microbial resistance genes (is there a universal database of known AMR variants of genes?)

Our method is new because:

- nobody else interprets low MAPQ reads as a good alignment in the wrong place
- nobody else interprets secondary alignments as evidence against taxon being present
- nobody else uses network methods

Our method is good because:

- the method works at very low abundances, it’s even better at it than EukDetect
- the method reports mixtures of related species more sensitively than EukDetect
- the method does not skip, or bias counts against, species that only approximately match the reference

Our method would be even better if:

- the method modelled the gap between reference and signal
 - reporting it could be interesting
 - one clear filter for taxa could be more accurate than a few sequential ones
 - an explanation how the gap between reference and signal looks in read mapping results when they’re summarized by taxon

Definitions

taxon - an organism, in this context an organism that had its genome sequenced, appears in the reference database, and may or may not be present in the sequenced sample

BUSCO - a family of genes that are mostly present in each taxon and mostly single-copy

marker - a DNA sequence of a gene in a taxon that is assumed to be unique to the taxon. Markers in related taxa can be similar, for example if they belong to the same BUSCO

reference database - one of the inputs for an aligner, in this context it's a reference of markers that can be matched to

alignment / hit - a read in the sequenced sample that was found by an aligner to match a marker in the reference

Aligners like **bowtie2** can report multiple alignments per read. In that case, we can distinguish:

primary alignment - the best match for the read (based on alignment score, sequence identity, or other metric)

secondary alignments - alignments corresponding to matches for a read that are not the best one

We would also like to differentiate between:

target hit - a read that comes from an organism A , and matches a marker M' for a taxon A' , such that A' is the closest taxon to A

off-target hit - a read that comes from an organism A , but matches a marker M' for a taxon A'' , and there is at least one taxon A' closer to A than A'' is

Sources of off-target hits

to do:

this part is all anecdotal

it's about hits I saw in the data that I think are off-target and that I think I understand

add evidence to it or get rid of it

Even if the reference contains all taxa that might be in the sample, there are still some possible sources of off-target hits, like:

1. random bias: the sequencing process introduces errors and short sequences can coincide by chance

2. ubiquitous subsequences: in end-to-end alignments, these look like clipped alignments to where a reference finishes with a sequence corresponding to e.g. a binding site common in proteins
3. missing markers: reads for a marker that is missing in the reference instead align to similar markers

Our method addresses these kinds of off-target hits through a combination of filters or thresholds on read quality and length, alignment length, and numbers of markers required to detect a taxon.

An entirely different kind of off-target hits is due to a sequenced organism having no exact match in the reference. We will describe how this happens, and how it can be addressed by making use of secondary alignments.

A model for off-target hits due to inexact matches

to do:

this is my conjecture for how off-target hits can contain signal:

- an inexact match looks in alignments as a mix of related matches
- asking for best alignment sets up competitive mapping and that's not good
- secondary alignments are good for the next part

make it a cartoon or less long

merge with the next part (marker clusters)

show evidence that it happens, and that it happens in the way described

Suppose an organism A has a version b_A of a BUSCO b , and the reference contains markers b_{A_1}, \dots, b_{A_n} for taxa A_1, \dots, A_n . Let us say A is most similar to A_1 - perhaps it's another strain of the same species. Assume also a least common ancestor A_0 of A and A_1 , and A_{00} of A_0, A_2, \dots, A_n . As mutations accumulate over time, we can predict b_A will be most similar to b_{A_1} , but - in places where A_1 has diverged from A_0 - some segments of b_A are most similar to other b_{A_i} . Some segments of b_A could also be equally similar in all b_{A_i} , if there has been reason for that sequence not to change since the joint common ancestor A_{00} .

When reads from b_A are aligned to each of the b_{A_i} , we expect match identity to form a distribution. The A_i which differ more from A should have to lower average match identity and fewer matches, but might still attract high identity matches.

So, competitive alignment of reads from b_A between $b_{A_1} \dots b_{A_n}$ will not entirely favour the closest A_1 .

