RiPP precursor prediction using machine learning on conservation patterns

Maarten Boneschansker
Rijksuniversiteit Groningen

under external supervision of dr. M.H. Medema
Department of Plant Sciences
Bioinformatics
Wageningen University and Research Center

internal supervision of prof. dr. G.S. van Doorn
Faculty of Science and Engineering
Groningen Institute for Evolutionary Life Sciences
Rijksuniversiteit Groningen

A thesis presented in partial fulfillment of the degree of Msc. Biology

Abstract

Genome mining holds great promise for a new 'Golden Age' in natural product discovery. Given the developments in bioinformatics and the current torrent of genome data, the future looks bright for the field. It is also necessary as a decrease in discovery of new compounds, notably also because of a lack of interest in natural discovery, coincides with a steep rise in antimicrobial resistance - with as many as 50.000.000 to die a year worldwide in 2050. Many classes of natural products can be mined from genomes directly, one class specifically well-suited for this are ribosomally and post translationally modified peptides (RiPPs), and new tools based on machine learning methods have shown their value predicting RiPPs. One such tool is decRiPPter, which predicts RiPP biosynthetic gene clusters using an SVM classifier. However, as an exploratory tool decRiPPter prizes novelty over accuracy and a large amount of false positives is thus expected. RiPP precursor peptides have a unique leader-core structure which has been shown to be differentially conserved. Presented here is a bioinformatic pipeline that predicts RiPPs in genomic data using a random forest model trained on conservation patterns. The presented model achieves a high accuracy, most notably with a very low false positive rate in held-out validation. Cross-validation experiments show that the model is also able to distinguish negative from positive training data well. Predictions on putative RiPP precursors as predicted by decRiPPter are largely negative though, which raises some concern about the predictive value of the model as decRiPPter data has already been experimentally verified. Even though the current model might no be practically useful yet as a RiPP prediction tool, work presented here proves that RiPPs can in fact be detected based on conservation patterns, which opens up the way to yet another paradigm in RiPP detection.

My time at here in Wageningen has been great and I would like to personally thank Marnix Medema for not only supervising me in a friendly and keen way, but also for inspiring me to pursue a career in natural product discovery. I also express my gratitude to Dr. Ron Wehrens of Wageningen University who has been of great help with programming in R. Finally I would like to thank my peers at the bioinformatics group, who have been of much help to me and have pitched sometimes crucial ideas. I have felt welcome from the beginning.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Natural Products

Natural products can be defined as chemicals produced by nature and humanity has benefited from their use greatly for thousands of years. Natural products come in all shapes and sizes: from simple molecules like alcohol and carbon dioxide, used for preservation or other purposes, to increasingly complex molecules like β-lactam antibiotics or Taxol - an anti-cancer agent. Common classes include terpenes, polyketides, non-ribosomally synthesized peptides, ribosomally synthesized peptides, alkaloids, glycosides, phospohonates, and phenylpropanoids. Most natural products are produced as secondary metabolites. That is, often not crucial to an organisms direct survival, but a benefit to overall fitness nevertheless. They can serve as venoms, toxins, scents, pigments, hormones, quorum sensors, in a sense anything that is not directly involved in homeostasis and, crucially, thus usually directed at external targets.[1] Sometimes natural products are used natively, for example the antibiotic penicillin derives from the penicillin fungus which uses it as such. Sometimes as something completely different, for example caffeine and nicotine, both originally plant insecticides.

Natural products are an important source of bioactive compounds in many fields, but one of their most distinct contributions is in antibiotics - one of the pillars of modern medicine. [2] In fact, almost a 100 years after Alexander Fleming discovered the first antibiotic in 1928 - penicillin - a large proportion of antibiotics in use today are still natural products or derivatives thereof, of which two-thirds derive from the bacterial phylum *Actinobacteria* alone. These soil-dwelling bacteria are common and quite recognizable as they produce geosmin, responsible for the typical smell of wet soil after rain.[3, 4] Unsurprisingly as cutting-edge antibiotics can thus be found in the average backyard, pharmaceutical companies once encouraged their personnel to take home soil-samples from holidays for analysis. A famous example includes Avermectin, an antibiotic still in wide use, which was discovered by Nobel laureate professor Satoshi Ōmura from analysis of a soil sample he took on his local golf course. This straightforward method of collecting environmental samples, cultivating, and screening for cell growth inhibition or death resulted in more than a 1000 new natural products in the decades after WWII. [5]

1.2 Discovery

The rate of discovery of natural products has however slowed down significantly in recent decades, critically also in antibiotics. While during "The Golden Age" of drug discovery, during the 1940s to 1970s, dozens of new antimicrobials were introduced, saving countless lives, the subsequent 20-year period produced only one truly novel antibiotic: daptomycin. Furthermore, those that were discovered were far more likely to be similar to known compounds than in previous eras.[6, 7] Most newly approved antibiotics were 2nd to 4th derivatives of known classes, which are more susceptible to resistance as they are more chemically similar. Because of disappointing results from natural product research, the 1990s then saw the rise of combinatorial chemistry, where millions of compounds were generated based on iterations of variation on a base compound and screened for biological activity, mostly to the exclusion of natural product research. Unfortunately, this brute force strategy was a dead end with no viable drug reported at all.[8, 5]

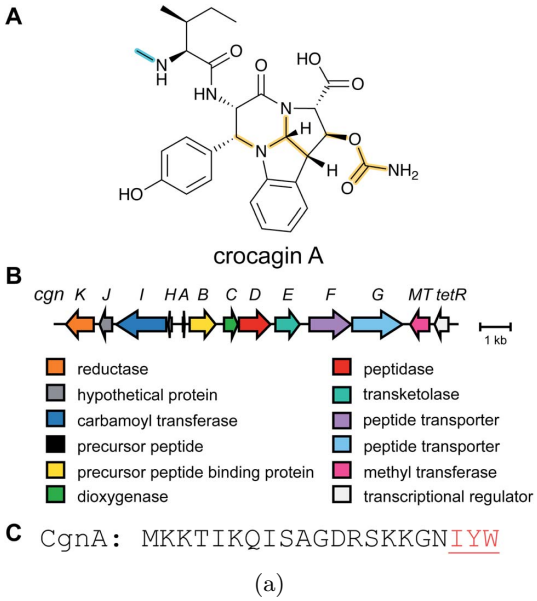
As the low-hanging fruit in natural product research seems to have been picked, another approach is thus required and here *in silico* methods like genome mining hold great promise. The advent of next generation sequencing has allowed researchers to sequence the (meta)genomes of many microorganisms and deposit them in publicly available databases. This opened up vast genomic landscapes of biosynthetic potential not otherwise accessible to classical fermentation-based approaches.[9, 1, 5] As even though 99% of bacteria can not be cultured - dubbed 'the great plate anomaly' - or do not express a certain gene (*silent/cryptic*), they still harbour their biosynthetic potential in the genome.[10, 11, 12]

Combined with advances in bioinformatics the recent and ongoing avalanche of genome sequences holds great promise for a new age of fruitful natural product discovery and engineering.[13] Indeed, uncultured soil bacteria have already been proven to be a reservoir of antibiotic resistance genes, which going by 'where there is smoke, there is fire' suggests presence of a reservoir of antibiotic genes.[14] Even mining of earlier mentioned well-studied *Actinobacteria* genomes revealed huge amounts of potential biosynthetic gene clusters, which suggests that our current knowledge of natural products is only the 'tip of the iceberg'.[3] Estimates range, but a recent review put the number of potential natural products at 18,000,000, of which only 500,000 are currently known.[1]

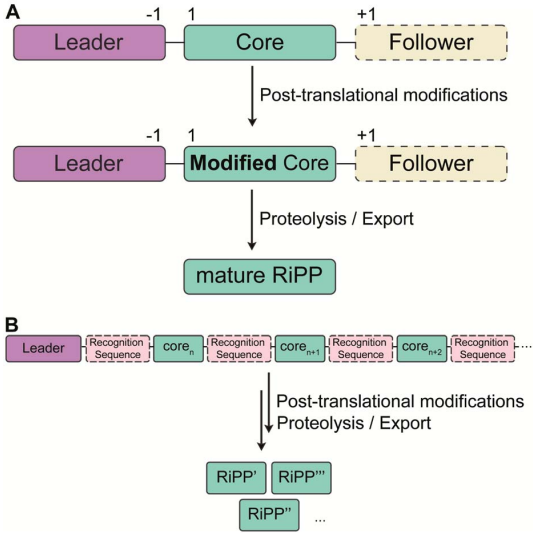
1.3 RiPPs

Knowledge on one class of natural products called **Ri**bosomaly synthesized and **P**ost-translationally modified **P**eptides (RiPPs) has expanded much thanks to genome mining. RiPPs are a large and diverse class of natural products and have mostly been found in bacteria and fungi, but also in plants and some animals. Like with many other natural products, RiPP-encoding genes tend to cluster on the genome in structures called biosynthetic gene clusters (BGCs). A key aspect of RiPPs however is the presence of a precursor within the BGC. Often the first or second gene in a RiPP BGC encodes a peptide, usually 20–110 residues (but even down to 5[15] or 7 [16] and up to 293 residues [17]), that serves as a precursor to maturing enzymes (maturases) that are found further down the BGC. Thus, a RiPP BGC uniquely contains both precursor and modifying machinery. This genomic organization is (among other theories) theorized to allow for synchronized transcription of precursor and modifying enzymes and precursor-maturase specificity and efficiency. [18, 19]

RiPP precursors can be divided into an N-terminal leader, and C-terminal core, sometimes followed by another peptide dubbed a follower, and, in eukaryotes, a signal peptide N-terminal to the leader. After transcription the leader peptide plays a role in correct and efficient maturation of the core region, which involves cleaving of the leader peptide from the core region.[20, 18, 21, 19] Shown in figure 1 are the typical organization of a RiPP BGC and RiPP synthesis. Usually the core sequence is less than 10 residues, with the rest of the precursor serving as leader, follower or recognition sequences. A notable exception to the usual organization are cyanobactins, which can be organized in multiple, often similar, core sequences per BGC accompanied by recognition sequences, preceded by a single leader.



(a)



(b)

Figure 1: (a): RiPP example crogacin A (A), BGC (B), and precursor with core in red (C). (b): General RiPP biosynthesis (A), with multiple cores as in cyanobactins (B).

Whereas the sequence of the leader peptide can have great effect on maturation of the final product, maturase enzymes seem more promiscuous to core peptide sequence.[22] Leader-core regions have even been shown to be interchangeable between classes of RiPPs.[23] Indeed, maturase enzymes often contain domains called RiPP Recognition Elements (RREs) that allow enzymes to recognize leader peptides, but not core peptides. [24, 25]

In accordance with respective maturase promiscuity, leader peptides are generally conserved, whereas core peptides are more variable, sometimes even hyper-variable. This leader-core variability is thought to endow an organism with a lot of biosynthetic potential from large combinatorial libraries at a relatively low genetic cost. Theoretically, a single nucleotide or codon change could change the resulting RiPP greatly, whereas a significant change in other classes of NP's usually requires modification of an entire enzyme domain. RiPPs are in a sense both very sensitive and insensitive to mutations; sensitive

in that small leader mutations can have large effects, insensitive in that core mutations still result in a natural product with only a slight modification.[26]

RiPPs are usually classified based on common biosynthetic machinery and/or shared motifs[18]. However, as novel RiPPs are being discovered that are hybrids between classes or of completely novel class carrying previously unknown modifications, these classifications are subject to change. The latest comprehensive review identified roughly 20 RiPP classes, but it is thought that there is still a very large reservoir of yet undiscovered RiPP classes.[19] More recent (meta)genome studies sampling from various environments do indeed report novel RiPP classes, novel modification machinery, and even entirely new species with biosynthetic potential.[27, 28]

1.4 decRiPPter

Many bioinformatic tools have been developed to detect RiPPs.[18, 19] Even whole ready-to-use analysis shells are available, such as antiSMASH.[29]However, most of these tools are based on known 'core' modifying enzymes for a given RiPP (sub)class. This makes them sensitive to specific (sub)classes with well established characteristics such as lanthipeptides, but insensitive to novelty and classes which are not as easily defined.

