Supplementary Text

Table of Contents

Overview of tools and algorithms 1
Differences between fai and cblaster 4
Parameters used in fai and zol for showcase applications 5
Supplementary text references 10

Overview of tools and algorithms

prepTG - processing and preparing target genomes for searching with fai

prepTG allows users to create a database of target genomes that can be searched for homologous instances of query gene clusters with fai. In addition to formatting and producing files for optimizing fai searches, prepTG integrates pyrodigal[1], prodigal[2], and miniprot[3] for gene-calling or gene-mapping in prokaryotic and eukaryotic genomes as well as metagenomes to aid consistency in fai's performance and limit bias due to potential differences in gene-calling methods. For miniprot-based gene mapping, coding sequence predictions are required to exhibit an identity of at least 80% to the reference protein and instances of overlapping mRNA features are resolved by retaining only the highest scoring mapping.

fai - automated identification of homologous instances of gene clusters

fai allows for rapid detection of gene neighborhoods exhibiting homology to query gene clusters provided by users. It implements HMM-based and gene distance-based approaches for determining homologous gene cluster instances in target genomes, which can further be combined in a hybrid approach. For both approaches, homologs of proteins from query gene clusters are first searched for in predicted proteomes of target genomes using DIAMOND alignment[4]. Then, in "Gene-Clumper" mode, which is the default, scaffolds with homologs of query proteins are dynamically assessed for whether homologs are within a maximum number of CDS predictions to be regarded as belonging to the same gene cluster. In "HMM" mode, scaffolds of target genomes are instead scanned gene-by-gene using an HMM and neighborhoods or sets of genes are regarded as being in a state of homology to the query gene cluster if several individual genes depict homology to the proteins from the query gene cluster(s). The algorithm is similar to *lsa*BGC-Expansion[5], however, it is not dependent on a preliminary genome-wide orthology grouping analysis and thus features a different set of filters to still enable high-throughput automated detection of homologous gene cluster segments as a result. *lsa*BGC-Expansion is reliant on a preliminary orthology analysis to identify BGC-specific genes that could be used to differentiate true homologous instances of BGCs and customize weighting of HMM emission probabilities for distinct genes. It further requires the length of genes within putative homologous regions to be within a certain deviation from the median

length of known gene instances. In contrast, fai has preconfigured emission probabilities which can be customized by users and has no length requirement for potential homologous instances of genes. fai further allows the "HMM-based" approach to be run with the parameter for aggregating CDS predictions for the "Gene-Clumper" mode, whereby, gene cluster segments detected by the HMM can be joined with other such segments if they are withing a certain number of CDS features from each other. Similar to *lsa*BGC-Expansion, syntenic similarity between candidate and query gene cluster segments can also be used to filter candidate segments using a gene cluster-wide correlation metric[5].

By default, fai requires filters pertaining to the number of genes from query gene clusters to be met for each homologous gene cluster candidate segment, but it can identify split gene cluster segments due to assembly fragmentation in "draft mode". In this mode, thresholds for detection of gene clusters within target genomes are assessed in aggregate for gene clusters found near scaffold edges (< 2,000 bp). Visual reports produced by fai showcasing the sequence similarity of target genome proteins to the query protein(s) can then be manually investigated by users to assess the validity of fragmented gene-cluster instances. In addition, fai features an option to filter for paralogous, overlapping candidate segments of a gene cluster in target genomes and offers an intuitive visualization of gene cluster segments, if requested, to allow users to assess their quality, including proximity of candidate segments to scaffold edges. Together, these options enable the large-scale identification of orthologous gene clusters across genomes which can then be leveraged by zol to perform context-specific inference of protein ortholog groups.

zol - computes a variety of evolutionary statistics and can perform gene cluster specific dereplication

The zol workflow begins by processing the input directory of gene cluster GenBanks to assess validity and perform filtering of gene clusters or individual proteins. Filtering can be performed at the gene cluster level by requesting filtering of draft-quality gene clusters, those marked as being near scaffold edges, or low-quality gene clusters, those with ≥10% missing base-pairs (e.g. Ns) in their sequence. Filtering of individual proteins which are near scaffold edges can also be performed if fai was used to identify the input gene cluster set, because fai marks these proteins with a special feature tag in resulting gene cluster GenBanks.

zol will next perform dereplication of gene clusters, if requested by users, with skani[6] by clustering gene clusters which depict some user-defined coverage and identity thresholds using single linkage clustering or more resolved MCL based clustering, for which the inflation parameter can be adjusted. Representative gene clusters are selected from each cluster as part of the dereplication based on maximum length and, if comparative analysis is requested, if the representative gene cluster is part of the focal or focal-complement set of isolates provided by the user.