If we ask an aligner to report a single best alignment for each read, we expect to see h_1 hits to b_{A_1} , and smaller amounts h_i of hits for other b_{A_i} , such that $H = \sum_{i=1}^n h_i$ is proportional to the count of reads coming from sequencing $b(A)$.

We also expect ratios of H to h_i to be related to sequence similarity between b_A and b_{A_i} - the further A is from A_1 , the larger the number of off-target hits $H - h_1$.

The effect of $b_{A_1} \dots b_{A_n}$ ‘competing’ for the best alignment of each read is illuminated when an aligner is asked to report all reads. Some of $h_2 \dots h_n$ are then accompanied by secondary alignments to b_{A_1} - call them s_1 , and some of h_1 will be accompanied by secondary alignments to $b_{A_2} \dots b_{A_n}$, $s_2 \dots s_n$. If h_1 is much larger than h_i , s_1 should be much smaller than s_i and thus $\frac{h_1}{s_1}$ should be larger than $\frac{h_i}{s_i}$ and independent of H .

Thus secondary alignments help us differentiate the presence of an organism A reported as many hits to A_1 and fewer hits to A_2 from the presence of two unrelated organisms X and Y . For example, we can report a ratio of primary to secondary alignments for each taxon.

It is possible for b_{A_1} to be very different from b_A , or missing from the reference entirely. Perhaps the genome of A_1 is incorrectly annotated, or A_1 has lost b when adapting to its niche.

Building marker clusters

to do:

this introduces marker clusters

it tries to theorise what kinds of clusterings there can be
EukDetect uses a taxonomic clustering but we do an empirical one

remove most of it, it’s incredibly wordy
add a visualisation of marker clusters in examples

The effect of including secondary alignments is to only add hits to sequences similar to ones already present - after all, they both match on a read. Because identifying off-target hits requires grouping similar markers, secondary alignments provide valuable context for what markers are generally similar to what is present in the sample.

In our method, we run an aligner with as many secondary alignments as we can computationally afford, and then build a similarity graph where all matched markers are nodes and counts of reads that align to both markers are weighted edges. We then pass the triples (marker1, marker2, weight) to a clustering program, MCL.

Using a machine learning program like MCL relieves us from more precise modelling of what it means for two markers to be similar, or relying on prior information on what should be matched together.

MCL produces clusters with several valuable properties:

- markers in a cluster generally come from a single BUSCO and closely related taxa, but not always
- broadly similar markers are grouped in larger clusters
- unique hits correspond to clusters with single markers
- a marker sharing reads with multiple putative clusters either gets assigned to one of them, or results in two clusters being merged

Reporting taxa

to do:

this is about how the software makes calls on the taxon level
 currently we sequentially apply a few filters and transforms
 (this doesn't even describe them all)
 the unknown taxon transform doesn't work as well as it should yet
 get some clarity on the method and try to improve it?

Filter built from marker clusters

Clusterings of markers which attract similar matches allows for classifying taxa by having each marker cluster “vote” for its taxon, based on how that taxon's markers do in that cluster. This can be done in a number of ways as long as the marker clusters are required to only be approximately correct.

We choose to measure how good an alignment is through match identity - a number of bases that agree between query and reference divided by the alignment length. Then for each marker cluster, we compute an average for all matches in the cluster, as well as an average for matches in each marker. Then we discard taxa where less than half of each taxon's markers that are at least average in their cluster.

We can expect that this filter will work well enough to discard the additional taxa introduced to the result when setting the aligner to report secondary alignments, since they are on average inferior. Similarly, a version of a BUSCO that is overall inferior but has a locally better subsequence can be expected to accrue bad matches, get paired up with overall better versions of the BUSCO in the marker clustering process, and then help get its taxon rejected.

The filter based on marker clusters leaves us with taxa whose markers are not mostly off-targets of other markers in the reference. There can still be a gap between reference and signal we might get a mixture of matches at a marker level.