To address this, tools have been developed that are not class-based, but rather use overarching characteristics true of RiPPs to detect novelty. One such tool is decRiPPter[30], which uses a support vector machine (SVM) based on 36 physio-chemical features to recognize RiPP precursors in predicted open reading frames (ORFs), and subsequently pan-genomic analyses based on the knowledge that RiPPs are usually secondary metabolites and therefore not part of the set of shared genes within most species in a genus - the core genome. However, because decRiPPter is meant as an exploratory tool, novelty is prized at the cost of accuracy.

As of late, an unpublished faster version of decRiPPter has been used by Nico Louwen of Wageningen University to mine a large set of genomes related to the microbiome, resulting in 91,424 putative RiPP BGCs grouped into 1,132 distinct Gene Cluster Families (GCFs). A considerable segment of these 91,424 BGCs are estimated to be false positives however and therefore there is a need to identify the most promising candidates, which can then be experimentally validated. It is important to note that predicting the product of a BGC exactly remains challenging [27], let alone predict bioactivity. Therefore experimental validation remains vital, which requires orders of tens, not ten-thousands, candidate BGCs. Aside from decRiPPter's internal filtering methods, N. Louwen used the following requirements to decrease the number of false positives: only BGC's with no antiSMASH overlap to ensure novelty; presence of at least two known biosynthetic enzymes; presence of at least one peptidase, transporter, and regulator; and an average COG score (as generated by decRiPPter) below 0.1. Application of these filters reduced the number of clusters from 91.424 to 4.290, which is still a lot and we still expect large numbers of false positives to be present. N. Louwen suggests two options to further decrease false positive rates. First, mapping metagenome and transcriptome from microbiome data can help validate RiPP clusters, additionally also in healthy vs disease phenotypes. This was done by S. Quiroga et al. of Princeton University[31]. Second, to explore the conservation patterns of precursors in RiPP BGC's to further prioritize candidate clusters.

This thesis describes a method using machine learning (ML) on peptide conservation patterns to increase the accuracy of decRiPPter without compromising novelty. Based on knowledge of differential precursor leader-core conservation from literature and RiPP BGSs being part of a secondary genome and not the core genome, it is hypothesized that RiPP precursors are conserved according to RiPP-specific patterns. Here a Self Organizing Map (SOM) and a random forest (RF) model, trained on conservation patterns of RiPP-precursor peptides and non-RiPP peptides, are used to calculate RiPP-prediction scores on putative RiPP precursors generated by decRiPPter.

2 Methods

A summary of methods is shown in figure 2. The first being *enrichment* on the left side of the flowchart. Sequences from N. Louwen's decRiPPter results and data from the training data from A. Kloosterman et al. were downloaded and merged with homologs from a custom NR database. This was done to ensure enough sequences for the next step: *RiPP-Prediction-score* generation, where conservation scores were calculated for each enriched sequence/cluster. Conservation scores were calculated per tenth segment of each cluster to asses any difference between core and leader conservation patterns. Resulting conservation scores where then used to construct a random forest model and a Self-organizing map. Finally, only the RF model was used to predict RiPP-prediction-scores on putative precursors.

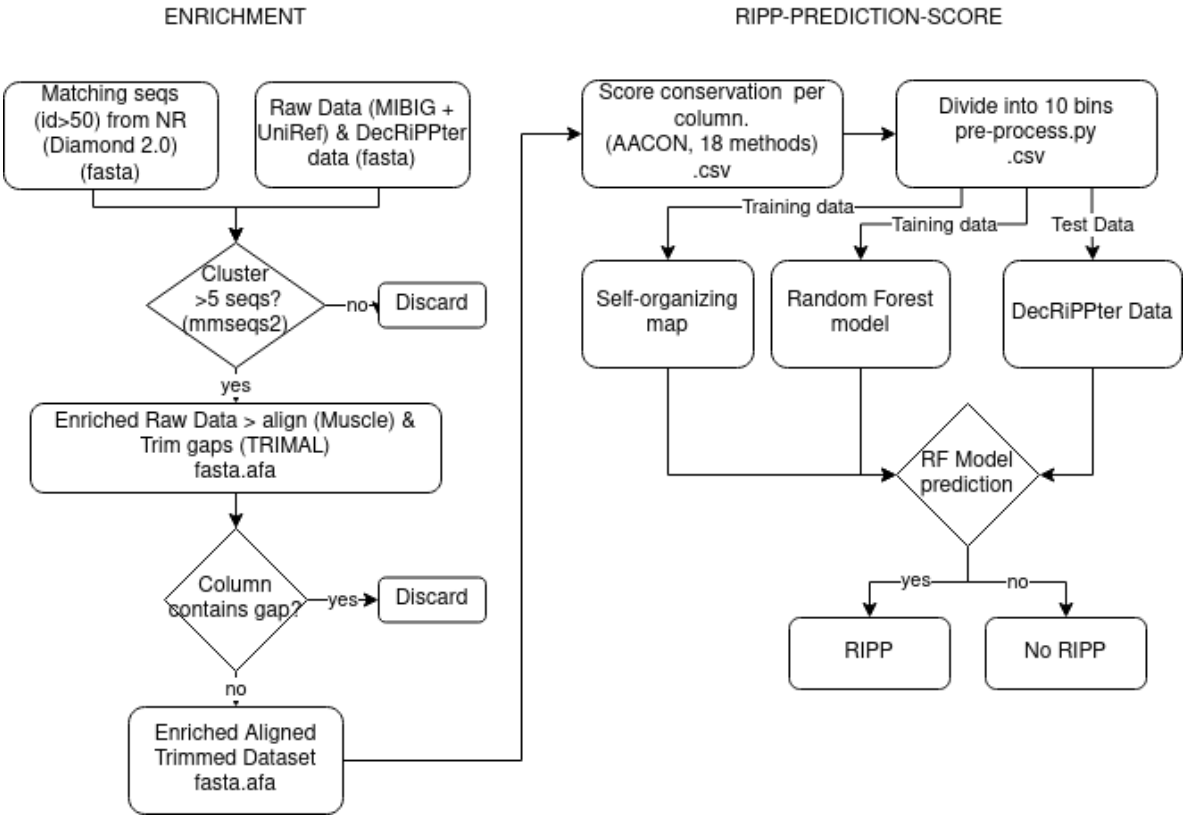


Figure 2: Schematic overview of methods. Tools used and data file type/extensions are shown.

2.1 Data

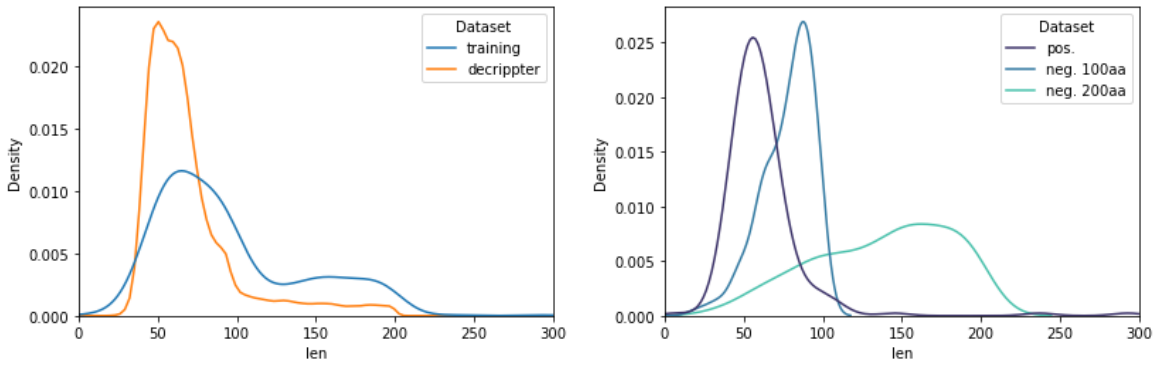
2.1.1 Data acquisition

Both a negative and positive dataset were constructed for use in ML algorithms. For the positive dataset known RiPP precursors were the same as used by Kloosterman et al.[30] to train the SVM of decRiPPter, data is available online (https://zenodo.org/record/3834818). In total 195 precursors were used, all of which are present in the MIBiG 2.0[32] repository or recently reported to be RiPPs. These precursors ranged in length from 7 (microcin C7) to 293 (megacin) residues, but with most precursors (184/195) between 40 and 110 residues long. Precursors spanned 10 RiPP classes and were derived from multiple genera.

A negative dataset was constructed by sampling from the negative dataset used for training in decRiPPter, also as described by Kloosterman et al. The original dataset comprised two parts: 10,000 short proteins (<175 amino acids) from Uniprot (2014 query) [33] and a set of 10,000 predicted proteins. The negative dataset used here was constructed using only the Uniprot data from decRiPPter. The negative dataset also comprises two parts. First a dataset of 175 sequences with a maximum length of a 100 amino acids was randomly sampled from Uniprot used by Kloosterman et al., then a dataset

of 200 sequences with no length requirement was randomly sampled from the same Uniprot dataset. This resulted in two similar datasets, but with different lengths.

Data from Nico Louwen's experiment as described in his thesis was obtained from Wageningen University servers. [34] Precursors were predicted by decRiPPter per genus and this separation was kept, thus for every genus of Nico Louwen's decRiPPter results, a FASTA file containing all putative precursors was obtained. Data obtained from Nico Louwen is referred to as *decRiPPter dataset*. Positive and negative datasets are referred to as *training dataset*. Length distributions of datasets are shown in figure 3.



(a) *training dataset* vs *decRiPPter dataset*

(b) breakdown of *training dataset*

Figure 3: Kernel Density Estimate distribution of the length of sequences in enriched datasets. (a) the *training dataset* is skewed towards longer sequences compared to *decRiPPter dataset* (b) this is due to the fact that half of the negative data, and thus roughly a third of training data, has a length distribution that is more skewed to lengths of more than a 100 residues.

2.1.2 Enrichment

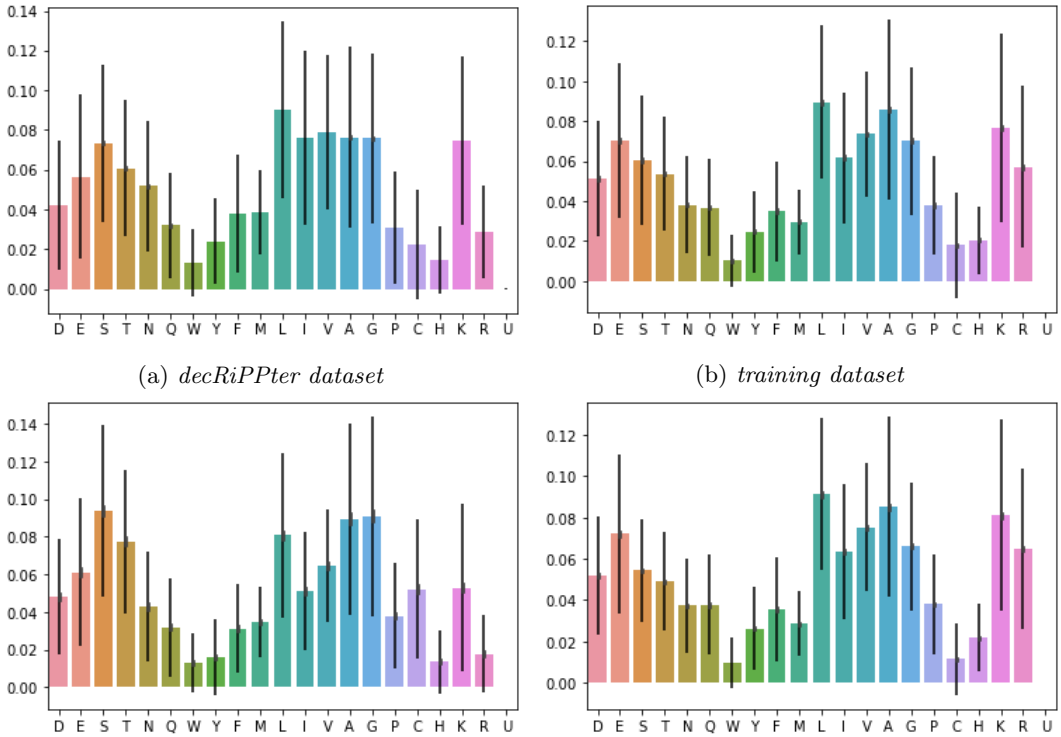
Most sequences from both the *training* and *decRiPPter dataset* could not be clustered into multiple sequence files containing an appropriate number of sequences for meaningful conservation scores (≥ 5). In order to allow for more meaningful conservation scores data was enriched with homologs from other genomes. Ideally RiPP precursors only would be used but there is not enough data available yet to do this, thus data was enriched with homologs which were mostly non-RiPPs. This is justified as we only look at RiPP precursor conservation patterns and not characteristics of the enriched data itself. The enrichment method consisted of two steps and was performed on the *training* and *decRiPPter datasets*.