The input set of gene clusters or set of dereplicated representative gene clusters is then used to identify protein ortholog groups with an InParanoid-type approach[7]. Briefly, DIAMOND[4] is used to perform all vs. all pairwise alignment between proteins from the set of gene clusters after which the alignments are processed to identify reciprocal best hits (RBH) between pairs of gene clusters. In-paralogs are identified within each gene cluster based on whether two coding

sequences depict more similarity to each other than one does to an RBH with a different gene cluster. Bitscores, normalized through division by reflexive bitscore values for query proteins, are used to assess homology. Specifically, the average normalized bitscore between each pair of orthologs and in-paralogs is recorded. Afterwards, bitscores between such protein pairs are further normalized through dividing them with the average values between pairs of gene clusters to aid proper clustering of proteins downstream. This is akin to the genome-wide normalization procedure recommended in OrthoMCL, owing to the realization that orthologs between distantly related species are also more likely to exhibit lower sequence similarity, which should be corrected for prior to MCL clustering[8]. This information is input into MCL with a default inflation parameter set to 1.5. The inflation parameter and minimum identity and coverage cutoffs to consider valid pairs of in-paralogs and orthologs are adjustable by users. Finding of orthologs is designed for and largely tested on gene clusters with fewer than 100 proteins and a maximum of 100 gene cluster instances. To run zol with more than 100 gene clusters, we recommend using the dereplication and re-inflation parameters. This will also increase the likelihood that at least one core homolog group is identified across [dereplicated] gene clusters which is a requirement for continuing forward in the workflow.

Re-inflation can be requested by users following dereplication by expanding ortholog groups to include proteins from the full input set of gene clusters. This is done by first performing comprehensive and granular clustering of proteins from all input gene clusters using CD-HIT[9], requiring proteins to depict >98% sequence similarity and > 95% bi-directional coverage to the representative sequences of clusters. Proteins in CD-HIT clusters are then mapped to ortholog groups if they co-cluster with proteins from dereplicated gene clusters which are already assigned to ortholog groups. Dereplication and re-inflation are not recommended if sequence redundancy amongst the set of input gene clusters is low. Stringent cutoffs used for CD-HIT clustering during re-inflation are based on the assumption that dereplication was also run with stringent parameters to only collapse highly similar gene clusters. Otherwise, re-inflation could miss more distant instances of ortholog groups, resulting in an underestimation of ortholog group conservation amongst gene clusters.

Next, zol will partition protein and nucleotide sequences from gene clusters according to ortholog groups, perform protein alignment using MUSCLE[10], and create codon alignments using PAL2NAL[11]. We also offer an option to use reference proteins to refine and filter sequences based on multiple sequence alignment using MUSCLE[10], which might be useful to further filter intronic sequences in eukaryotic ORFs. Codon alignments are filtered for regions with high ambiguity ($\geq$10% gaps) using trimAL[12] which are then used downstream for calculation of evolutionary statistics and ortholog group approximate phylogenies constructed with FastTree 2[13]. Consensus protein sequences for each ortholog group are finally constructed using HMMER3[14].

Using protein consensus sequences of each ortholog group, zol is next able to linearize annotation of ortholog groups with various annotation databases including KOfam[15], the PGAP database[16], VFDB[17], CARD[18], MIBiG[19], ISfinder[20], and Pfam[21]. A custom FASTA file can also be provided by users to annotate ortholog groups. The best hit per ortholog group for each annotation database is selected by bitscore and a default E-value cutoff of 1e-5. The E-value of the alignment is provided in the zol report for each putative annotation.