Taxon transform + identity filter

We build another graph where taxa are nodes and (directed) edges is a fraction of shared reads. Then we run MCL again to get taxon clusters.

This groups taxa which are similar and, hopefully, each corresponds to at most one taxon.

Minimal evidence filter

After that there's still a need for general filtering like EukDetect and Metaphlan do.

We do it at the end and have separate thresholds for called taxa and vaguely identified blobs, as the burden of evidence is higher for taxa with lower identity.

This part is a combination of read counts, marker counts.

Other tools

OPAL is a benchmark: <https://genomebiology.biomedcentral.com/articles/10.1186/s13059-019-1646-y>

MetaPhyler is like Metaphlan, but uses BLAST.

TIPP inserts each read into a taxonomic tree and uses the insertion location to identify the taxonomic lineage of the read. The novel idea behind TIPP is that rather than using the single best alignment and placement for taxonomic identification, we use a collection of alignments and placements and consider statistical support for each alignment and placement. <https://github.com/TeraTrees/TIPP>

k-mer based: CommonKmers <https://github.com/dkoslicki/CommonKmers/>

<https://academic.oup.com/bioinformatics/article/29/17/2096/239249> quikr

motu: <https://motu-tool.org/> 40 universal marker genes

focus: <https://peerj.com/articles/425/> <https://onestopdataanalysis.com/metagenome-profile/> k-mer method. Really good distinction between binning and profiling!

#Bibliography

- Clausen, Philip TLC, Frank M Aarestrup, and Ole Lund. 2018. “Rapid and Precise Alignment of Raw Reads Against Redundant Databases with KMA.” *BMC Bioinformatics* 19 (1): 1–8.
- Gibson, Molly K, Bin Wang, Sara Ahmadi, Carey-Ann D Burnham, Phillip I Tarr, Barbara B Warner, and Gautam Dantas. 2016. “Developmental Dynamics of the Preterm Infant Gut Microbiota and Antibiotic Resistome.” *Nature Microbiology* 1 (4): 1–10.
- Hawksworth, David L, and Robert Lücking. 2017. “Fungal Diversity Revisited: 2.2 to 3.8 Million Species.” *Microbiology Spectrum* 5 (4): 5–4.
- Hovhannisyan, Hrant, Ahmed Hafez, Carlos Llorens, and Toni Gabaldón. 2020. “CROSSMAPPER: Estimating Cross-Mapping Rates and Optimizing Experimental Design in Multi-Species Sequencing Studies.” *Bioinformatics* 36 (3): 925–27.
- Huerta-Cepas, Jaime, François Serra, and Peer Bork. 2016. “ETE 3: Reconstruction, Analysis, and Visualization of Phylogenomic Data.” *Molecular Biology and Evolution* 33 (6): 1635–38.
- Katz, Kenneth S, Oleg Shutov, Richard Lapoint, Michael Kimelman, J Rodney Brister, and Christopher O’Sullivan. 2021. “STAT: A Fast, Scalable, MinHash-Based k-Mer Tool to Assess Sequence Read Archive Next-Generation Sequence Submissions.” *Genome Biology* 22 (1): 1–15.
- Kriventseva, Evgenia V, Dmitry Kuznetsov, Fredrik Tegenfeldt, Mosè Manni, Renata Dias, Felipe A Simão, and Evgeny M Zdobnov. 2019. “OrthoDB V10: Sampling the Diversity of Animal, Plant, Fungal, Protist, Bacterial and Viral Genomes for Evolutionary and Functional Annotations of Orthologs.” *Nucleic Acids Research* 47 (D1): D807–11.
- Langmead, Ben, and Steven L Salzberg. 2012. “Fast Gapped-Read Alignment with Bowtie 2.” *Nature Methods* 9 (4): 357–59.
- Langmead, Ben, Cole Trapnell, Mihai Pop, and Steven L Salzberg. 2009. “Ultrafast and Memory-Efficient Alignment of Short DNA Sequences to the Human Genome.” *Genome Biology* 10 (3): 1–10.
- Li, Heng. 2011. “Wgsim-Read Simulator for Next Generation Sequencing.” *Github Repository*.
- Li, Heng, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer, Gabor Marth, Goncalo Abecasis, and Richard Durbin. 2009. “The Sequence Alignment/Map Format and SAMtools.” *Bioinformatics* 25 (16): 2078–79.
- Lind, Abigail L, and Katherine S Pollard. 2021. “Accurate and Sensitive Detection of Microbial Eukaryotes from Whole Metagenome Shotgun Sequencing.” *Microbiome* 9 (1): 1–18.
- Marcelino, Vanessa R, Philip TLC Clausen, Jan P Buchmann, Michelle Wille, Jonathan R Iredell, Wieland Meyer, Ole Lund, Tania C Sorrell, and Edward C Holmes. 2020. “CCMetagen: Comprehensive and Accurate Identification of Eukaryotes and Prokaryotes in Metagenomic Data.” *Genome Biology* 21 (1): 1–15.
- Meyer, Fernando, Andreas Bremges, Peter Belmann, Stefan Janssen, Alice C McHardy, and David Koslicki. 2019. “Assessing Taxonomic Metagenome