First a sequence similarity search was ran using using DIAMOND 2.0 [35] vs a custom NCBI RefSeq non-redundant proteins (NR) database. This custom NCBI NR database was constructed by filtering for sequences that were shorter than 200 amino acids. The main purpose of using this custom database was to increase the speed of the DIAMOND 2.0 search and since only one (megacin) RiPP sequence exceeded 200 amino acids this was deemed safe. Second, resulting sequences with hits with ≥ 50 sequence identity similarity [1] to a query sequence were downloaded via the command line Efetch (v16.2, NCBI) tool and added to the original sequence file, resulting in a new 'enriched' multiple sequence file for each sequence. Not all sequences could be enriched properly and enriched multiple sequence files containing fewer than 5 sequences were discarded.

In total, for the *training dataset*, from the negative datasets 172/175 and 192/200 sequences were successfully enriched. The positive *training dataset* was successfully enriched for 130/175 sequences. The *decRiPPter dataset* was successfully enriched for 54,198/130,903 sequences. Data from the *decRiPPter dataset* was clustered into 11,636 clusters. No clustering was applied to the *training dataset* to ensure enough data for training. Because of this, some redundancy is present in the *training dataset* as sequences that could have clustered together are kept separated and sometimes have overlap in enrichment sequences. See table 1 for a more detailed description.

[1] it is important to note that sequence similarity is only a proxy to true homology[36]

Enriched datasets were then assessed for amino acid distribution, shown in figure 4. A slight difference in composition between the positive and negative subsets of the *training dataset* is present, with the positive dataset being relatively enriched for Serine (S) and relatively impoverished for lysine (K) and arginine (R). Interestingly, decRiPPter data is also relatively arginine poor and serine rich. Amino acid composition varies highly though, this is reflected in the high standard definitions. Generally speaking amino acid composition of datasets is similar.



(a) *decRiPPter dataset* (b) *training dataset*

(c) *positive training dataset* (d) *negative training dataset*

Figure 4: Ratios of amino acid composition average and standard deviation of *training* and *decRiPPter dataset*. Amino acids are ordered left to right from more electronegative to more electropositive. Rare amino acid selenocysteine (U) is listed as it occurred several times in in the *decRiPPter dataset*.

Dataset	raw	orig.	mean	stdev.	enrich	clust.	/clust	stdev.
train	550	495	21.49	7.31	11145	495	22.52	4.94
pos	175	130	16.05	9.45	2444	130	18.80	6.93
Neg_100	175	172	24.14	3.85	4084	172	23.74	3.14
Neg_200	200	193	23.82	4.45	4617	193	23.92	3.02
decRiPPter	130903	54198	6.87	4.01	170587	11636	14.66*	22.40
Genus	raw	orig	mean	stdev.	enrich	clust.	/clust	stdev.
Akkermansia	104	27	5.14	3.98	58	7	8.29	4.07
Alistipes	456	80	5.17	3.89	371	39	9.51	3.93
Anaerostipes	204	49	5.78	3.74	439	40	10.98	5.01
Bacillus	23260	10009	7.19	3.68	32569	1595	20.42	34.37
Bacteroides	3582	1916	5.43	3.72	3613	273	13.23	14.67
Bifidobacterium	6254	2542	4.42	3.76	3577	304	11.77	11.98
Blautia	4281	1893	5.97	3.78	4343	364	11.93	11.35
Citrobacter	1523	762	7.9	3.41	2422	163	14.86	18.19
Clostridioides	4678	1184	6.11	3.91	7007	567	12.36	14.38
Clostridium	16738	6367	6.31	3.86	21862	1252	17.46	29.58
Collinsella	689	130	6.18	4	436	37	11.78	5.81
Coprococcus	783	292	5.98	3.69	1152	113	10.19	6.85
Corynebacterium	6808	3125	5.43	3.86	4773	385	12.40	15.13
Desulfovibrio	1380	130	5.33	4.05	782	62	12.61	9.62
Enterobacter	3410	1455	6.5	3.78	5168	367	14.08	16.9
Enterococcus	3788	1808	7.26	3.74	5704	409	13.95	16.92
Escherichia	6552	2848	6.92	3.7	11271	800	14.09	16.55
Eubacterium	917	204	7.57	3.61	1274	108	11.80	7.03
Exiguobacterium	265	134	5.69	3.89	353	29	12.17	7.29
Fusobacterium	291	154	5.95	3.97	827	76	10.88	7.11
Klebsiella	2373	751	7.52	3.36	2545	240	10.60	6.99
Lactobacillus	5537	1841	6.48	3.91	6445	513	12.56	15.58
Lactococcus	788	346	6.15	3.82	1272	98	12.98	13.66
Megasphaera	163	15	6.43	3.72	138	14	9.86	1.46
Olsenella	324	17	4.57	3.88	100	10	10.00	3.97
Parabacteroides	731	366	5.05	4.07	837	75	11.16	8.85
Prevotella	3297	1037	5.42	3.81	2491	246	10.13	9.25
Roseburia	1661	705	4.44	3.59	2164	210	10.30	6.87
Ruminococcus	2369	716	5.97	3.69	2861	248	11.54	8.13
Salmonella	5965	1498	6.06	3.79	3877	403	9.62	5.2
Staphylococcus	4576	1961	5.64	3.85	9439	752	12.55	11.6
Streptococcus	15710	9215	7.64	3.49	28436	1689	16.84	30.06
Vagococcus	260	83	7.97	3.25	506	31	16.32	14.94
Veillonella	192	65	6.39	3.9	253	24	10.54	5.88
Weissella	994	473	6.73	3.77	1222	93	13.14	12.49

Table 1: Enrichment for each dataset and subsequent breakdown of *decRiPPter dataset* per genus. *Raw*: number of sequences in the original, unenriched dataset. *orig.*: number of sequences out of raw data that could be successfully enriched with at least one homolog. *mean, std.*: average number and standard deviation of enrichment sequences added to each original sequence. *enrich*: number of sequences added in total during enrichment, after filtering. *clust.*: amount of clusters in enriched datasets. Note that for the *training datasets* the number of clusters is the same as the number of original sequences. *median = 10.00.

2.2 Conservation

2.2.1 Multiple sequence alignment

A multiple sequence alignment (MSA) was made for each enriched multiple sequence file using MUSCLE (v5.1) [37] on default settings. MSAs were then trimmed of all gaps using trimAL (v1.4) [38]

using no-gaps settings. This was done because earlier try-outs revealed conservation scoring to be unfairly sensitive to gaps in the alignment, drowning out any relevant sequence-based conservation signal. Resulting trimmed MSAs were visualized using Jalview 2 (v2.11.2)[39] to allow for visual inspection.

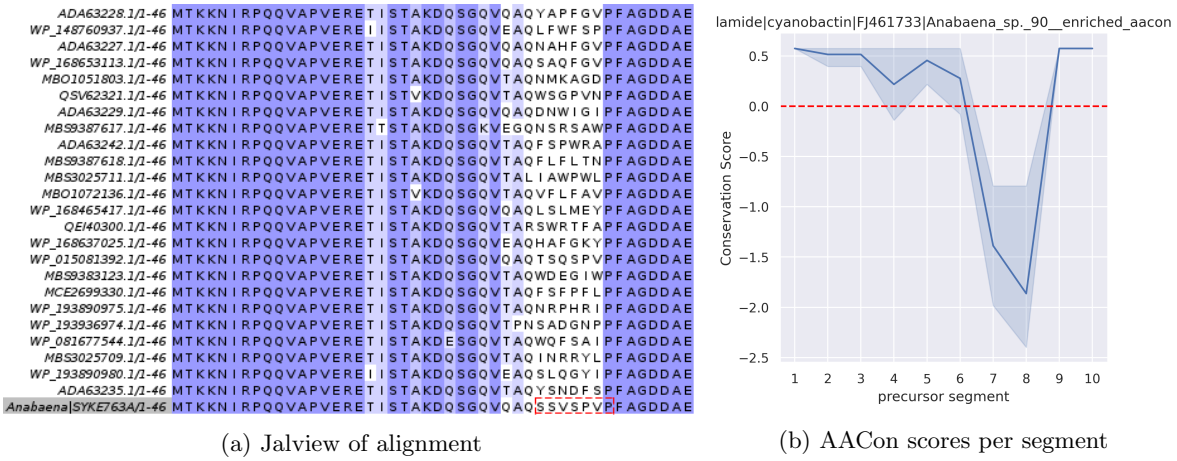
2.2.2 Conservation scoring

Trimmed MSAs were then scored for conservation per residue using standalone AACon (v1.1) [40] scoring tool with flag -n, which calculates normalized conservation scores for each residue using 18 different conservation scoring methods as reviewed by Valdar et al. [41]. As MSAs varied in length, the resulting tsv file was further 'horizontally' normalized over residue number to be able to make inter-MSA comparisons. This was done by dividing each MSA into 10 equal segments and taking the average of the corresponding AACon scores using a custom python(v.3.9.13) script (pre_process_aacon.py). Note that peptides are of different length and thus also are the segments (*training dataset* mean/stdev: 11.0/4.5, *decRiPPter dataset* mean/stdev: 6.9/3.1). Normalization was performed using the z-score method of standardization according to the expression:

$$Z = \frac{v - n}{\sigma}$$

Where: Z = z-score value, v = conservation score, n = mean, σ = standard deviation

These further normalized and segmented AACon scores per MSA were then visualized using python seaborn(v.0.11.2) package.[42] An example alignment and respective score per segment are shown in figure 5.



(a) Jalview of alignment

(b) AACon scores per segment

Figure 5: (a) an example trimmed enriched alignment. Color intensity indicates conservation by BLOSUM62 score. The core sequence (SSV-IWG-SPVP) for Anabaena SYKE763A/1 anacyclamide precursor and aligned sequences is contained in the box. Note that the core sequence is not complete; columns containing IGW were trimmed. The core sequence is hypervariable compared to leader and follower sections of the precursor peptide. (b) Mean conservation scores on all methods as calculated by AACon per segment of the precursor with confidence interval. The hypervariability of the core region is clearly reflected in the conservation scores..

2.3 Segment correlation

2.3.1 Pairplot

Correlations between segments were assessed using python seaborn package. Correlations between segments exist by definition as they are ratios, that is the sum of ratios is 1 so if one segment goes up, another has to come down. RiPP-specific patterns might be possible though, a high score on segment 0 might for example be a good predictor for a low score on segment 9. To asses this first a pairplot was constructed consisting of a matrix of scatterplots containing all *training data*. This was done for

each method and repeated for the *decRiPPter dataset*. Based on patterns in the pairplots, ARMON method was chosen as the method to continue with as it generally provided the most clear distinction between RiPPs and non-RiPPs.

2.3.2 Hierarchical clustering

A hierarchically clustered heatmap was also constructed using seaborn's built-in hierarchical clustering method on default settings. This was done for both *decRiPPter dataset* and *training dataset* to asses any possible clustering. This was done only for ARMON method conservation scores. The clustering method used was the unweighted pair group method with arithmetic mean (UPGMA) algorithm.

2.4 Machine Learning

Two machine-learning models were trained on the *training dataset* and used to predict the probability of peptides to be a RiPP precursor. Because RiPPs were found to be variable in conservation patterns and because RiPPs are known to be of great variability in sequence and structure, an unsupervised approach was initially taken.

2.4.1 Self-Organizing Map

First training data was used as input into an unsupervised Self-Organizing Map (SOM) machine learning algorithm using the Kohonen package in R. [43, 44, 45] A SOM is a dimensionality reduction artificial neural network. Based on high dimensionality data a SOM constructs a lower dimensional (usually 2D) map which fits the data best. A map consists of nodes, with each node having a certain value for each dimension, in our case each node thus has 10 values, one for the conservation of each segment. While the number of nodes in a SOM is predefined, nodes can be empty, so the amount of nodes data is grouped on is determined by optimization rather than predefined. This is a distinct difference compared to k-means clustering and is an advantage when the structure of data is unknown. Each node is similar to its neighbour, such that a datum assigned to node A will be similar to a datum assigned to neighbouring node B. Because similar data is grouped in similar topological space, a SOM offers a very natural visual way to asses patterns in data.