Next, zol will compute basic statistics per ortholog group including the consensus order, consensus directionality, whether proteins are single-copy across gene clusters, the median length of ortholog group sequences, and their median GC% percentage and GC skew values. The consensus order and directionality are performed similarly to *lsa*BGC-PopGene[5]. Afterwards, in the sixth step, zol will calculate evolutionary statistics for each ortholog group including Tajima's D[22], the proportion of filtered codon alignments which correspond to segregating sites, the average sequence entropy of the filtered codon alignment and the 100 upstream region, and the median and maximum Beta-RDgc. Beta-RDgc is a statistic that is derived from the Beta-RD statistic which we described in *lsa*BGC[5] and measures the divergence of a pair of protein sequences based on the expected divergence between the gene clusters. Values below one suggest that protein divergence is larger for the pair than expected based on other shared proteins between the two gene clusters; conversely, the opposite trend might suggest high conservation of the particular protein between the gene clusters and potentially gene-specific horizontal gene transfer. Finally, we perform site-specific selection analyses using the FUBAR[23] and GARD[24] methods offered in the HyPhy suite. While highly scalable relative to comparable methods[23], these analyses can still take considerable time and are turned off by default. Importantly, GARD recombination detection[24] and partitioning of input alignments for ortholog groups can also be used for alternate HyPhy analyses with HyPhy Vision[25], extending beyond the site-specific selection analyses using FUBAR[23] supported directly in zol.

The step prior to generation of a report is to perform comparative analysis for a user-defined set of focal gene clusters to the complementary set or a user-defined complement-set of gene clusters. In these comparative analyses, the conservation of ortholog groups is computed for each set separately and F_{ST} is calculated for the focal set of gene clusters[26].

Finally, we generate a consensus report and a spreadsheet in XLSX format where each row corresponds to an ortholog group and columns correspond to basic statistics, evolutionary statistics, and annotation information. Quantitative fields are automatically colored to make visual inspection of patterns more digestible for users. A basic heatmap showing the presence of ortholog groups across gene clusters is also produced.

Differences between fai and cblaster

Similar to fai, cblaster[27] is a software aiming to enable high-throughput identification of homologous gene-clusters. However, each program has a different range of functionalities and options which gives them unique advantages depending on research objectives and use-cases (Table S1). Both cblaster and fai take as input a query gene-cluster and use DIAMOND to rapidly search for homologous proteins in target genomes. In addition, fai can also take coordinates along a reference genome or GenBanks of multiple known instances of the gene-cluster as input. The latter two input formats preserve gene-order information in query gene-clusters and allow users to directly assess and filter for syntenic similarity of candidate homologous gene clusters to the reference queries using a correlation based methodology[5].

Unlike cblaster, fai performs a preliminary step to remove redundancy in query proteins using CD-HIT[9] or, if multiple gene-clusters are provided as a single query, the ortholog grouping algorithm used with zol. Bitscores are then used to assign the best matching non-redundant

query protein clusters to coding sequences in target genomes. In contrast, cblaster searches for matches for each query protein independently and can thus assign multiple query proteins to the same coding sequence in the target genome. This can lead to unintended results when the query gene cluster involves homologous proteins, such as a BGC featuring recently duplicated non-ribosomal protein synthetases (NRPSs)[28]. Additionally, while cblaster and fai both rely on user-defined E-value cutoffs to identify homology, cblaster further allows filtering using percent identity and coverage cutoffs. A coverage cutoff option is not provided within fai because differences in gene-calling between the reference gene-cluster and target genomes might result in a query protein not being fully represented by a predicted coding sequence (CDS) in target genomes (Figure S1A).

Another key difference between cblaster and fai is the approach taken to delineate candidate gene-clusters. cblaster provides an innovative analytical mode to infer the appropriate distance to allow between homologous proteins to query proteins to regard them as belonging to the same gene cluster. fai provides two different approaches for gene-cluster delineation, one, referred to as the “Gene-Clumper” approach, which is similar to the approach taken by cblaster and simply groups together genes into the same cluster if they are separated by *M* or fewer CDS features, where *M* is by default set to 5. The other approach is based on a Hidden Markov Model (HMM) with tunable customizable transition and emission probabilities[29] which can be used to first identify smaller sets of gene-clusters and then aggregate those with the maximum CDS separator parameter described for the “Gene-Clumper” approach (Figure 1AC).

