Reinforcement learning to optimize fungal Biosynthetic Gene Clusters

Hayda Almeida [1 2] **Adrian Tsang** [1 2] **Abdoulaye Baniré Diallo** [1 3]

Abstract

Biosynthetic Gene Clusters (BGCs) encode metabolic pathway genes that produce secondary metabolites, compounds which are vital sources for the production and discovery of many drugs. State-of-the-art tools to identify BGCs face limitations to accurately predict cluster boundaries and composition.This work proposes a reinforcement learning approach that relies on protein domains, and integrates functional annotations curated from experts to optimize predictions from BGC discovery tools. Our approach yields an increase above 15% in gene precision and above 25% in cluster precision of BGC predictions from state-of-the-art tools.

1. Background

Secondary metabolites produced by fungi are an important source of bioactive compounds. These compounds are of particular interest in the pharmaceutical industry for the production of various medications(Kjærbølling et al., 2019) such as antibiotics, immunossupressants, anti-tumor, and antifungal drugs. Biosynthetic pathways producing these compounds are often encoded by gene clusters known as Biosynthetic Gene Clusters (BGCs) (Keller, 2019; Kautsar et al., 2020). BGCs generally contain minimal components such as backbone and tailoring enzymes. They may also contain other components such as transporters, transcription factors, and hypothetical proteins. (Keller, 2015; 2019).

Due to the genomic diversity of fungal genomes, accurately identifying BGCs in fungi remains a challenging task(Chavali & Rhee, 2017; Kjærbølling et al., 2019). Fungal BGCs can vary substantially in composition and location even among related species (Keller, 2019; Kjærbølling et al., 2020; Evdokias et al., 2021). Previous state-of-the-art BGC discovery tools such as fungiSMASH (Blin et al., 2021), DeepBGC (Hannigan et al., 2019), and TOUCAN (Almeida et al., 2020) often overpredict cluster boundaries, which can make the manual curation and chemical synthesis of these compounds more labor intensive. In this work, we propose a reinforcement learning approach based on Pfam (El-Gebali et al., 2019) protein domains and functional annotations to support improving the discovery of BGC regions in fungi.

2. Methodology

A Q-learner (Watkins & Dayan, 1992) is trained on Pfam protein domains obtained from publicly available benchmark BGC datasets (Almeida et al., 2019). Training is based on the domain occurrences in publicly available benchmark BGC datasets (details in (Almeida et al., 2019)) composed of annotated BGC instances obtained from MIBiG (Minimum Information about a Biosynthetic Gene cluster) (Kautsar et al., 2020), and synthetic non-BGC instances built from OrthoDB (Kriventseva et al., 2018) fungal orthologous genes. The reinforcement learning agent is tested on candidate BGCs generated from the *Aspergillus niger* NRRL3 genomic sequence (`https://gb.fungalgenomics.ca/portal`), obtained by extracting sequential 10,000 amino acid sliding windows with a 30% overlap (see details in (Almeida et al., 2020)). Candidate BGC predictions for *A. niger* are obtained with three state-of-the-art tools: fungiSMASH, DeepBGC, and TOUCAN.

A majority vote pre-processing step is performed in state-of-the-art candidate BGC predictions before they are optimized by the proposed reinforcement learning method. The majority vote pre-processing handles potentially duplicated predicted regions based on a local consensus. A gene g appearing in m candidate BGC predictions is represented as a label vector $L = l_0, l_1, ..., l_m$, where l_i is the prediction label (0 for predicted as non-BGC, and 1 otherwise). A majority vote score v_{score} is computed as the average value of its predicted labels $\overline{L}$, and sequential genes with $v_{score} \geq 0.5$ are merged. A set of 85 manually curated *A. niger* BGCs (further details described in (Inglis et al., 2013)) was considered as gold standard for our evaluation.

Reinforcement learning method For the task of optimzing BGC composition, a protein domain d is represented as an occurrence vector $C = c_0, c_1, ..., c_n$, for a dataset of n training instances, in which c_i holds the domain occurrence in a given instance ($c_i > 0$ if a BGC instance, and $c_i < 0$ if synthetic non-BGC). A reinforcement learning agent is

[1] Department d'Informatique, UQAM, Montréal, QC, Canada [2] Center for Structural and Funcional Genomics, Concordia University, Montréal, QC, Canada [3] Centre (CERMO), Montréal, QC, Canada. Correspondence to: Abdoulaye Baniré Diallo <diallo.abdoulaye@uqam.ca>.