After construction of a SOM in training mode, new data can be mapped to a SOM as a means of classification. In classification mode a SOM classifies each node to be either a 1 or 0, in this case RiPP or non-RiPP. New data is then mapped to nodes, membership of either a 1 or 0 node indicates a positive or negative RiPP prediction. It is important to note that the node classification threshold is arbitrary and positive nodes do not contain RiPPs only and negative nodes do not only contain non-RiPPs. It can thus be said that membership of a RiPP rich node indicates a RiPP-like conservation pattern.

Accuracy of SOM classification was assessed using a cross-validation method where a random sample of training data was held out on several iterations of the algorithm. The held-out data was not used for parameter optimization. Several iterations of constructing and validating different SOMs were tried using different hyperparameters with different values for training rate (α 0.001-0.05), test/training ratio (0.5-0.9), number of learning iterations (rlen 500-300,000), and SOM grid size (range: 4x4, 5x5 ... 12x12). An optimum on test data was achieved on the following settings: $\alpha = 0.05$, rlen 1000, grid size 6x6, test/train 0.1/0.9. The SOM model with the best accuracy, based on 10 iterations with a random seed on held-out test data was selected for use in predicting scores on the *decRiPPter dataset*. SOM accuracies on training data remained low however (see Results) and another model was chosen.

2.4.2 Random Forest & Decision Tree

As the SOM model proved ineffective, a random forest (RF) model was chosen as RF-models are considered to perform better on dissimilar data. A RF classifier model from the Python scikit learn package [46] was constructed based on training data. The RF model was trained on 80% of the training data, with 20% withheld for validation. The 80/20 ratio was chosen as it provided good training accuracy while still allowing for a substantial test set. As a first step towards the RF model, a decision tree model was constructed. Then an RF model was constructed based on training data. Model accuracy increased for the RF classifier compared to the decision tree. RF models are also less sensitive to overtraining so the RF model was chosen to use for further analysis. The RF criterion for

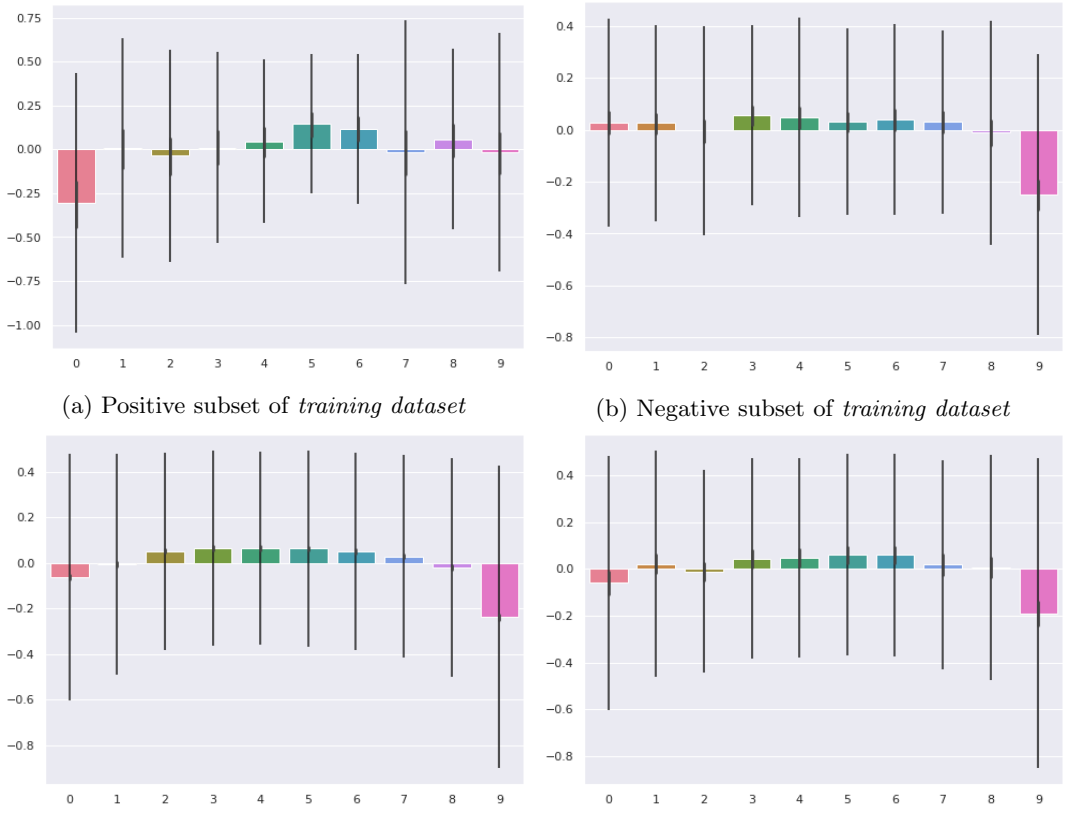
best split was 'entropy' and the RF model consisted of a 100 trees. Fine tuning of hyperparameters for the RF model was done using python *itertools* package. Based on multiple iterations on max features and max depths. The max features setting defines how many features can be used at each split/decision step of the tree and the the following options were tried: 1 feature, square root of features (sqrt(10) = 3.16), and log2 of features (log2(10) = 3.32). The max depths settings defines how many split/decision steps a tree can consist of and the following options were tried for this setting: 'None' (were depth is unlimited and the tree goes as deep as to define every individual datum) and numbers 2 to 15. An optimal on predictions was found on the following settings: max features = log2 and max depth = 11.

The final RF model was used to predict RiPPs from putative precursors from the *decRiPPter dataset*. An ROC curve was then constructed using the RocCurveDisplay module from sklearn to asses the model. The final RF model was then also used to predict scores for the positive and both negative *training dataset* as if they were new datasets, as a means of validation.

3 Results

3.1 Conservation

Surprisingly, conservations scores for *training data* and *decRiPPter data* did not follow expected patterns as can be seen in figure 6. Firstly positive data, containing RiPPs and homologs was expected to be mostly conserved in the leader region in the first segments of the precursor. Less conservation was expected more halfway as that is were the core region is expected. Previous literature had pointed to a conservation pattern of high conservation first and then less conservation or even hypervariability as that is were the core region(s) are located. In fact the first segment was the least conserved at all and the segments associated with the core region were most conserved. The negative training data also surprised. This dataset was theorized to be more uniform in conservation as even though differences in conservation are expected, this was not expected to follow a pattern as a leader-core structure is absent in regular peptides. However, data clearly shows uniform conservation for the first 9 segments and then a sharp drop in conservation score for the 10th segment. Why this is the case is unsure, but it is very interesting to note that this is directly opposite to conservation patterns found for the positive dataset. Perhaps most striking is that data from decRiPPter seems to be somewhere in between, with lower conservation at both the first and last segments, suggesting at least presence of some RiPP-like conservation patterns. More generally though, standard deviations are much higher than averages, indicating that variation within segments is much larger than variation between segments.



(a) Positive subset of *training dataset*

(b) Negative subset of *training dataset*

(c) *decRiPPter dataset*

(d) *Training dataset*

Figure 6: Mean and standard deviations of conservation scores per dataset.

3.2 Segment correlation

Pairplots were generated for each conservation scoring method, a pairplot for ARMON conservation scoring method *training data* is shown in figure 7. as it seemed to contain the most difference between RiPP's and non-RiPPs. Most plots skew to the upper right corner, that is high scores for both

segments, which is to be expected as both datasets contain homologs. The first row (0) represents the first segment and a cluster of separate RiPP precursors with relatively low scores at segment 0, but higher scores at other segments is clearly visible. Slight differences also seem to occur at other segments, most notably 7 and 9. These differences in conservation scores, only present in RiPPs, indicate low conservation at certain segments while high at others, whereas for non-RiPPs this pattern is more stable and outlier clusters are rare. However, most datapoints fall into the same space and no clear general distinction between RiPP and non-RiPP is distinguishable. Yet this distribution does suggest that at least for some RiPP conservation patterns there is difference compared to non-RiPPs.

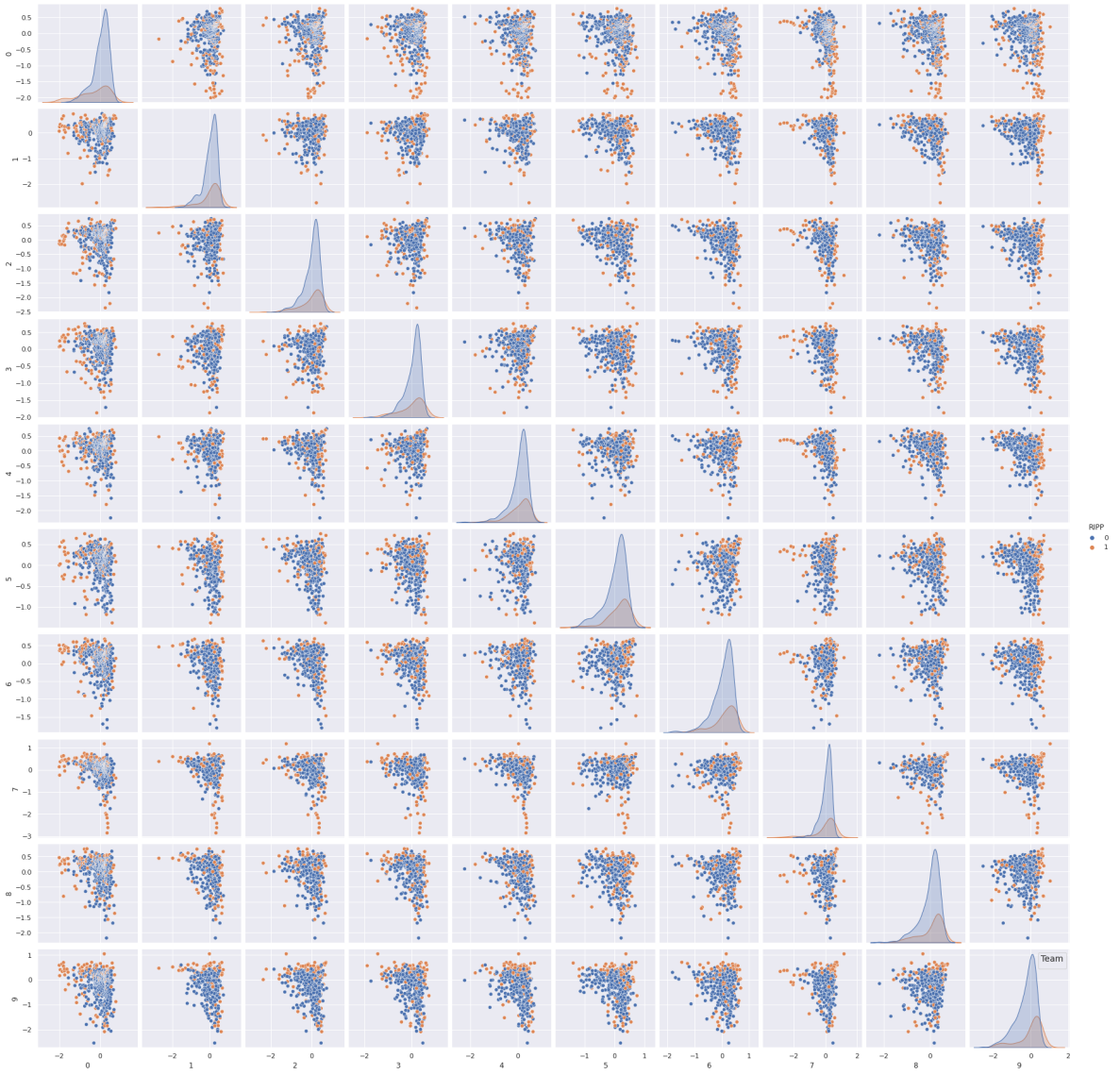
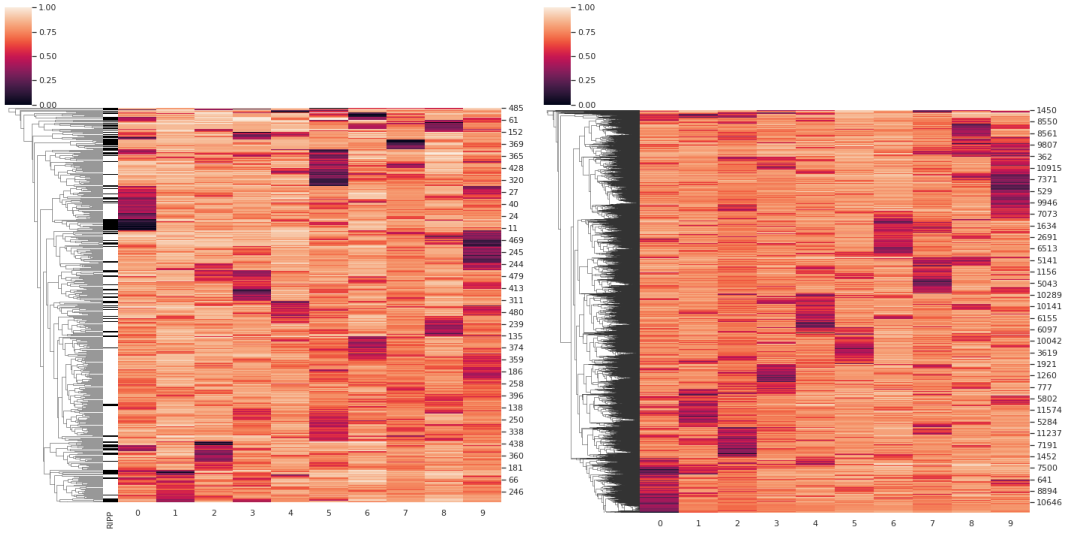