Finally, cblaster, and the web-application CAGECAT[30] - which runs cblaster, offer remote searching of BLAST databases, which fai does not support. In contrast, fai and prepTG, the software used for target genome database construction, feature unique options and capabilities ranging from integrated gene-calling or gene-mapping software to finding gene-clusters split due to assembly fragmentation, which we detail in the Results section “fai and zol allow for the rapid inference of gene-cluster orthologs across diverse genomes”.

Parameters used in fai and zol for showcase applications

Parameters used for “Application of fai and zol to identify phages within metagenomes”

Parameters used for fai analysis:

```
Input directory with Gene-Cluster GenBanks: Query_Phage/
Reference Genome for Gene Cluster: None
Reference Scaffold for Gene Cluster: None
Reference Start Coordinate of Gene Cluster: None
Reference End Coordinate of Gene Cluster: None
Protein Queries: None
Target Genomes Prepared Directory by prepTG: prepTG_database/
Output Directory:
/home/salamzade/zol_development/showcase_examples/Phage_in_Lake_MGs/fai_Results/
Run in Draft-Assembly Mode?: False
Filter for Paralogous/Gene-Content Overlapping Segments: True
E-value cutoff for Detection of general Protein Homologs in Genome: 1e-10
Minimum Proportion of Hits for Gene Presence in Genome: 0.5
FASTA file with Key Proteins to Consider: None
E-values for Key Proteins to be Considered as Smoking Gun for Gene Cluster Presence: 1e-20
Minimum Proportion of Key Protein Hits Required for Gene Cluster Presence in Genome: 0.0
```

Syntenic Correlation to Known Instance Threshold Required For Gene Cluster Presence: 0.6
Maximum distance in between candidate gene-cluster segments to perform merging: 5
Base pair for flanking context to extract: 1000
Emission probability of gene being in gene-cluster state with homologous hit to gene-cluster: 0.95
Emission probability of gene being in background state with homologous hit to gene-cluster: 0.2
Probability for gene-cluster to gene-cluster transition in HMM: 0.9
Probability for background to background transition in HMM: 0.9
Perform plotting?: True
DIAMOND Sensitivity: very-sensitive
Delineation Mode: GENE-CLUMPER
Number of CPUs Requested: 20

Parameters used for zol analysis:

Input directory with Loci GenBanks:
/home/salamzade/zol_development/showcase_examples/Phage_in_Lake_MGs/fai_Results/Homologous_GenBanks_Directory/
Output directory:
/home/salamzade/zol_development/showcase_examples/Phage_in_Lake_MGs/zol_Results/
Use super5 Mode in MUSCLE Alignments?: False
Run FUBAR Selection Analyses?: False
Skip GARD Partitioning by Recombination Breakpoints?: False
Focal GenBanks Listing: None
Comparator GenBanks Listing: None
Filter Low Quality?: False
Filter Draft/Incomplete?: False
Perform Broad Level Estimation of Homolog Group Conservation if Dereplication Requested?: False
Comprehensive Reporting of Evolutionary Statistics, Including for Non-Single Copy Homolog Groups: False
Rename Locus Tags?: False
Use CDS features with attribute near_scaffold_edge=True.: False
Perform Dereplication?: False
Perform Reinflation?: False
Dereplication Identity Threshold: 99.0
Dereplication Coverage Threshold: 95.0
Dereplication Clustering Method / MCL Inflation: None
Custom Annotation Database: None
Refine Gene Calling using the Custom Annotation Database: False
Plot Height: 7
Plot Width: 14
Use Full GenBank Labels?: False
Number of CPUs Requested: 20

Parameters used for “Microevolutionary investigations of leporin and aflatoxin BGCs in *Aspergillus flavus*”

Parameters used for fai analysis of leporin BGC:

Input directory with Gene-Cluster GenBanks: leporinB_BGC/
Reference Genome for Gene Cluster: None
Reference Scaffold for Gene Cluster: None
Reference Start Coordinate of Gene Cluster: None
Reference End Coordinate of Gene Cluster: None
Protein Queries: None
Target Genomes Prepared Directory by prepTG: ../prepTG_database/
Output Directory:
/home/salamzade/zol_development/showcase_examples/Aflavus_aflatoxin_and_leporinB/leporinB/fai_Results/
Run in Draft-Assembly Mode?: True
Filter for Paralogous/Gene-Content Overlapping Segments: True
E-value cutoff for Detection of general Protein Homologs in Genome: 1e-10

