`clinker` & `clustermmap.js`: Automatic generation of gene cluster comparison figures

Cameron L.M. Gilchrist^{*1}, Yit-Heng Chooi^{*1}

¹School of Molecular Sciences, The University of Western Australia, 35 Stirling Hwy, Crawley, 6009

E-mail: cameron.gilchrist@research.uwa.edu.au, yitheng.chooi@uwa.edu.au

Abstract

Summary

Genes involved in biological pathways are often colocalised in gene clusters, the comparison of which can give valuable insights into their function and evolutionary history. However, comparison and visualisation of gene cluster homology is a tedious process, particularly when many clusters are being compared. Here, we present **clinker**, a Python based tool, and **clustermap.js**, a companion JavaScript visualisation library, which used together can automatically generate accurate, interactive, publication-quality gene cluster comparison figures directly from sequence files.

Availability and Implementation

Source code and documentation for **clinker** and **clustermap.js** is available on GitHub (github.com/gamcil/clinker and github.com/gamcil/clustermap.js, respectively) under the MIT license. **clinker** can be installed directly from the Python Package Index via **pip**.

Contact

E-mail: cameron.gilchrist@research.uwa.edu.au, yitheng.chooi@uwa.edu.au

Introduction

Genes involved in biological pathways are often colocalised in the genome in gene clusters, the study of which is important given their potential to encode traits such as the biosynthesis of bioactive small molecules, virulence and drug resistance. Gene clusters are typically conserved to some extent across taxa, and comparative analysis can reveal significant insights into their evolution and potential differences in the pathways they encode. For instance, the conservation (or lack thereof) of gene clusters encoding the biosynthesis of secondary metabolites can guide the search for new drug compounds and bioactivities (Gilchrist *et al.*, 2018, Ziemert *et al.*, 2016).

Comparative analysis of large genomic datasets has become increasingly common in the post-genomics era, with visualisations of gene cluster conservation across species often featured prominently in such work (de Vries *et al.*, 2017, Nielsen *et al.*, 2017, Doroghazi and Metcalf, 2013). This is a tedious process, and as such has driven the development of numerous tools (e.g. EasyFig (Sullivan *et al.*, 2011), Artemis Comparison Tool (Carver *et al.*, 2005), GeneSpy (Garcia *et al.*, 2019), Gene Graphics (Harrison *et al.*, 2018)) and libraries (e.g. GenomeDiagram (Pritchard *et al.*, 2006), genoPlotR (Guy *et al.*, 2010)). However, these tools have drawbacks: most have limited interactivity, with clusters unable to be repositioned, reordered or resized without changing input files; visualisation options are inflexible, if not fixed; some require sequence comparison files to be generated externally using tools such as BLAST (Camacho *et al.*, 2009), complicating the process; and libraries require some level of programming knowledge, making them inaccessible to the average biologist. Thus in many cases, researchers choose either to forgo specialist tools entirely, creating figures manually in other software (e.g. Microsoft PowerPoint, Adobe Illustrator), or directly use the output generated by analysis tools such as MultiGeneBlast (Medema *et al.*, 2013), cblaster (Gilchrist, 2020), or antiSMASH (Blin *et al.*, 2019). This quickly becomes a daunting prospect when many clusters are to be compared. There is therefore a need for tools which produce clear, publication-quality visualisations, are more flexible and intuitive than existing tools, and require no significant effort on the part of the user other than providing sequence data.

Here we describe **clinker**, a Python-based gene cluster comparison pipeline, and **clustermmap.js**, a companion visualisation JavaScript library, which can automatically generate interactive, to-scale gene cluster comparison figures directly from sequence files.