Figure 7: Pairplot of *training data* ARMON conservation scores with RiPP data in orange and non-RiPP data in blue where each box represents a scatterplot. Each box represents a segment vs a segment such that the top row represents 1/1, 1/2 ... 1/10 and the second row 2/1, 2/2, ... 2/10. etc. Graphs representing all data per column are in the diagonal. Positive training data (RiPPs) are in orange, whereas negative training data is in blue. (NB: The plot is symmetrical such that eg. 0/9 is the same plot as 9/0 mirrored.)

Clustering data with the seaborn built-in clustermap function for both *decRiPPter dataset* and *training dataset* yielded some clustering patterns shown in figure 8. As can be expected from figure 6 on conservation, a cluster of RiPPs exists with very low conservation scores on the first segment (8a). Also in accordance with figure 6 is that for the 10th segment there appears to be a larger

non-RiPP cluster of low conservation at segment 9. Overall RiPPs can not be said to cluster neatly into categories, maybe roughly into 3 or 4 thicker RiPP bands, but generally the RiPP landscape is fractured. Clustering on decRiPPter data (8b) did not yield any clear clustering either. Just like clustering on the training data, clustering manifests itself mostly per individual segment. The fact that clustering is based on grouping within single segments, that is a cluster on segment A does not coincide with a cluster on segment B, means that there is little correlation between segments. By default there is a correlation as conservation scores are relative, lower conservation at one segment by definition means higher conservation on the others on average, but there is no case where high conservation at one segment is countered by low conservation at another. Conservation seems to follow a pattern with one segment and then uniform distribution. This is expected as data consists sequences selected for homology which leaves little room for more than one segment to be of low conservation.



(a) *Training dataset* (b) *decRiPPter dataset*

Figure 8: Clustered heatmaps of datasets. Legend: conservation Z-score, min-max scaled to yield a value between 0 and 1. Dendrites/trees left of the graph indicate calculated hierarchy. Right of graph are the indices of data.

3.3 Self-organizing map

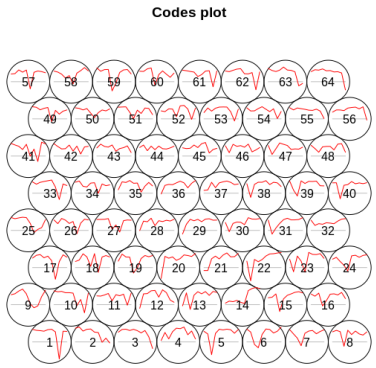
Accuracy and sensitivity of the constructed SOM were low with a maximum of 70% and an average of 63% on 10 iterations with a random seed, shown in table 2. Most notably with a very low ratio of true positive predictions. In fact, when the model would choose at random (0.5 chance) based on the distribution in the entire set being of ratio 0.26 positives, the ratio of true positives would be 0.14. A negative prediction from the SOM model has a 77% chance of beging correct, a positive prediction only a chance of 27%. This is close to the original distribution of the dataset and indicates that the model is almost random with an accuracy of only 63% on average; the mode performs poorly.

It was speculated that this is due to the fact that a SOM specializes in grouping similar data together, while RiPP conservation patterns turned out to be very dissimilar. To try to address this a re-run of the algorithm was performed using lanthipeptides alone, which were speculated to be more similar in conservation patterns, but this was unsuccessful as well.

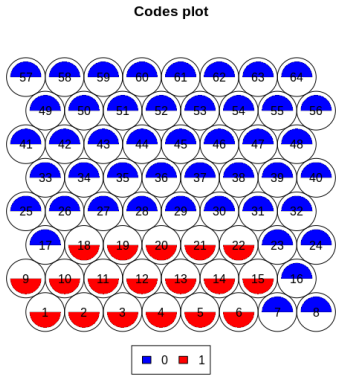
	Actual			
Predicted	0	1	predicted total	% correctly predicted
0	0.55	0.16	0.71	77%
1	0.21	0.08	0.29	27%
Actual total	0.76	0.24	1	63% Accuracy

Table 2: SOM predicted vs actual ratios. Average of 10 iterations of the algorithm.

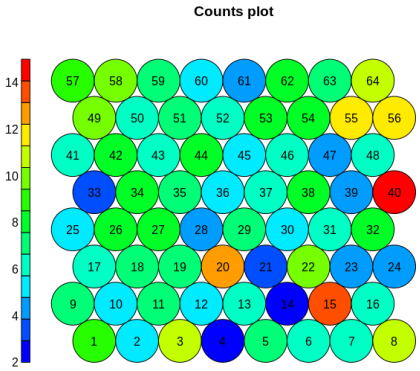
Even though the constructed self-organizing map performed badly on a classification task with held-out test data, interesting patterns emerged. Shown first in figure 9a is a codes plot with each of the 64 nodes shown with respective values for the 10 precursor segments represented as a line graph. Figure 9b shows the same nodes but classified as either RiPP or non-RiPP node. Recall that in prediction mode this classification is used to predict RiPPs. RiPP-nodes seem to have a tendency to have a lower conservation at the first segments, not unexpected given the knowledge from conservation scoring (figure 6) and hierarchical clustering (figure 8) that at least some RiPPs have characteristically low conservation at the first segments. This idea of at least some RiPPs having a low first conservation pattern but not all is further confirmed in figure 9c, counts, as the two largest nodes (15 and 20), clearly show a low-first conservation pattern in the codes plot. Interestingly, the largest node in figure 9c (40) also corresponds to a low-first conservation, but it is not classified as a RiPP node. Also interesting to note is that no nodes were empty, which with 494 datapoints as input points to a high level of spread, which means data is quite dissimilar. Figure 9d shows neighbour distance plotting, the higher the value the higher the distance to neighbour nodes. There is a clear increase in dissimilarity to neighbours on the boundary between RiPP and non-RiPP nodes, this indicates that RiPPs nodes are more similar to another and that non-RiPP nodes are similar to another and that the biggest difference is between RiPP nodes and non-RiPP nodes.



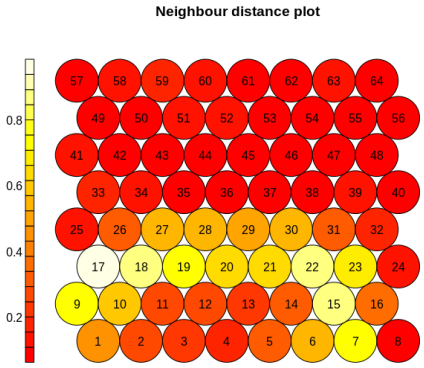
(a) SOM codes plot. Each node contains a conservation pattern



(b) Classification of nodes, either RiPP or non-RiPP



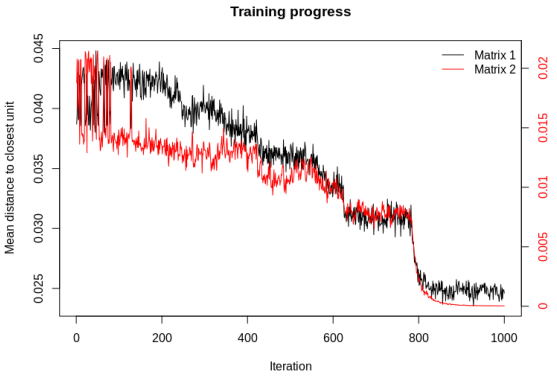
(c) Number of data points, RiPP and non-RiPP, per node.



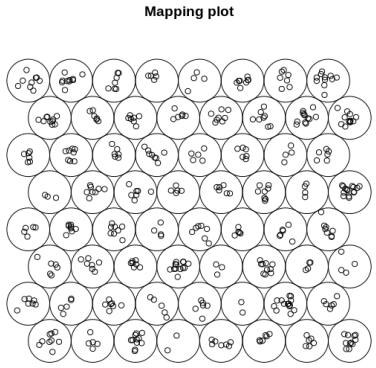
(d) Neighbour distance, a higher value indicates a higher distance to neighbouring nodes

Figure 9: SOM sub-maps generated with the Kohonen R package. Index is presented in each node.

Figure 10a shows the training rate of the SOM with 1000 iterations. An optimum on training is achieved after about 800 iterations, with matrix 2 learning approaching zero just after 800 iterations. Figure 10b shows mapping of data points onto the map itself. Each point is placed in node space, distance to the center of the node indicates similarity to other nodes, direction indicates to what nodes a datum is more similar.



(a) SOM training progress.



(b) Kohonen mapping plot. Each datum is plotted into a node, distance from center and direction indicates similarity to other nodes.

Figure 10

3.4 Random Forest & Decision Tree

As a first step to a random forest model, a decision tree model was constructed. Shown in figure figure 11 are the nodes of the decision tree with conditions at each split listed. As expected, feature 9 is the most important, followed by 6 and 7, both placed high in the decision tree. Feature importance scores of the decision tree are shown in figure 12a, with feature 9 clearly being the most important. Recall from chapter 3.1 Conservation Scores that non-RiPPs show specific low conservation on average at feature 9, also recall then that for RiPPs feature 0 shows low conservation on average. Whereas feature 9 is important as expected, feature 0 does not play a role at all in the decision tree with a feature importance score of zero. This is a surprising result, but theorized to be because of the hierarchical nature of a decision tree; that is when the model has already decided upon a classification based on feature 9 and others, no further information is gained by feature 0 and thus the importance score is zero. To test this the decision tree was ran excluding feature 9, resulting in a slightly higher score on feature 0 at 0.068, but still the lowest score. Repeating this step, shown in figure 12b, with the next most important feature 6 confirmed this hypothesis as feature 0 became the most important feature (0.219) after excluding both feature 6 and 9.

The confusion matrix for the decision tree model is shown in figure 13a with the model performing quite well. Especially false positives are low, with only 6.8% of true negatives falsely classified as positives. The model performs less well at positives with only a 56% ratio of true positives classified correctly. One potential explanation for this poor performance on true positives might be that the model actually trains on only part of the data, as maybe a subset of positive data is distinguishable from negative data, but another subset is not. The model could theoretically then perform quite well on this subset. Because the model is intended as a filter on the *decRiPPter dataset*, the focus is on largely excluding false positives as they are expected to be myriadly present in the *decRiPPter dataset*. Thus, the decision tree model seems to hold promise.