Minimum Proportion of Hits for Gene Presence in Genome: 0.5
FASTA file with Key Proteins to Consider: key_proteins.faa
E-values for Key Proteins to be Considered as Smoking Gun for Gene Cluster Presence: 1e-20
Minimum Proportion of Key Protein Hits Required for Gene Cluster Presence in Genome: 1.0
Syntenic Correlation to Known Instance Threshold Required For Gene Cluster Presence: 0.6
Maximum distance in between candidate gene-cluster segments to perform merging: 3
Base pair for flanking context to extract: 1000
Emission probability of gene being in gene-cluster state with homologous hit to gene-cluster: 0.95
Emission probability of gene being in background state with homologous hit to gene-cluster: 0.2
Probability for gene-cluster to gene-cluster transition in HMM: 0.9
Probability for background to background transition in HMM: 0.9
Perform plotting?: True
DIAMOND Sensitivity: very-sensitive
Delineation Mode: GENE-CLUMPER
Number of CPUs Requested: 40

Parameters used for zol analysis of leporin BGC:

Input directory with Loci GenBanks: /home/salamzade/zol_development/showcase_examples/Aflavus_aflatoxin_and_leporinB/leporinB/fai_Results/Homologous_GenBanks_Directory/
Output directory: /home/salamzade/zol_development/showcase_examples/Aflavus_aflatoxin_and_leporinB/leporinB/zol_Results/
Use super5 Mode in MUSCLE Alignments?: False
Run FUBAR Selection Analyses?: False
Skip GARD Partitioning by Recombination Breakpoints?: False
Focal GenBanks Listing: PopB_GCs.txt
Comparator GenBanks Listing: None
Filter Low Quality?: False
Filter Draft/Incomplete?: True
Perform Broad Level Estimation of Homolog Group Conservation if Dereplication Requested?: False
Comprehensive Reporting of Evolutionary Statistics, Including for Non-Single Copy Homolog Groups: False
Rename Locus Tags?: False
Use CDS features with attribute near_scaffold_edge=True.: False
Perform Dereplication?: False
Perform Reinflation?: False
Dereplication Identity Threshold: 99.0
Dereplication Coverage Threshold: 95.0
Dereplication Clustering Method / MCL Inflation: None
Custom Annotation Database: leporinB_proteins.faa
Refine Gene Calling using the Custom Annotation Database: False
Plot Height: 7
Plot Width: 14
Use Full GenBank Labels?: False
Number of CPUs Requested: 40

Parameters used for fai analysis of aflatoxin BGC:

Input directory with Gene-Cluster GenBanks: aflatoxin_BGC/
Reference Genome for Gene Cluster: None
Reference Scaffold for Gene Cluster: None
Reference Start Coordinate of Gene Cluster: None
Reference End Coordinate of Gene Cluster: None
Protein Queries: None
Target Genomes Prepared Directory by prepTG: ../prepTG_database/
Output Directory: /home/salamzade/zol_development/showcase_examples/Aflavus_aflatoxin_and_leporinB/aflatoxin/fai_Results/
Run in Draft-Assembly Mode?: True
Filter for Paralogous/Gene-Content Overlapping Segments: True

E-value cutoff for Detection of general Protein Homologs in Genome: 1e-10
Minimum Proportion of Hits for Gene Presence in Genome: 0.5
FASTA file with Key Proteins to Consider: key_proteins.faa
E-values for Key Proteins to be Considered as Smoking Gun for Gene Cluster Presence: 1e-20
Minimum Proportion of Key Protein Hits Required for Gene Cluster Presence in Genome: 1.0
Syntenic Correlation to Known Instance Threshold Required For Gene Cluster Presence: 0.6
Maximum distance in between candidate gene-cluster segments to perform merging: 3
Base pair for flanking context to extract: 1000
Emission probability of gene being in gene-cluster state with homologous hit to gene-cluster: 0.95
Emission probability of gene being in background state with homologous hit to gene-cluster: 0.2
Probability for gene-cluster to gene-cluster transition in HMM: 0.9
Probability for background to background transition in HMM: 0.9
Perform plotting?: True
DIAMOND Sensitivity: very-sensitive
Delineation Mode: GENE-CLUMPER
Number of CPUs Requested: 20