The 2022 ICML Workshop on Computational Biology. Baltimore, Maryland, USA, 2022. Copyright 2022 by the author(s).

trained on the protein domain occurrence to decide on the best fitting action within a set of actions $A = keep, skip$ for the set of states of a candidate BGC, represented by the Pfam domains within each gene. The learner receives a reward R for each action choice. Before computing rewards per action, s_{keep} and s_{skip} scores are obtained for a domain d as in:

$$s_{keep} = \sum_{x \in C} \frac{x}{|C|} \qquad s_{skip} = |1 - s_{keep}|$$

Then rewards per action R_{keep} and R_{skip} for a domain d are computed considering the domain score and a $keepSkip$ threshold as in:

$$R_{keep}, R_{skip} = \begin{cases} s_{keep}, -s_{keep} & \text{if } s_{keep} > (s_{skip} * keepSkip) \\ -s_{skip}, s_{skip} & \text{otherwise.} \end{cases}$$

A penalty is given to the agent when it decides on an action with negative rewards $R < 0$. The total penalty is computed after an episode, meaning when the agent has processed the complete training set. Often genes composing candidate BGCs present multiple domains. When evaluating candidate BGCs, the optimal action for a gene g containing a set of domains $D = d_0, d_1, ...d_n$ for n domains found in g is computed by the reinforcement learning agent as in:

$$g_a = argmax(\sum_{i=0}^{n} d_i(R_{keep}), \sum_{i=0}^{n} d_i(R_{skip}))$$

Candidate BGC genes with $R_{skip} > R_{keep}$ receive the action $g_a = skip$, and be skipped. Otherwise they receive $g_a = keep$, and will be maintained.

Integrating functional annotations Components forming BGCs play different roles in the clusters. Generally, minimal BGC building blocks are backbone enzymes, responsible for defining the core chemical compound produced, and tailoring enzymes, capable of modifying the core compound and create variants (Keller, 2019). Other known components are cluster-specific transcription factors, transporters, and hypothetical proteins (Keller, 2015).

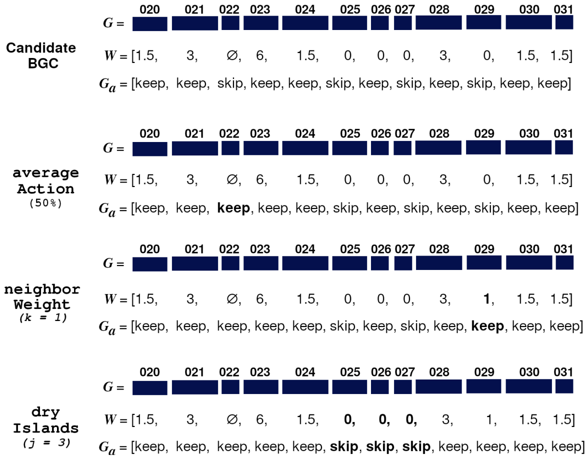
Candidate BGC
020 021 022 023 024 025 026 027 028 029 030 031
G =
W = [1.5, 3, ∅, 6, 1.5, 0, 0, 0, 3, 0, 1.5, 1.5]
Ga = [keep, keep, skip, keep, keep, skip, keep, skip, keep, skip, keep, keep]
average Action (50%)
020 021 022 023 024 025 026 027 028 029 030 031
G =
W = [1.5, 3, ∅, 6, 1.5, 0, 0, 0, 3, 0, 1.5, 1.5]
Ga = [keep, keep, keep, keep, keep, skip, keep, skip, keep, skip, keep, keep]
neighbor Weight (k = 1)
020 021 022 023 024 025 026 027 028 029 030 031
G =
W = [1.5, 3, ∅, 6, 1.5, 0, 0, 0, 3, 1, 1.5, 1.5]
Ga = [keep, keep, keep, keep, keep, skip, keep, skip, keep, keep, keep, keep]
dry Islands (j = 3)
020 021 022 023 024 025 026 027 028 029 030 031
G =
W = [1.5, 3, ∅, 6, 1.5, 0, 0, 0, 3, 1, 1.5, 1.5]
Ga = [keep, keep, keep, keep, keep, skip, skip, skip, keep, keep, keep, keep]