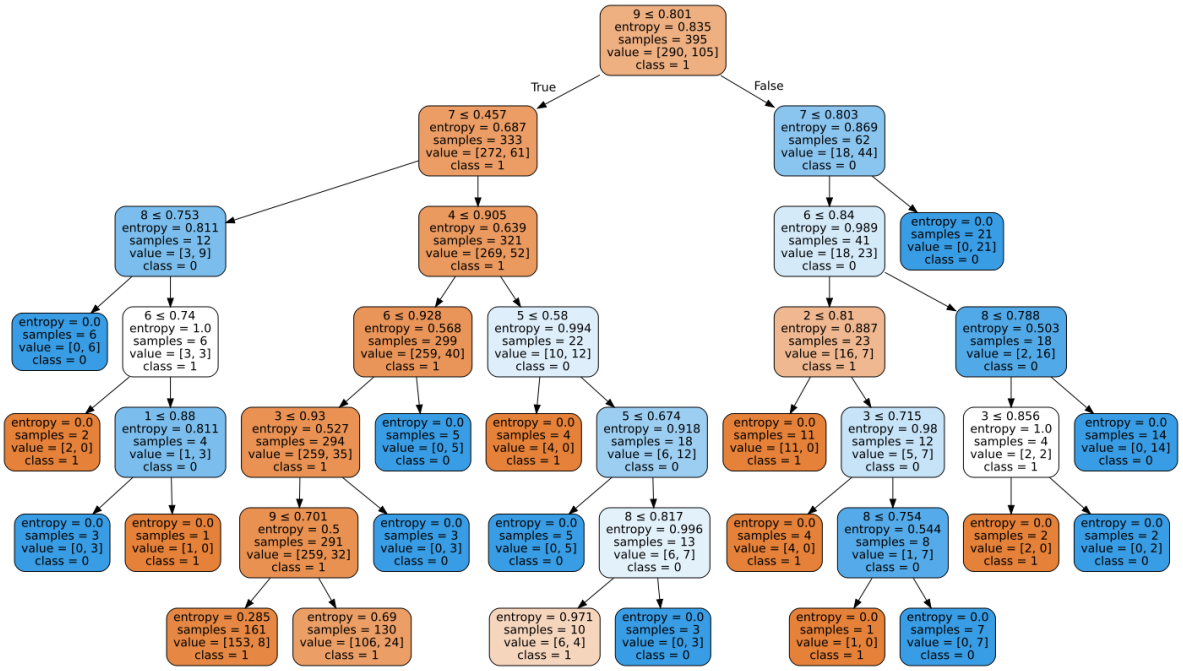
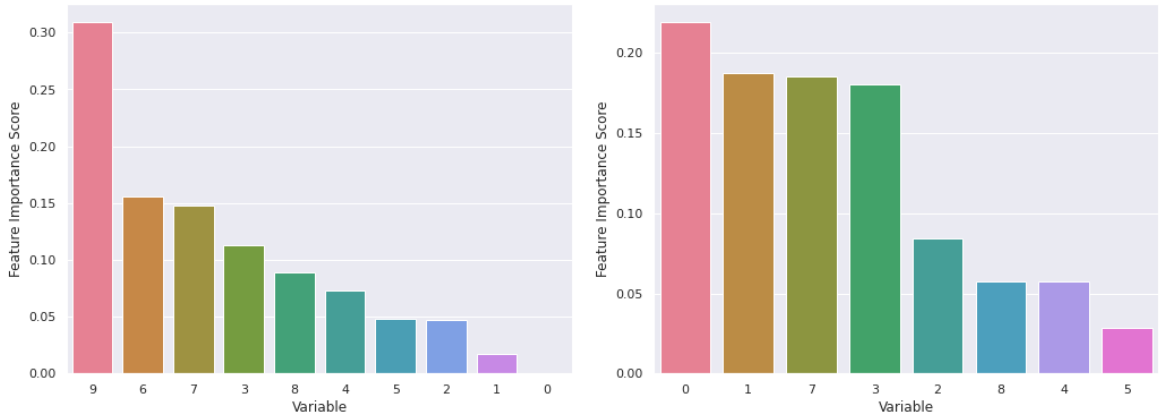


Figure 11: Schematic overview of decision tree model

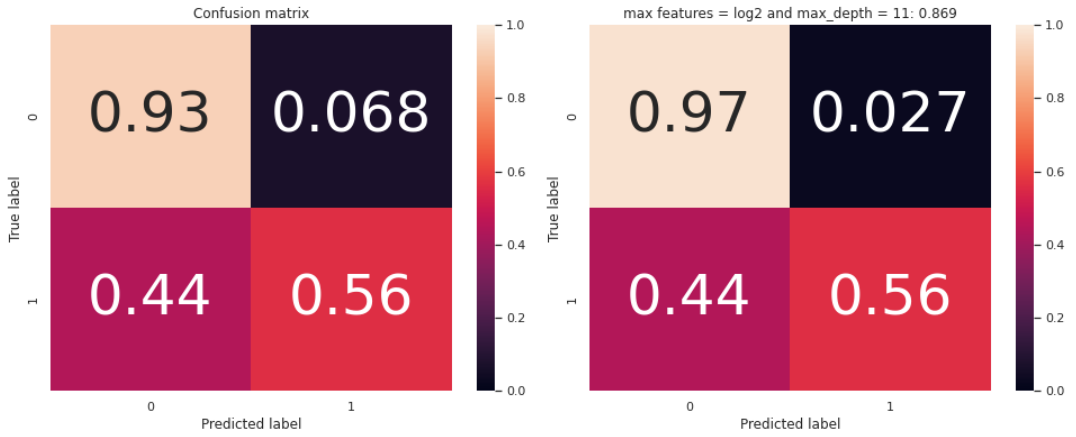


(a) Feature importance scores of the different features 0-9.

(b) Feature importance Scores of decision tree model excluding feature 9 and 6; the model flips

Figure 12: Feature Importance Scores of decision tree model

Decision trees are prone to overtraining though and thus a random forest model was subsequently constructed, which outperformed the decision tree by a little as can be seen in figure 13b. There is no difference in performance on true positives, but the false positive rate drops from 6.8% to 2.7%. This is quite low and thus the model is seems fit for use as a filter on decRiPPter data. It is also promising to see that the RF model classifies almost all true negatives correctly. A negative score on a datum in the *decRiPPter dataset* can therefore be considered trustworthy. 56% (73/130) of RiPP precursors from the training data were correctly identified by the model, whereas only 2.7% of non-RiPP peptides were incorrectly classified as such (10/365). In total 83 peptides were classified as RiPPs of which 10 were false positives. This is a strong enrichment in true positives compared to random chance. Thus, the RF model can be used as a scoring mechanism to prioritize putative RiPP precursors based on conservation score.

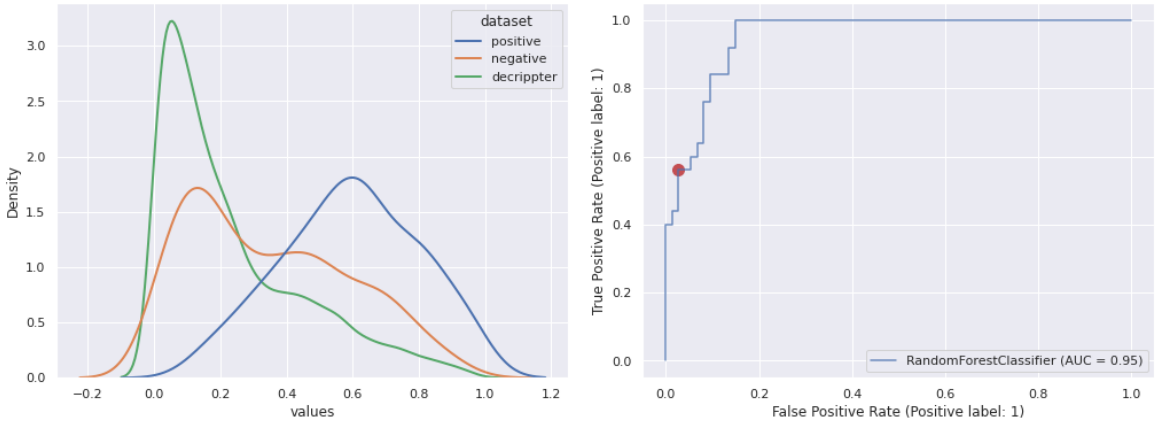


(a) confusion matrix decision tree

(b) confusion matrix random forest

Figure 13: Besides being less sensitive to overtraining, the random forest model slightly outperforms a decision tree on training data

The constructed random forest model was then used to predict RiPP precursors from the *decRiPPter dataset* shown in figure 14a, which resulted in a largely negative scoring. Especially very low chances in the range of 0-0.2 were common, indicating low similarity to RiPP conservation patterns. With threshold for classification set at 0.5, the model only classifies 1,336 out of 11,276 clusters as RiPPs. Also shown in figure 14a, as a means of validation of the model, positive and negative subsets of the *training dataset* were used for cross-validation, as if the training data was unseen by the model. A perfect model performance would see the positive subset and negative subset completely separated. Additionally, to asses whether any difference in predictions might show up per genus, predictions on each genus from the *decRiPPter dataset* are shown in figure 15. All genera show roughly the same distribution with maybe an outlier for *Akkermansia*, but since this is the smallest of genera with only 7 clusters this is not deemed very relevant.



(a) RF prediction scores on *decRiPPter dataset* and positive and negative subsets of *training dataset*.

(b) ROC curve for the random forest model. Red dot indicates a classification threshold of 0.5

Figure 14

However, the default threshold for classification at 0.5 would actually classify quite a large chunk of negative data as positive, therefore a slightly increased threshold of 0.6 might be more appropriate. Applying this threshold yields 833 positive predictions on *decRiPPter dataset*. Another approach focused on weeding out false positives - as is required for a filter on decRiPPter - would be to set the threshold low at for example 0.2, where almost all positive data from *training dataset* is classified correctly, which means that any score below 0.2 is almost certainly a true negative. Applying this threshold to predictions on *decRiPPter data* yields 4,595 positive predictions, of which a large amount

can be expected to be false positives. The negative predictions, 6,681 out of 11,276 datums, carry high accuracy though and this can be useful as a filter on decRiPPter as the number of sequences is greatly reduced. These thresholds are of course not set in stone and can be varied according to the intended use of the presented work.

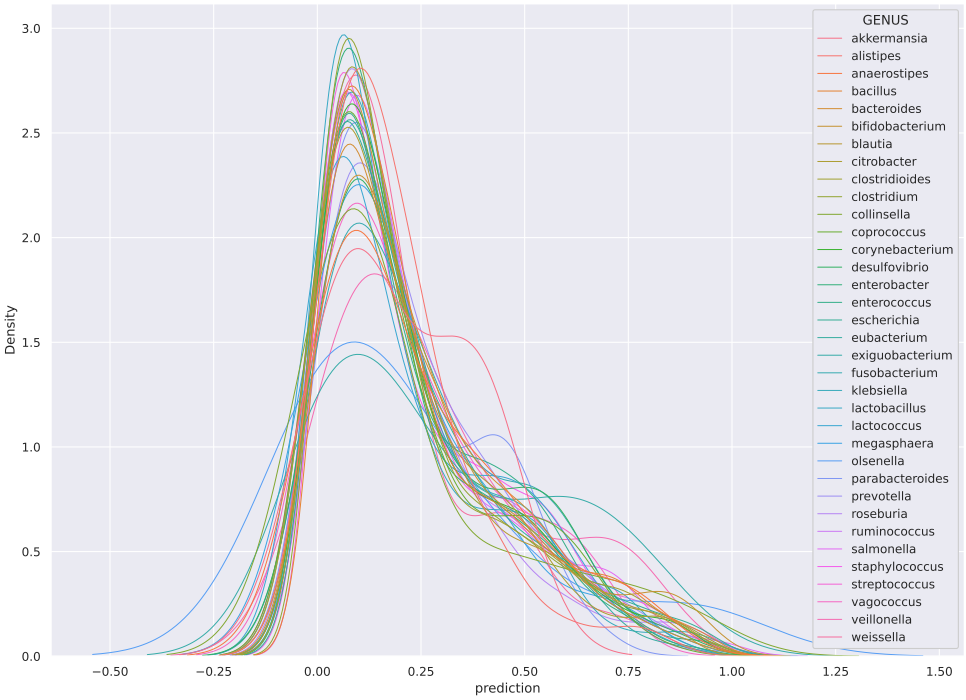
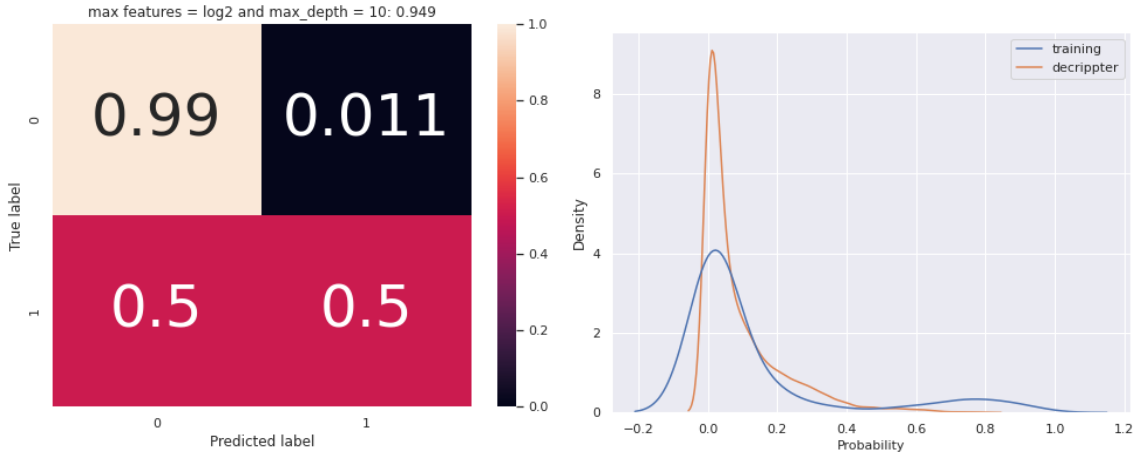


Figure 15: Kernel Density Estimation of RF prediction scores on *decRiPPter dataset* per genus.