Parameters used for zol analysis of aflatoxin BGC:

Input directory with Loci GenBanks: /home/salamzade/zol_development/showcase_examples/Aflavus_aflatoxin_and_leporinB/aflatoxin/fai_Results/Homologous_GenBanks_Directory/
Output directory: /home/salamzade/zol_development/showcase_examples/Aflavus_aflatoxin_and_leporinB/aflatoxin/zol_Results/
Use super5 Mode in MUSCLE Alignments?: False
Run FUBAR Selection Analyses?: False
Skip GARD Partitioning by Recombination Breakpoints?: False
Focal GenBanks Listing: None
Comparator GenBanks Listing: None
Filter Low Quality?: False
Filter Draft/Incomplete?: True
Perform Broad Level Estimation of Homolog Group Conservation if Dereplication Requested?: False
Comprehensive Reporting of Evolutionary Statistics, Including for Non-Single Copy Homolog Groups: False
Rename Locus Tags?: False
Use CDS features with attribute near_scaffold_edge=True.: False
Perform Dereplication?: False
Perform Reinflation?: False
Dereplication Identity Threshold: 99.0
Dereplication Coverage Threshold: 95.0
Dereplication Clustering Method / MCL Inflation: None
Custom Annotation Database: aflatoxin_proteins.faa
Refine Gene Calling using the Custom Annotation Database: False
Plot Height: 7
Plot Width: 14
Use Full GenBank Labels?: False
Number of CPUs Requested: 20

Parameters used for “Evolutionary investigations of the *epa* locus across *Enterococcus*”

Parameters used for fai analysis:

Input directory with Gene-Cluster GenBanks: None
Reference Genome for Gene Cluster: GCF_000007785.1_ASM778v1_genomic.fna
Reference Scaffold for Gene Cluster: NC_004668.1
Reference Start Coordinate of Gene Cluster: 2071671
Reference End Coordinate of Gene Cluster: 2115174
Protein Queries: None
Target Genomes Prepared Directory by prepTG: prepTG_database/
Output Directory: /home/salamzade/zol_development/showcase_examples/Enterococcus_Epa/fai_Results/

Run in Draft-Assembly Mode?: True
Filter for Paralogous/Gene-Content Overlapping Segments: True
E-value cutoff for Detection of general Protein Homologs in Genome: 1e-10
Minimum Proportion of Hits for Gene Presence in Genome: 0.1
FASTA file with Key Proteins to Consider: key_proteins.faa
E-values for Key Proteins to be Considered as Smoking Gun for Gene Cluster Presence: 1e-20
Minimum Proportion of Key Protein Hits Required for Gene Cluster Presence in Genome: 0.5
Syntenic Correlation to Known Instance Threshold Required For Gene Cluster Presence: 0.0
Maximum distance in between candidate gene-cluster segments to perform merging: 5
Base pair for flanking context to extract: 20000
Emission probability of gene being in gene-cluster state with homologous hit to gene-cluster: 0.95
Emission probability of gene being in background state with homologous hit to gene-cluster: 0.2
Probability for gene-cluster to gene-cluster transition in HMM: 0.9
Probability for background to background transition in HMM: 0.9
Perform plotting?: True
DIAMOND Sensitivity: very-sensitive
Delineation Mode: GENE-CLUMPER
Number of CPUs Requested: 50
Maximum Memory in GB: None

Parameters used for *E. faecalis* Genome-Wide dereplication zol analysis:

Input directory with Loci GenBanks:
/home/salamzade/zol_development/showcase_examples/Enterococcus_Epa/dRep_Efaecalis_GCs/
Output directory:
/home/salamzade/zol_development/showcase_examples/Enterococcus_Epa/Benchmarking_Dereplication/GenomeWide_Derep_zol/
Use super5 Mode in MUSCLE Alignments?: True
Run FUBAR Selection Analyses?: True
Skip GARD Partitioning by Recombination Breakpoints?: True
Focal GenBanks Listing: None
Comparator GenBanks Listing: None
Filter Low Quality?: True
Filter Draft/Incomplete?: True
Perform Broad Level Estimation of Homolog Group Conservation if Dereplication Requested?: False
Comprehensive Reporting of Evolutionary Statistics, Including for Non-Single Copy Homolog Groups: True
Rename Locus Tags?: False
Use CDS features with attribute near_scaffold_edge=True.: False
Perform Dereplication?: False
Perform Reinflation?: False
Dereplication Identity Threshold: 99.0
Dereplication Coverage Threshold: 95.0
Dereplication Clustering Method / MCL Inflation: None
Custom Annotation Database: epa_prots_for_custom_annotation.faa
Refine Gene Calling using the Custom Annotation Database: False
Plot Height: 7
Plot Width: 14
Use Full GenBank Labels?: False
Number of CPUs Requested: 20
Maximum Memory in GB: None

Parameters used for *E. faecalis* & *E. faecium* (focal) zol analysis:

Input directory with Loci GenBanks:
/home/salamzade/zol_development/showcase_examples/Enterococcus_Epa/Efaecalis_and_Efaecium_GCs/
Output directory:
/home/salamzade/zol_development/showcase_examples/Enterococcus_Epa/zol_Efaecium_and_Efaecalis_reinflated/
Use super5 Mode in MUSCLE Alignments?: True
Run FUBAR Selection Analyses?: False
Skip GARD Partitioning by Recombination Breakpoints?: False

```
Focal GenBanks Listing: Listings/Efaecium_GCs.txt
Comparator GenBanks Listing: Listings/Efaecalis_GCs.txt
Filter Low Quality?: True
Filter Draft/Incomplete?: True
Perform Broad Level Estimation of Homolog Group Conservation if Dereplication Requested?: False
Comprehensive Reporting of Evolutionary Statistics, Including for Non-Single Copy Homolog Groups:
True
Rename Locus Tags?: False
Use CDS features with attribute near_scaffold_edge=True.: False
Perform Dereplication?: True
Perform Reinflation?: True
Dereplication Identity Threshold: 99.0
Dereplication Coverage Threshold: 99.0
Dereplication Clustering Method / MCL Inflation: None
Custom Annotation Database: epa_prots_for_custom_annotation.faa
Refine Gene Calling using the Custom Annotation Database: False
Plot Height: 7
Plot Width: 14
Use Full GenBank Labels?: False
Number of CPUs Requested: 50
Maximum Memory in GB: None
```

Supplementary text references

1. Larralde, M. Pyrodigal: Python bindings and interface to Prodigal, an efficient method for gene prediction in prokaryotes. *J. Open Source Softw.* **7**, 4296 (2022).
2. Hyatt, D. *et al.* Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics* **11**, 119 (2010).
3. Li, H. Protein-to-genome alignment with miniprot. *Bioinformatics* **39**, (2023).
4. Buchfink, B., Xie, C. & Huson, D. H. Fast and sensitive protein alignment using DIAMOND. *Nat. Methods* **12**, 59–60 (2014).
5. Salamzade, R. *et al.* Evolutionary investigations of the biosynthetic diversity in the skin microbiome using lsaBGC. *Microb Genom* **9**, (2023).
6. Shaw, J. & Yu, Y. W. Fast and robust metagenomic sequence comparison through sparse chaining with skani. *bioRxiv* 2023.01.18.524587 (2023) doi:10.1101/2023.01.18.524587.
7. Remm, M., Storm, C. E. & Sonnhammer, E. L. Automatic clustering of orthologs and in-paralogs from pairwise species comparisons. *J. Mol. Biol.* **314**, 1041–1052 (2001).
8. Li, L., Stoeckert, C. J., Jr & Roos, D. S. OrthoMCL: identification of ortholog groups for eukaryotic genomes. *Genome Res.* **13**, 2178–2189 (2003).