Gene cluster alignments using clinker

The **clinker** workflow is detailed in Figure 1a. **clinker** accepts GenBank files as input; multi-record GenBank files are treated as gene clusters with multiple loci. Amino acid translations of genes in each cluster are extracted and aligned using the BioPython package (Cock *et al.*, 2009). By default, global alignments are performed using the BLOSUM62 substitution matrix, a gap open penalty of -10 and extension penalty of -0.5, however parameters

Figure 1: The **clinker** pipeline (a) and visualisation of the burnettramic acid biosynthetic gene cluster (Li *et al.*, 2019) and homologues generated by **clinker** using **clustermap.js** (b). The gene cluster comparison figure in (b) is presented exactly as was achieved within the **clustermap.js** visualisation, with only minor further editing of text labels done in external software (italicising species names). The static HTML file used to generate (b) is provided in the supplementary information.

can be changed when directly using the `clinker` Python API. Any alignments not reaching the user-defined sequence identity threshold are discarded.

Optimal ordering of clusters for visualisation is determined through hierarchical clustering. First, a cluster similarity score is calculated for every pair of input clusters. `clinker` uses a modified version of the formula implemented by Medema *et al.* (2013) in MultiGeneBlast, which incorporates both sequence homology and syntenic conservation. Briefly, the similarity of two clusters is given by $S = h + i \cdot s$, where h is the cumulative sequence identity of homologous sequences in each cluster, s is the total number of shared contiguous sequence pairs, and i is a weighting factor which determines the weight of synteny compared to sequence homology (0.5 by default). The resulting similarity matrix is then hierarchically clustered using the Ward variance minimization algorithm as implemented in the SciPy package (Virtanen *et al.*, 2020). The leaves of the dendrogram generated in this step are used as the default order of clusters in the `clustermap.js` visualisation.

Interactive visualisations using `clustermap.js`

Cluster alignments generated using `clinker` are visualised using `clustermap.js` (Figure 1b). Though designed in conjunction with `clinker`, `clustermap.js` is a standalone library and can take data generated elsewhere as long as it abides by the correct schema.

Clusters are drawn to scale based on a user-definable scaling factor (15 pixels per 1000 base pair by default). Clusters can be renamed by clicking their labels and reordered by dragging them, enabling comparisons between clusters outside of the computed optimal ordering.

`clustermap.js` is capable of displaying multi-locus clusters. For example, the biosynthetic gene cluster for the burnettramycin acids (Li *et al.*, 2019) is split over three contigs due to fragmented genome assembly, but is readily visualised by our tool (top-most cluster in Figure 1b). Hovering over a locus displays a grey box with handles at both ends; loci can be freely repositioned by clicking and dragging the box, inverted by double clicking it, or resized (hiding genes) by dragging the handles. Any changes in the visible area of specific loci is reflected by the coordinates in the cluster label. Genes are drawn as arrows, whose shape is easily altered using sliders for body height, tip height and length. The visualisation can be anchored around a specific gene by clicking on it: `clustermap.js` will identify the closest homologous genes in other clusters and automatically reposition them to align with the clicked gene. Gene labels can be resized, repositioned or hidden entirely.

Links are drawn between homologous genes on neighbouring clusters and are shaded based on sequence identity (0% white, 100% black). By default, all links above the identity threshold in `clinker` are drawn; this threshold can be raised within the `clustermap.js` visualisation to dynamically hide links. Alternatively, one can choose to show only the highest scoring links between clusters, which is useful when multiple similar genes exist within a

cluster, resulting in many overlapping links. Groups of homologous genes are established via single linkage of gene links. Each homology group is assigned a unique colour, which is used as the fill of both the genes in the group and its corresponding entry in the figure legend.

The scale bar, colour bar and homology group legend are embedded in the figure, and are automatically repositioned based on cluster and locus placement. The length of the scale bar can be changed by clicking on its text. Likewise, legend entries can be renamed by clicking their text.

Figures generated by `clinker` and `clustermmap.js` are easily exported as scalable vector graphics (SVG) files that can be further modified in vector image software.

`clinker` and `clustermmap.js` can serve as a useful extension of gene cluster search tools such as MultiGeneBlast (Medema *et al.*, 2013) or cblaster (Gilchrist, 2020).

In conclusion, `clinker` and `clustermmap.js` enable the easy creation of publication quality gene cluster comparison figures, and can help facilitate and accelerate large-scale comparative genomic analysis, visualisation and presentation of gene clusters.

Software Implementation and Availability

`clinker` is implemented in Python 3 and can be freely installed from the Python Package Index (pypi.org/project/clinker).