Because it is suspected that RiPPs vary greatly in conservation, another RF model was trained on the RiPP-class lanthipeptides only. Lanthipeptides were chosen as they comprise a significant part of RiPPs in the *training dataset* with 63/130 sequences. With a threshold at 0.5 probability this resulted in 177/11,276 positive predictions from the *decRiPPter dataset* and 53 positive predictions on *training dataset*, with a comparable true positive rate as prediction of RiPPs in general. Prediction scores and confusion matrix for the RF model trained on lanthipeptides is shown in figure 16, note that the depth of the model is changed from 11 to 10 compared to the model trained on RiPPs in general. The same was tried for other classes but none were present in sufficient numbers for proper training. Other classes tried were thiopeptides (13 sequences), cyanobactins (12), lasso-peptides (4), and sactipeptides (2); recall that the total number of positive data was 130 sequences.



(a) Confusion matrix random forest for lanthipeptides (b) Predictions cores random forest for lanthipeptides

Figure 16: RF model prediction and confusion matrix for lanthipeptides

4 Discussion

Natural product mining holds great potential over classical culture based discovery. Especially with the coming of age of bioinformatics and the current torrent of genome data the future looks bright for the field of genome mining. Results presented here, as well as results from existing tools like decRiPPter, NeuRiPP[47], Rodeo[48], and RiPPER[49], already show that machine learning tools are very useful in detecting RiPPs - and as machine learning methods are particularly well suited to handling large amounts of data, this is expected to increase. However, as an exploratory tool, decRiPPter prizes novelty over accuracy and a large amount of false positives is thus expected. An algorithm to prioritize putative RiPP precursors based on conservation patterns is presented here.

Two machine learning models were constructed based on enriched conservation data. One of which, the random forest model, is able to distinguish RiPPs from non-RiPPs quite well. Especially for the purpose of filtering decRiPPter results, the random forest model can be useful as it predicts few false positives on held-out training data and is able to distinguish negative and positive training data in cross-validation experiments. Results from the self-organizing-map(SOM) model show that RiPPs do not share straightforward shared conservation patterns and that classification based on similarity is difficult. The SOM model showed that RiPPs have greater variability between them than non-RiPPs. It is suggested that because of this high variation and the fact that a SOM mostly builds on clusters of similarity, the SOM model failed to produce accurate predictions on training data and was discontinued as a model.

Even though the RF model holds promise, in order to be useful however the model needs to be more specific and better be able to separate supervised positive and negative data. The model could benefit from more sophisticated scoring methods which could for example instead of just trimming gaps only penalize them. Another interesting approach might be scoring based on amino acid class or electronegativity. The current model is quite flexible and with some tweaking it would be relatively easy to reproduce conservational patterns on for example hydrophobicity and turn propensity. It is also interesting to note that conservation scoring methods are not always in agreement, there is variation and outliers are present. Random forests are particularly well suited to handle data of different sign and class and future work might benefit from grouping all data from all conservation scoring methods together. For example earlier mentioned conservation scores on hydrophobicity could be added. Conservation scoring based on sequence similarity is fast enough with tools like Diamond 2.0 and length restrictions on reference databases, but improvements could be made by using protein similarity networks. This might even be necessary because already the homology search step is by far the most computationally intensive step. As more data becomes available, more sequences are to be enriched and as databases expand exponentially at the same time, the search space and thus computational workload of the presented algorithm will increase. This is true already for precursor peptides alone, a similar analysis for conservation patterns in enzymes - which are larger and more numerous than precursors in BGC's - is unfeasible at this point. Methods like enzyme similarity networks might therefore replace direct homology searches in order to constrain computational limits, but might also be more sensitive.

Another idea worth exploring would be to split the data into less or more than 10 bins per peptide. The number of bins can theoretically be increased to the length of the smallest peptide. This would provide the RF model with a lot more data, which might increase accuracy. However, it could also be that an increase in data does not provide more accuracy as the signal is just not there. It has been speculated that RiPPs are diverse, results presented here confirm that conservation patterns in RiPPs are diverse accordingly. It could be that RiPPs are just too divergent to be captured by a RF model under the single classification 'RiPP'. Indeed training on lanthipeptides alone increased the models accuracy slightly. A better approach might be try and identify single classes of RiPPs or to first cluster RiPPs into categories by conservation pattern and then training a RF model by cluster specifically. A SOM might be a good candidate for such clustering, but this requires a lot more positive training data.

The method of conservation scoring and prediction presented here aids detection of natural products in vast amounts of genomic data. Currently the model is trained on only 130 confirmed RiPP precursors from the MIBIG repository, but as more RiPPs continue to be experimentally validated or annotated this number is expected to grow considerably in forthcoming years. The latest version of MIBIG (3.0) [50] is a most recent example, but also efforts to explore biomes like the deep ocean [27] or mapping

underexplored fungal species [2] from the unruly edges of human society can not only yield a new appreciation of nature itself, but troves of useful data on natural products as well.[51]

Not only is the amount of solid training data expected to grow, new methods for detecting RiPPs - and other natural products - are constantly being devised. A paradigm shift from genomics based detection towards a more integrative approach including metagenome, transcriptome, and metabolome data is currently underway, with promising results on transcriptome analysis already[52]. Future RiPP prediction are expected to be more streamlined, yet sophisticated, as new algorithms - like the one presented here - can be integrated into familiar platforms like antiSMASH.

Maybe, hopefully, new data, and new ideas and paradigms will allow the huge resources of nature to be tapped and help induce a new "Golden Age" in natural product discovery, which will not only fuel economic growth and an increase in living standards, but could also help stave off the forthcoming crisis in antibiotic resistance.

[2] SPUN, https://www.spun.earth/

References

[1] Marnix H. Medema, Tristan de Rond, and Bradley S. Moore. Mining genomes to illuminate the specialized chemistry of life. *Nature Reviews Genetics*, 22(9):553–571, Sep 2021.

[2] David J. Newman and Gordon M. Cragg. Natural products as sources of new drugs over the nearly four decades from 01/1981 to 09/2019. *Journal of Natural Products*, 83(3):770–803, 2020. PMID: 32162523.

[3] Essaid Ait Barka, Parul Vatsa, Lisa Sanchez, Nathalie Gaveau-Vaillant, Cedric Jacquard, Hans-Peter Klenk, Christophe Clément, Yder Ouhdouch, and Gilles P. van Wezel. "taxonomy, physiology, and natural products of actinobacteria". *Microbiology and Molecular Biology Reviews*, 80(1):1–43, 2016.

[4] Jiaoyang Jiang, Xiaofei He, and David E. Cane. Biosynthesis of the earthy odorant geosmin by a bifunctional streptomyces coelicolor enzyme. *Nature chemical biology*, 3(11):711–715, Nov 2007. 17873868[pmid].

[5] Leonard Katz and Richard H Baltz. Natural product discovery: past, present, and future. *Journal of Industrial Microbiology and Biotechnology*, 43(2-3):155–176, 03 2016.

[6] Matthew A. Cooper and David Shlaes. Fix the antibiotics pipeline. *Nature*, 472(7341):32–32, Apr 2011.

[7] Cameron R. Pye, Matthew J. Bertin, R. Scott Lokey, William H. Gerwick, and Roger G. Linington. Retrospective analysis of natural products provides insights for future discovery trends. *Proceedings of the National Academy of Sciences*, 114(22):5601–5606, 2017.

[8] Philip Hunter. Harnessing nature's wisdom. turning to nature for inspiration and avoiding her follies. *EMBO reports*, 9(9):838–840, Sep 2008. 18762775[pmid].

[9] Micheal C. Wilson and Jörn Piel. Metagenomic approaches for exploiting uncultivated bacteria as a resource for novel biosynthetic enzymology. *Chemistry & Biology*, 20(5):636–647, 2013.

[10] Staley, j. t., and konopka, a. (1985). measurement of in situ activities of nonphotosynthetic microorganisms in aquatic and terrestrial habitats. annual review of microbiology, 39, 321–346. https://doi.org/10.1146/annurev.mi.39.100185.001541.

[11] Kim Lewis. Platforms for antibiotic discovery. *Nature Reviews Drug Discovery*, 12(5):371–387, May 2013.

[12] Karen G. Lloyd, Andrew D. Steen, Joshua Ladau, Junqi Yin, and Lonnie Crosby. Phylogenetically novel uncultured microbial cells dominate earth microbiomes. *mSystems*, 3(5):e00055–18, Sep 2018. 30273414[pmid].

[13] Michael Fischbach and Christopher A. Voigt. Prokaryotic gene clusters: A rich toolbox for synthetic biology. *Biotechnology Journal*, 5(12):1277–1296, 2010.

[14] Christian S. Riesenfeld, Robert M. Goodman, and Jo Handelsman. Uncultured soil bacteria are a reservoir of new antibiotic resistance genes. *Environmental Microbiology*, 6(9):981–989, 2004.

[15] Mitja M. Zdouc, Mohammad M. Alanjary, Guadalupe S. Zarazúa, Sonia I. Maffioli, Max Crüsemann, Marnix H. Medema, Stefano Donadio, and Margherita Sosio. A biaryl-linked tripeptide from planomonospora reveals a widespread class of minimal ripp gene clusters. *Cell Chemical Biology*, 28(5):733–739.e4, 2021.

[16] J E González-Pastor, J L San Millán, M A Castilla, and F Moreno. Structure and organization of plasmid genes required to produce the translation inhibitor microcin C7. *J Bacteriol*, 177(24):7131–7140, December 1995.

[17] Antal Kiss, Gabriella Balikó, Attila Csorba, Tungalag Chuluunbaatar, Katalin F Medzihradszky, and Lajos Alföldi. Cloning and characterization of the DNA region responsible for megacin A-216 production in bacillus megaterium 216. *J Bacteriol*, 190(19):6448–6457, August 2008.

[18] Paul G. Arnison, Mervyn J. Bibb, Gabriele Bierbaum, Albert A. Bowers, Tim S. Bugni, Grzegorz Bulaj, Julio A. Camarero, Dominic J. Campopiano, Gregory L. Challis, Jon Clardy, Paul D. Cotter, David J. Craik, Michael Dawson, Elke Dittmann, Stefano Donadio, Pieter C. Dorrestein, Karl-Dieter Entian, Michael A. Fischbach, John S. Garavelli, Ulf Göransson, Christian W. Gruber, Daniel H. Haft, Thomas K. Hemscheidt, Christian Hertweck, Colin Hill, Alexander R. Horswill, Marcel Jaspars, Wendy L. Kelly, Judith P. Klinman, Oscar P. Kuipers, A. James Link, Wen Liu, Mohamed A. Marahiel, Douglas A. Mitchell, Gert N. Moll, Bradley S. Moore, Rolf Müller, Satish K. Nair, Ingolf F. Nes, Gillian E. Norris, Baldomero M. Olivera, Hiroyasu Onaka, Mark L. Patchett, Joern Piel, Martin J. T. Reaney, Sylvie Rebuffat, R. Paul Ross, Hans-Georg Sahl, Eric W. Schmidt, Michael E. Selsted, Konstantin Severinov, Ben Shen, Kaarina Sivonen, Leif Smith, Torsten Stein, Roderich D. Süssmuth, John R. Tagg, Gong-Li Tang, Andrew W. Truman, John C. Vederas, Christopher T. Walsh, Jonathan D. Walton, Silke C. Wenzel, Joanne M. Willey, and Wilfred A. van der Donk. Ribosomally synthesized and post-translationally modified peptide natural products: overview and recommendations for a universal nomenclature. *Natural product reports*, 30(1):108–160, Jan 2013. 23165928[pmid].