9. Huang, Y., Niu, B., Gao, Y., Fu, L. & Li, W. CD-HIT Suite: a web server for clustering and comparing biological sequences. *Bioinformatics* **26**, 680–682 (2010).
10. Edgar, R. C. Muscle5: High-accuracy alignment ensembles enable unbiased assessments of sequence homology and phylogeny. *Nat. Commun.* **13**, 6968 (2022).
11. Suyama, M., Torrents, D. & Bork, P. PAL2NAL: robust conversion of protein sequence alignments into the corresponding codon alignments. *Nucleic Acids Res.* **34**, W609-12 (2006).
12. Capella-Gutiérrez, S., Silla-Martínez, J. M. & Gabaldón, T. trimAl: a tool for automated alignment trimming in large-scale phylogenetic analyses. *Bioinformatics* **25**, 1972–1973 (2009).
13. Price, M. N., Dehal, P. S. & Arkin, A. P. FastTree 2--approximately maximum-likelihood trees for large alignments. *PLoS One* **5**, e9490 (2010).
14. Eddy, S. R. Accelerated Profile HMM Searches. *PLoS Comput. Biol.* **7**, e1002195 (2011).
15. Aramaki, T. *et al.* KofamKOALA: KEGG Ortholog assignment based on profile HMM and adaptive score threshold. *Bioinformatics* **36**, 2251–2252 (2019).
16. Li, W. *et al.* RefSeq: expanding the Prokaryotic Genome Annotation Pipeline reach with protein family model curation. *Nucleic Acids Res.* **49**, D1020–D1028 (2021).
17. Liu, B., Zheng, D., Jin, Q., Chen, L. & Yang, J. VFDB 2019: a comparative pathogenomic platform with an interactive web interface. *Nucleic Acids Res.* **47**, D687–D692 (2019).
18. Alcock, B. P. *et al.* CARD 2023: expanded curation, support for machine learning, and resistome prediction at the Comprehensive Antibiotic Resistance Database. *Nucleic Acids Res.* **51**, D690–D699 (2023).
19. Terlouw, B. R. *et al.* MIBiG 3.0: a community-driven effort to annotate experimentally validated biosynthetic gene clusters. *Nucleic Acids Res.* **51**, D603–D610 (2023).
20. Siguier, P., Perochon, J., Lestrade, L., Mahillon, J. & Chandler, M. ISfinder: the reference centre for bacterial insertion sequences. *Nucleic Acids Res.* **34**, D32-6 (2006).

21. Finn, R. D. *et al.* Pfam: the protein families database. *Nucleic Acids Res.* **42**, D222-30 (2014).
22. Tajima, F. Statistical method for testing the neutral mutation hypothesis by DNA polymorphism. *Genetics* **123**, 585–595 (1989).
23. Murrell, B. *et al.* FUBAR: a fast, unconstrained bayesian approximation for inferring selection. *Mol. Biol. Evol.* **30**, 1196–1205 (2013).
24. Kosakovsky Pond, S. L., Posada, D., Gravenor, M. B., Woelk, C. H. & Frost, S. D. W. GARD: a genetic algorithm for recombination detection. *Bioinformatics* **22**, 3096–3098 (2006).
25. Kosakovsky Pond, S. L. *et al.* HyPhy 2.5—A Customizable Platform for Evolutionary Hypothesis Testing Using Phylogenies. *Mol. Biol. Evol.* **37**, 295–299 (2019).
26. Hudson, R. R., Slatkin, M. & Maddison, W. P. Estimation of levels of gene flow from DNA sequence data. *Genetics* **132**, 583–589 (1992).
27. Gilchrist, C. L. M. *et al.* Cblaster: A remote search tool for rapid identification and visualization of homologous gene clusters. *Bioinformatics Advances* **1**, (2021).
28. Medema, M. H., Cimermancic, P., Sali, A., Takano, E. & Fischbach, M. A. A systematic computational analysis of biosynthetic gene cluster evolution: lessons for engineering biosynthesis. *PLoS Comput. Biol.* **10**, e1004016 (2014).
29. Schreiber, J. Pomegranate: fast and flexible probabilistic modeling in python. *J. Mach. Learn. Res.* (2017).
30. van den Belt, M. *et al.* CAGECAT: The CompArative GEne Cluster Analysis Toolbox for rapid search and visualisation of homologous gene clusters. *BMC Bioinformatics* **24**, 181 (2023).