Figure 1. Examples of functional annotation strategies

Components within the set of 85 *A. niger* gold standard BGCs were manually curated with their functional annotations, and Pfam protein domaisn were extracted from gold standard genes annotated with a BGC component role. Three strategies, as shown in Figure 1, were developed to integrate functional annotations in the optimization of candidate BGCs. Genes lacking Pfam domains are handled by the `averageAction` strategy, which assigns them an action $g_a = keep$ if at least 50% of genes g in a candidate BGC G were also assigned an action $g_a = keep$. For the other two strategies, `neighborWeight` and `dryIslands`, weights w are computed for each gene g in a candidate BGC G as follows:

$$w = \sum_{i=0}^{n} h_i \qquad h_i = \begin{cases} \beta & \text{if backbone,} \\ \lambda & \text{if other annotation,} \\ \sigma & \text{otherwise.} \end{cases}$$

The `neighborWeight` strategy handles presence of annotations in neighboring genes: if a k number of surrounding neighbors of a given gene g holds a weight $\sum_{i=0}^{k} w_i > 1$, then the gene weight $g_w = 1$ and the gene action $g_a = keep$. The `dryIslands` strategy handles absence of annotations in contiguous neighboring genes: if $\sum_{i=0}^{j} g_w = 0$ for j sequential genes in G, then the gene action $g_a = skip$. For the evaluations described in Section 3, the following parameters were considered for the weights: $\beta = 2, \lambda = 1.5, \sigma = 0$, and an optimization of $j = [3, 4, 5]$ yield the most suitable parameter as $j = 3$.

3. Results

The reinforcement learning agent performance was evaluated in terms of *gene metrics* and *cluster metrics*, for which precision (P), recall (R), and F-measure (F-m) were computed. To compute *cluster metrics*, true positive matches were considered as candidate BGCs holding at least one match with gold standard genes. *Gene metrics* are computed based on matches between genes in candidate BGCs and gold standard BGCs, being true positives genes that are identical or similar, being candidate BGC and gold standard gene similarities obtained through local BLAST alignment (minimum $pident \geq 20, qcov \geq 10$). An average F-m is also computed between both *gene* and *cluster metrics* F-m.

Table 1. Distribution of *A. niger* BGC components in dataset

	Training		**Test**	
Component type	**BGCs**	**non-BGCs**	**gold BGCs**	**non-gold BGCs**
Backbones	17.0%	2.0%	15.9%	2.2%
Tailoring enzymes	30.5%	7.8%	9.9%	11.9%
Transcription factors	4.8%	2.1%	5.9%	4.3%
Transporters	5.6%	2.8%	7.4%	4.6%
Non-component domains	44.7%	46.93%	49.3%	58.9%
No domains	14.6%	41.15%	15.5%	23.2%
Total # genes	2833	1781	624	11239

BGC components distribution in datasets An analysis was performed to understand the distribution of BGC components associated to a role in the training and test datasets. The components distribution is shown in Table 1, which demonstrates that components associated to a role