[19] Manuel Montalbán-López, Thomas A. Scott, Sangeetha Ramesh, Imran R. Rahman, Auke J. van Heel, Jakob H. Viel, Vahe Bandarian, Elke Dittmann, Olga Genilloud, Yuki Goto, María José Grande Burgos, Colin Hill, Seokhee Kim, Jesko Koehnke, John A. Latham, A. James Link, Beatriz Martínez, Satish K. Nair, Yvain Nicolet, Sylvie Rebuffat, Hans-Georg Sahl, Dipti Sareen, Eric W. Schmidt, Lutz Schmitt, Konstantin Severinov, Roderich D. Süssmuth, Andrew W. Truman, Huan Wang, Jing-Ke Weng, Gilles P. van Wezel, Qi Zhang, Jin Zhong, Jörn Piel, Douglas A. Mitchell, Oscar P. Kuipers, and Wilfred A. van der Donk. New developments in ripp discovery, enzymology and engineering. *Nat. Prod. Rep.*, 38:130–239, 2021.

[20] Rustem Khusainov and Oscar P. Kuipers. When the leader gets loose: In vivo biosynthesis of a leaderless prenisin is stimulated by a trans-acting leader peptide. *ChemBioChem*, 13(16):2433–2438, 2012.

[21] Trent J. Oman and Wilfred A. van der Donk. Follow the leader: the use of leader peptides to guide natural product biosynthesis. *Nature chemical biology*, 6(1):9–18, Jan 2010. 20016494[pmid].

[22] Sebastian W. Fuchs, Gerald Lackner, Brandon I. Morinaka, Yohei Morishita, Teigo Asai, Sereina Riniker, and Jörn Piel. A lanthipeptide-like n-terminal leader region guides peptide epimerization by radical sam epimerases: Implications for ripp evolution. *Angewandte Chemie - International Edition*, 55(40):12330–12333, September 2016.

[23] Graham A Hudson and Douglas A Mitchell. Ripp antibiotics: biosynthesis and engineering potential. *Current Opinion in Microbiology*, 45:61–69, 2018. Antimicrobials * Microbial systems biology.

[24] Brandon J. Burkhart, Graham A. Hudson, Kyle L. Dunbar, and Douglas A. Mitchell. A prevalent peptide-binding domain guides ribosomal natural product biosynthesis. *Nature chemical biology*, 11(8):564–570, Aug 2015. 26167873[pmid].

[25] Beata M. Wieckowski, Julian D. Hegemann, Andreas Mielcarek, Linda Boss, Olaf Burghaus, and Mohamed A. Marahiel. The pqqd homologous domain of the radical sam enzyme thnb is required for thioether bond formation during thurincin h maturation. *FEBS Letters*, 589(15):1802–1806, 2015.

[26] John A. McIntosh, Mohamed S. Donia, and Eric W. Schmidt. Ribosomal peptide natural products: bridging the ribosomal and nonribosomal worlds. *Natural product reports*, 26(4):537–559, Apr 2009. 19642421[pmid].

[27] Lucas Paoli, Hans-Joachim Ruscheweyh, Clarissa C. Forneris, Florian Hubrich, Satria Kautsar, Agneya Bhushan, Alessandro Lotti, Quentin Clayssen, Guillem Salazar, Alessio Milanese, Charlotte I. Carlström, Chrysa Papadopoulou, Daniel Gehrig, Mikhail Karasikov, Harun Mustafa, Martin Larralde, Laura M. Carroll, Pablo Sánchez, Ahmed A. Zayed, Dylan R. Cronin, Silvia G. Acinas, Peer Bork, Chris Bowler, Tom O. Delmont, Josep M. Gasol, Alvar D. Gossert, André Kahles, Matthew B. Sullivan, Patrick Wincker, Georg Zeller, Serina L. Robinson, Jörn Piel, and

Shinichi Sunagawa. Biosynthetic potential of the global ocean microbiome. *Nature*, 607(7917):111–118, Jul 2022.

[28] Alexander Crits-Christoph, Spencer Diamond, Cristina N. Butterfield, Brian C. Thomas, and Jillian F. Banfield. Novel soil bacteria possess diverse genes for secondary metabolite biosynthesis. *Nature*, 558(7710):440–444, Jun 2018.

[29] Kai Blin, Simon Shaw, Alexander M Kloosterman, Zach Charlop-Powers, Gilles P van Wezel, Marnix H Medema, and Tilmann Weber. antiSMASH 6.0: improving cluster detection and comparison capabilities. *Nucleic Acids Research*, 49(W1):W29–W35, 05 2021.

[30] Alexander M. Kloosterman, Peter Cimermancic, Somayah S. Elsayed, Chao Du, Michalis Hadjithomas, Mohamed S. Donia, Michael A. Fischbach, Gilles P. van Wezel, and Marnix H. Medema. Expansion of ripp biosynthetic space through integration of pan-genomics and machine learning uncovers a novel class of lanthipeptides. *PLoS biology*, 18(12):e3001026–e3001026, Dec 2020. 33351797[pmid].

[31] S. Quiroga. Ripp-space search using machine learning approaches in healthy and ibd subject gut meta-omic data. *contact: prof. M. Donia, Princeton University*, 2022.

[32] Satria A Kautsar, Kai Blin, Simon Shaw, Jorge C Navarro-Muñoz, Barbara R Terlouw, Justin J J van der Hooft, Jeffrey A van Santen, Vittorio Tracanna, Hernando G Suarez Duran, Victòria Pascal Andreu, Nelly Selem-Mojica, Mohammad Alanjary, Serina L Robinson, George Lund, Samuel C Epstein, Ashley C Sisto, Louise K Charkoudian, Jérôme Collemare, Roger G Linington, Tilmann Weber, and Marnix H Medema. MIBiG 2.0: a repository for biosynthetic gene clusters of known function. *Nucleic Acids Research*, 48(D1):D454–D458, 10 2019.

[33] The UniProt Consortium. UniProt: the universal protein knowledgebase in 2021. *Nucleic Acids Research*, 49(D1):D480–D489, 11 2020.

[34] N.L.L. Louwen. Genome mining for novel ribosomally synthesized and post-translationally modified peptide families in the gut microbiome. *unpublished, available at request at nico.louwen@wur.nl or marnix.medema@wur.nl*, 2022.

[35] Benjamin Buchfink, Klaus Reuter, and Hajk-Georg Drost. Sensitive protein alignments at tree-of-life scale using diamond. *Nature Methods*, 18(4):366–368, Apr 2021.

[36] Alexander Pertsemlidis and John W. Fondon. Having a blast with bioinformatics (and avoiding blastphemy). *Genome Biology*, 2(10):reviews2002.1, Sep 2001.

[37] Robert C. Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, 32(5):1792–1797, 03 2004.

[38] Salvador Capella-Gutiérrez, José M. Silla-Martínez, and Toni Gabaldón. trimAl: a tool for automated alignment trimming in large-scale phylogenetic analyses. *Bioinformatics*, 25(15):1972–1973, 06 2009.

[39] Andrew M. Waterhouse, James B. Procter, David M. A. Martin, Michèle Clamp, and Geoffrey J. Barton. Jalview Version 2—a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, 25(9):1189–1191, 01 2009.

[40] Agnieszka Golicz, Peter V. Troshin, Fábio Madeira, David M. A. Martin, James B. Procter, and Geoffrey J. Barton. Aacon: A fast amino acid conservation calculation service. 2018. "submitted paper".

[41] William S.J. Valdar. Scoring residue conservation. *Proteins: Structure, Function, and Bioinformatics*, 48(2):227–241, 2002.

[42] Michael L. Waskom. seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021.

[43] Teuvo Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):59–69, Jan 1982.

[44] Ron Wehrens and Lutgarde M. C. Buydens. Self- and super-organizing maps in R: The kohonen package. *Journal of Statistical Software*, 21(5):1–19, 2007.

[45] Ron Wehrens and Johannes Kruisselbrink. Flexible self-organizing maps in kohonen 3.0. *Journal of Statistical Software*, 87(7):1–18, 2018.

[46] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[47] Emmanuel L. C. de los Santos. Neuripp: Neural network identification of ripp precursor peptides. *Scientific Reports*, 9(1):13406, Sep 2019.

[48] Jonathan I Tietz, Christopher J Schwalen, Parth S Patel, Tucker Maxson, Patricia M Blair, Hua-Chia Tai, Uzma I Zakai, and Douglas A Mitchell. A new genome-mining tool redefines the lasso peptide biosynthetic landscape. *Nat Chem Biol*, 13(5):470–478, February 2017.

[49] Javier Santos-Aberturas, Govind Chandra, Luca Frattaruolo, Rodney Lacret, Thu H Pham, Natalia M Vior, Tom H Eyles, and Andrew W Truman. Uncovering the unexplored diversity of thioamidated ribosomal peptides in Actinobacteria using the RiPPER genome mining tool. *Nucleic Acids Research*, 47(9):4624–4637, 03 2019.

[50] Barbara R Terlouw, Kai Blin, Jorge C Navarro-Muñoz, Nicole E Avalon, Marc G Chevrette, Susan Egbert, Sanghoon Lee, David Meijer, Michael J J Recchia, Zachary L Reitz, Jeffrey A van Santen, Nelly Selem-Mojica, Thomas Tørring, Liana Zaroubi, Mohammad Alanjary, Gajender Aleti, César Aguilar, Suhad A A Al-Salihi, Hannah E Augustijn, J Abraham Avelar-Rivas, Luis A Avitia-Domínguez, Francisco Barona-Gómez, Jordan Bernaldo-Agüero, Vincent A Bielinski, Friederike Biermann, Thomas J Booth, Victor J Carrion Bravo, Raquel Castelo-Branco, Fernanda O Chagas, Pablo Cruz-Morales, Chao Du, Katherine R Duncan, Athina Gavriilidou, Damien Gayrard, Karina Gutiérrez-García, Kristina Haslinger, Eric J N Helfrich, Justin J J van der Hooft, Afif P Jati, Edward Kalkreuter, Nikolaos Kalyvas, Kyo Bin Kang, Satria Kautsar, Wonyong Kim, Aditya M Kunjapur, Yong-Xin Li, Geng-Min Lin, Catarina Loureiro, Joris J R Louwen, Nico L L Louwen, George Lund, Jonathan Parra, Benjamin Philmus, Bita Pourmohsenin, Lotte J U Pronk, Adriana Rego, Devasahayam Arokia Balaya Rex, Serina Robinson, L Rodrigo Rosas-Becerra, Eve T Roxborough, Michelle A Schorn, Darren J Scobie, Kumar Saurabh Singh, Nika Sokolova, Xiaoyu Tang, Daniel Udwary, Aruna Vigneshwari, Kristiina Vind, Sophie P J M Vromans, Valentin Waschulin, Sam E Williams, Jaclyn M Winter, Thomas E Witte, Huali Xie, Dong Yang, Jingwei Yu, Mitja Zdouc, Zheng Zhong, Jérôme Collemare, Roger G Linington, Tilmann Weber, and Marnix H Medema. MIBiG 3.0: a community-driven effort to annotate experimentally validated biosynthetic gene clusters. *Nucleic Acids Research*, 11 2022. gkac1049.

[51] Anna Tsing. Unruly edges: Mushrooms as companion species. *Environmental Humanities*, 1:141–154, 11 2012.

[52] Jennifer H. Wisecaver, Alexander T. Borowsky, Vered Tzin, Georg Jander, Daniel J. Kliebenstein, and Antonis Rokas. A global coexpression network approach for connecting genes to specialized metabolic pathways in plants. *The Plant cell*, 29(5):944–959, May 2017. 28408660[pmid].