- Profilers with OPAL.” *Genome Biology* 20 (1): 1–10.
- Nash, Andrea K, Thomas A Auchtung, Matthew C Wong, Daniel P Smith, Jonathan R Gesell, Matthew C Ross, Christopher J Stewart, et al. 2017. “The Gut Mycobion of the Human Microbiome Project Healthy Cohort.” *Microbiome* 5 (1): 1–13.
- Oliveira, Francison S, John Brestelli, Shon Cade, Jie Zheng, John Iodice, Steve Fischer, Cristina Aurrecoechea, et al. 2018. “MicrobiomeDB: A Systems Biology Platform for Integrating, Mining and Analyzing Microbiome Experiments.” *Nucleic Acids Research* 46 (D1): D684–91.
- Peng, Xiaoqing, Jianxin Wang, Zhen Zhang, Qianghua Xiao, Min Li, and Yi Pan. 2015. “Re-Alignment of the Unmapped Reads with Base Quality Score.” In *Bmc Bioinformatics*, 16:1–10. 5. Springer.
- R Marcelino, Vanessa, Edward C Holmes, and Tania C Sorrell. 2020. “The Use of Taxon-Specific Reference Databases Compromises Metagenomic Classification.” *BMC Genomics* 21 (1): 1–5.
- Segata, Nicola, Levi Waldron, Annalisa Ballarini, Vagheesh Narasimhan, Olivier Jousson, and Curtis Huttenhower. 2012. “Metagenomic Microbial Community Profiling Using Unique Clade-Specific Marker Genes.” *Nature Methods* 9 (8): 811–14.
- Sims, David, Ian Sudbery, Nicholas E Ilott, Andreas Heger, and Chris P Ponting. 2014. “Sequencing Depth and Coverage: Key Considerations in Genomic Analyses.” *Nature Reviews Genetics* 15 (2): 121–32.
- Smith, John Maynard. 1970. “Natural Selection and the Concept of a Protein Space.” *Nature* 225 (5232): 563–64.
- Turnbaugh, Peter J, Ruth E Ley, Micah Hamady, Claire M Fraser-Liggett, Rob Knight, and Jeffrey I Gordon. 2007. “The Human Microbiome Project.” *Nature* 449 (7164): 804–10.
- Van Dongen, Stijn, and Cei Abreu-Goodger. 2012. “Using MCL to Extract Clusters from Networks.” In *Bacterial Molecular Networks*, 281–95. Springer.
- Vatanen, Tommi, Aleksandar D Kostic, Eva d’Hennezel, Heli Siljander, Eric A Franzosa, Moran Yassour, Raivo Kolde, et al. 2016. “Variation in Microbiome LPS Immunogenicity Contributes to Autoimmunity in Humans.” *Cell* 165 (4): 842–53.