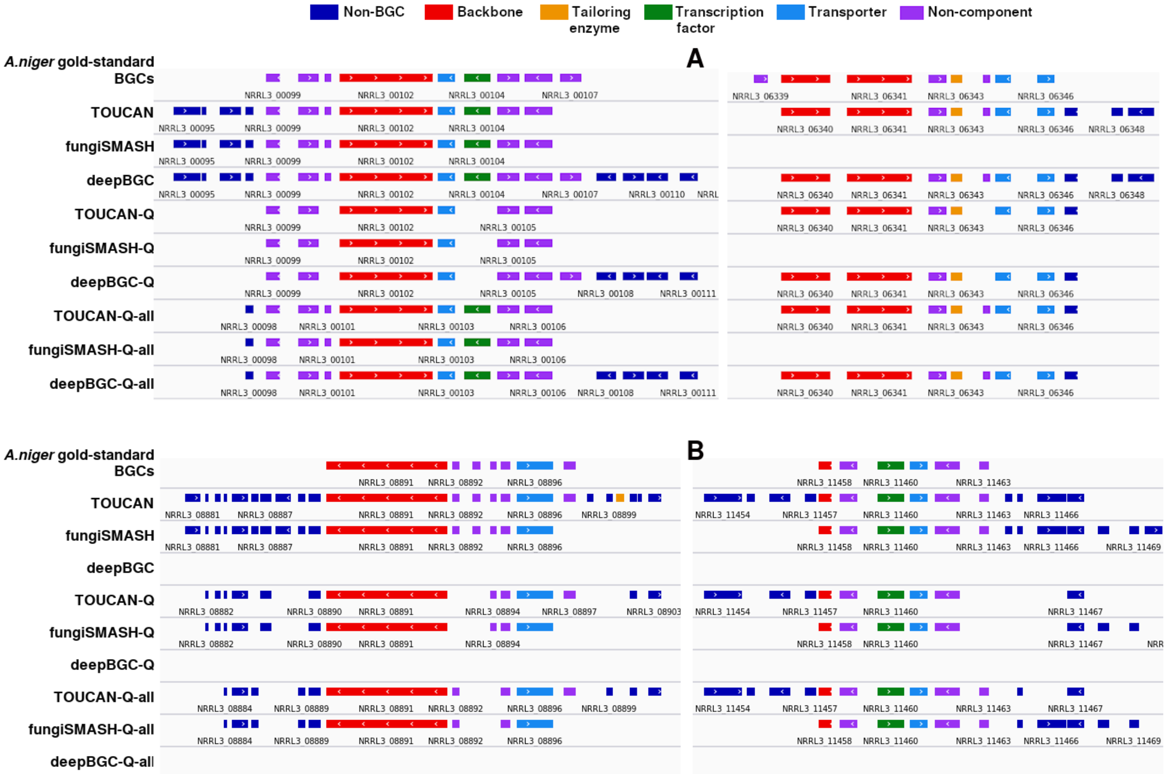

Figure 2. Comparison of *A. niger* gold-standard and candidate BGCs (non-BGC genes in dark blue). (**A**) Candidate BGCs which the agent correctly skipped most non-BGC genes. (**B**) Candidate BGCs which the agent kept most non-BGC genes – likely due to ambiguous protein domains.

are mostly concentrated in BGCs and gold standard BGCs, as opposed to in non-BGCs and non-gold standard BGCs. More than half of non-gold standard BGCs encode protein domains that are not associated to any component role.

Reinforcement learning enhances BGC predictions Candidate BGC predictions obtained with fungiSMASH, fungiSMASH combined with CASSIS (referred to as fungiSMASH/C), DeepBGC, and TOUCAN were processed by the proposed reinforcement learning approach. Prior to processing candidate BGCs, the reinforcement learning agent parameters were optimized, being learning rate $\alpha = 0.01$, discount-rate factor $\gamma = 0.01$, exploration-exploitation probability $\epsilon = 0.01$, and $keepSkip = 0.5$ the values yielding the smallest average penalty over 500 episodes.

Table 2 shows the results obtained for candidate BGCs outputted directly by each tool (TOUCAN, fungiSMASH, fungiSMASH/C, DeepBGC); candidate BGCs processed by the proposed reinforcement learning approach (TOUCAN-Q, fungiSMASH-Q, fungiSMASH/C-Q, DeepBGC-Q); and candidate BGCs processed by the proposed approach combined with the functional annotation strategies (TOUCAN-Q-all, fungiSMASH-Q-all, fungiSMASH/C-Q-all, DeepBGC-Q-all), as described in Section 2. Results obtained with the proposed reinforcement learning approach improved gene precision, and consequently gene F-m in candidate BGCs from all tools, yielding a performance increase of 14%, 15.4%, 15.2%, and 18.7% for TOUCAN-Q-all, fungiSMASH-Q-all, fungiSMASH/C-Q-all and DeepBGC-Q-all respectively.

Table 2. Performance on *A. niger* candidate BGCs from tools

	gene metrics			cluster metrics			average
model	P	R	F-m	P	R	F-m	F-m
TOUCAN	0.269	0.906	0.414	0.963	0.929	0.946	0.68
TOUCAN-Q	0.402	0.68	0.506	0.963	0.929	0.946	0.726
TOUCAN-Q-all	0.409	0.74	0.527	0.963	0.929	0.946	0.737
fungiSMASH	0.341	0.665	0.451	0.649	0.741	0.692	0.571
fungiSMASH-Q	0.521	0.516	0.519	1	0.741	0.851	0.685
fungiSMASH-Q-all	0.495	0.575	0.532	1	0.741	0.851	0.691
fungiSMASH/C	0.371	0.713	0.488	1	0.729	0.844	0.666
fungiSMASH/C-Q	0.523	0.508	0.515	1	0.729	0.844	0.680
fungiSMASH/C-Q-all	0.523	0.508	0.515	1	0.729	0.844	0.680
DeepBGC	0.351	0.481	0.406	0.732	0.612	0.667	0.536
DeepBGC-Q	0.574	0.42	0.485	1	0.612	0.759	0.622
DeepBGC-Q-all	0.538	0.46	0.496	1	0.612	0.759	0.627

Additionally, cluster metrics F-m for fungiSMASH-Q-all and DeepBGC-Q-all also improved considerably, yielding a performance increase of 15.9% and 9.2% respectively. Average F-m for all tools was improved with models that combined the reinforcement learning agent and functional annotations, a performance increase of 5.7%, 12%, 1.4%, and 9.1% for TOUCAN-Q-all, fungiSMASH-Q-all, fungiSMASH/C-Q-all and DeepBGC-Q-all respectively. The results indicate that overall the reinforcement learning agent was able to optimize BGC predictions without discarding regions correctly predicted by the three BGC discovery tools, therefore accurately targeting false positive regions. The improvement in results obtained by the agent when integrating functional annotations suggests that these can be relevant features to improve precision on predicted BGC regions.

Figure 2 shows examples comparing gold standard BGC components versus the components outputted by different models. In Figure 2-A the reinforcement learning agent improved BGC composition for all tools, and correctly skipped non-BGC genes (in dark blue). For clusters shown in Figure 2-B more non-BGC genes were mistakenly kept by the agent, which can lead to overpredicted BGC regions. Such mistakes can be the consequence of more complex cases, for which candidate BGC genes present ambiguous protein domains, which are present just as often in BGC and non-BGC instances in the datasets, and are usually found immediately close to true positive BGC components.

Since backbone enzymes are a vital BGC components, an analysis of the presence of backbone enzymes in candidate BGCs was performed. `TOUCAN` (92.9%), `fungiSMASH` (70.7%), `fungiSMASH/C` (69.7%) and `DeepBGC` (69.7%) percentages of candidate BGCs presenting a backbone enzyme remained the same even after applying the proposed reinforcement learning method. This is another indication that the agent is capable of targeting false positive regions and maintaining relevant BGC components in candidates.

Reproducible performance in *Aspergillus nidulans* The reproducibility of the proposed reinforcement learning approach was evaluated in the *A. nidulans* genome, considering a set of 72 gold standard BGCs (Drott et al., 2020). Pseudo-functional annotations were generated for the *A. nidulans* experiments since manually curated functional annotations were not available. The pseudo-annotations were assigned based on similar protein domain keyword descriptions matching *A. niger* previously annotated BGC components.

Table 3. Distribution of *A. nidulans* pseudo BGC components

	Training		**Test**	
Pseudo component type	**BGCs**	**non-BGCs**	**gold BGCs**	**non-gold BGCs**
Backbones	17.5%	2.13%	20%	2.45%
Tailoring enzymes	36%	3.70%	31.63%	4.5%
Transcription factors	4.83%	2.35%	5.92%	3.92%
Transporters	5.82%	3.65%	7.55%	5.2%
Non-component domains	33.15%	48.28%	35.3%	62.12%
No domains	14.6%	41.15%	12.65%	22.8%
Total # genes	2833	1781	490	10002

Table 3 shows the distribution of component pseudo-annotations found in the *A. nidulans* training and gold-standard data. Similarly to *A. niger*, candidate BGCs for *A. nidulans* were obtained from TOUCAN, fungiSMASH, fungiSMASH combined with CASSIS, and DeepBGC, and the predicted candidate BGCs were also pre-processed by the majority vote method as described in Section 2. The results obtained by the reinforcement learning agent are shown in Table 4. Our proposed method also improved gene precision in *A. nidulans* candidate BGCs from all tools, yielding an increase in gene precision of 13%, 15%, 16.6%, and 14.5%, and in average F-m of 5.2%, 6.4%, 3.6%, and 8.2% for `TOUCAN-Q-all`, `fungiSMASH-Q-all`, `fungiSMASH/C-Q-all` and `DeepBGC-Q-all`.

Table 4. Performance on *A. nidulans* candidate BGCs from tools

	gene metrics			**cluster metrics**			**average**
model	P	R	F-m	P	R	F-m	F-m
`TOUCAN`	0.272	0.681	0.389	1	0.685	0.813	0.601
`TOUCAN-Q`	0.441	0.591	0.505	1	0.681	0.810	0.657
`TOUCAN-Q-all`	0.402	0.646	0.495	1	0.681	0.810	0.653
`fungiSMASH`	0.319	0.727	0.443	0.817	0.795	0.806	0.624
`fungiSMASH-Q`	0.479	0.592	0.53	1	0.781	0.877	0.703
`fungiSMASH-Q-all`	0.469	0.605	0.529	1	0.736	0.848	0.688
`fungiSMASH/C`	0.318	0.762	0.449	1	0.792	0.884	0.666
`fungiSMASH/C-Q`	0.484	0.581	0.528	1	0.778	0.875	0.702
`fungiSMASH/C-Q-all`	0.484	0.581	0.528	1	0.778	0.875	0.702
`DeepBGC`	0.328	0.493	0.394	0.723	0.466	0.567	0.480
`DeepBGC-Q`	0.491	0.441	0.465	1	0.466	0.636	0.550
`DeepBGC-Q-all`	0.473	0.492	0.482	1	0.472	0.642	0.562

The results obtained for *A. nidulans* further demonstrate that the reinforcement learning agent is able to improve gene metrics without affecting cluster metrics, meaning that false positive regions are again properly targeted. Generally gene recall improved for the models relying on the pseudo-functional annotations, but at the same time a small decrease in gene precision was noticed. This is likely a consequence of a slight increase in false positive components considered due to the use of pseudo-annotations. However pseudo-components might be a helpful alternative when manually curated functional annotations are not available, or also in contexts which experts prefer to favor recall over precision.

4. Conclusion

Fungal secondary metabolites are a vital source of compounds that benefit human health. Identifying novel BGCs can potentially lead to novel natural products, and support drug discovery. The scarcity of openly available data on fungal BGCs, and the genomic diversity of secondary metabolite clusters in fungi turn the task of accurately defining BGC borders highly challenging. This work proposed a reinforcement learning approach combined with functional annotation strategies to support optimizing fungal candidate BGCs obtained with three state-of-the-art tools. The performance yielded by our proposed approach improved in 15% and 25% cluster and gene precision of BGC candidates, without affecting true positive predicted BGC regions. Combining reinforcement learning and functional annotation strategies yields the best average F-m performance in *A. niger*, and improved gene recall in *A. nidulans*. The results achieved in both fungal genomes evaluated are an indication of the approach generalization power and robustness.

Data availability The source code and datasets are publicly available at `https://github.com/bioinfoUQAM/RL-bgc-components` under the MIT software license.

References

Almeida, H., Tsang, A., and Diallo, A. B. Supporting supervised learning in fungal Biosynthetic Gene Cluster discovery: new benchmark datasets. In *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 1280–1287. IEEE, 2019. doi: 10.1109/BIBM47256.2019.8983041.

Almeida, H., Palys, S., Tsang, A., and Diallo, A. B. TOUCAN: a framework for fungal biosynthetic gene cluster discovery. *NAR Genomics and Bioinformatics*, 2(4), 11 2020. ISSN 2631-9268. doi: 10.1093/nargab/lqaa098.

Blin, K., Shaw, S., Kloosterman, A. M., Charlop-Powers, Z., van Wezel, G. P., Medema, M. H., and Weber, T. antiSMASH 6.0: improving cluster detection and comparison capabilities. *Nucleic Acids Research*, 2021.

Chavali, A. K. and Rhee, S. Y. Bioinformatics tools for the identification of gene clusters that biosynthesize specialized metabolites. *Briefings in Bioinformatics*, 19(5): 1022–1034, 04 2017. doi: 10.1093/bib/bbx020.

Drott, M., Bastos, R., Rokas, A., Ries, L., Gabaldón, T., Goldman, G., Keller, N., and Greco, C. Diversity of Secondary Metabolism in Aspergillus nidulans Clinical Isolates. *Msphere*, 5(2), 2020.

El-Gebali, S., Mistry, J., Bateman, A., Eddy, S. R., Luciani, A., Potter, S. C., Qureshi, M., Richardson, L. J., Salazar, G. A., Smart, A., et al. The Pfam protein families database in 2019. *Nucleic Acids Research*, 47(D1):D427–D432, 2019.

Evdokias, G., Semper, C., Mora-Ochomogo, M., Di Falco, M., Nguyen, T. T. M., Savchenko, A., Tsang, A., and Benoit-Gelber, I. Identification of a Novel Biosynthetic Gene Cluster in Aspergillus niger Using Comparative Genomics. *Journal of Fungi*, 7(5):374, 2021.

Hannigan, G. D., Prihoda, D., Palicka, A., Soukup, J., Klempir, O., Rampula, L., Durcak, J., Wurst, M., Kotowski, J., Chang, D., et al. A deep learning genome-mining strategy for biosynthetic gene cluster prediction. *Nucleic Acids Research*, 47:e110, 08 2019. ISSN 0305-1048.

Inglis, D. O., Binkley, J., Skrzypek, M. S., Arnaud, M. B., Cerqueira, G. C., Shah, P., Wymore, F., Wortman, J. R., and Sherlock, G. Comprehensive annotation of secondary metabolite biosynthetic genes and gene clusters of Aspergillus nidulans, A. fumigatus, A. niger and A. oryzae. *BMC Microbiology*, 13(91), 2013.

Kautsar, S. A., Blin, K., Shaw, S., Navarro-Muñoz, J. C., Terlouw, B. R., van der Hooft, J. J., Van Santen, J. A., Tracanna, V., Suarez Duran, H. G., Pascal Andreu, V., et al. MIBiG 2.0: a repository for biosynthetic gene clusters of known function. *Nucleic Acids Research*, 48 (D1):D454–D458, 2020.

Keller, N. P. Translating biosynthetic gene clusters into fungal armor and weaponry. *Nature Chemical Biology*, 11(9):671–677, 2015.

Keller, N. P. Fungal secondary metabolism: regulation, function and drug discovery. *Nature Reviews Microbiology*, 17(3):167–180, 2019.

Kjærbølling, I., Mortensen, U. H., Vesth, T., and Andersen, M. R. Strategies to establish the link between biosynthetic gene clusters and secondary metabolites. *Fungal Genetics and Biology*, 130:107–121, 2019.

Kjærbølling, I., Vesth, T., Frisvad, J. C., Nybo, J. L., Theobald, S., Kildgaard, S., Petersen, T. I., Kuo, A., Sato, A., Lyhne, E. K., et al. A comparative genomics study of 23 Aspergillus species from section Flavi. *Nature Communications*, 11(1106), 2020.

Kriventseva, E. V., Kuznetsov, D., Tegenfeldt, F., Manni, M., Dias, R., Simão, F. A., and Zdobnov, E. M. OrthoDB v10: sampling the diversity of animal, plant, fungal, protist, bacterial and viral genomes for evolutionary and functional annotations of orthologs. *Nucleic Acids Research*, 47(D1):D807–D811, 2018.

Watkins, C. J. and Dayan, P. Q-learning. *Machine Learning*, 8(3-4):279–292, 1992.