`clustermmap.js` is implemented in JavaScript and is built on top of the D3 visualisation library (Bostock *et al.*, 2011). Source code and documentation for both tools is freely available on GitHub (github.com/gamcil/clinker and github.com/gamcil/clustermmap.js).

Acknowledgements

C.L.M.G is supported by an Australian Government Research Training Project scholarship. Y-H.C is supported by an Australian Research Council Future Fellowship (FT160100233).

References

- Blin, K., Shaw, S., Steinke, K., Villebro, R., Ziemert, N., Lee, S. Y., Medema, M. H., and Weber, T. (2019). antiSMASH 5.0: Updates to the secondary metabolite genome mining pipeline. *Nucleic Acids Res*, **47**(W1), W81–W87.
- Bostock, M., Ogievetsky, V., and Heer, J. (2011). D3 Data-Driven Documents. *IEEE T Vis Comput Gr*, **17**(12), 2301–2309.

109 Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K., and Madden, T. L. (2009).
110 BLAST+: Architecture and applications. *BMC Bioinformatics*, **10**(1), 421.

111 Carver, T. J., Rutherford, K. M., Berriman, M., Rajandream, M.-A., Barrell, B. G., and Parkhill, J. (2005). ACT:
112 The Artemis comparison tool. *Bioinformatics*, **21**(16), 3422–3423.

113 Cock, P. J. A., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff,
114 F., Wilczynski, B., and de Hoon, M. J. L. (2009). Biopython: Freely available Python tools for computational
115 molecular biology and bioinformatics. *Bioinformatics*, **25**(11), 1422–1423.

116 de Vries, R. P., Riley, R., Wiebenga, A., Aguilar-Osorio, G., Amillis, S., Uchima, C. A., Anderluh, G., Asadollahi,
117 M., Askin, M., Barry, K., Battaglia, E., Bayram, Ö., Benocci, T., Braus-Stromeier, S. A., Caldana, C., Cánovas,
118 D., Cerqueira, G. C., Chen, F., Chen, W., Choi, C., Clum, A., dos Santos, R. A. C., Damásio, A. R. d. L.,
119 Diallinas, G., Emri, T., Fekete, E., Flipphi, M., Freyberg, S., Gallo, A., Gournas, C., Habgood, R., Hainaut,
120 M., Harispe, M. L., Henrissat, B., Hildén, K. S., Hope, R., Hossain, A., Karabika, E., Karaffa, L., Karányi,
121 Z., Kraševac, N., Kuo, A., Kusch, H., LaButti, K., Lagendijk, E. L., Lapidus, A., Levasseur, A., Lindquist,
122 E., Lipzen, A., Logrieco, A. F., MacCabe, A., Mäkelä, M. R., Malavazi, I., Melin, P., Meyer, V., Mielnichuk,
123 N., Miskei, M., Molnár, Á. P., Mulé, G., Ngan, C. Y., Orejas, M., Orosz, E., Ouedraogo, J. P., Overkamp,
124 K. M., Park, H.-S., Perrone, G., Piumi, F., Punt, P. J., Ram, A. F. J., Ramón, A., Rauscher, S., Record, E.,
125 Riaño-Pachón, D. M., Robert, V., Röhrig, J., Ruller, R., Salamov, A., Salih, N. S., Samson, R. A., Sándor, E.,
126 Sanguinetti, M., Schütze, T., Sepčić, K., Shelest, E., Sherlock, G., Sophianopoulou, V., Squina, F. M., Sun,
127 H., Susca, A., Todd, R. B., Tsang, A., Unkles, S. E., van de Wiele, N., van Rossen-Uffink, D., Oliveira, J.
128 V. d. C., Vesth, T. C., Visser, J., Yu, J.-H., Zhou, M., Andersen, M. R., Archer, D. B., Baker, S. E., Benoit,
129 I., Brakhage, A. A., Braus, G. H., Fischer, R., Frisvad, J. C., Goldman, G. H., Houbroken, J., Oakley, B.,
130 Pócsi, I., Scazzocchio, C., Seiboth, B., vanKuyk, P. A., Wortman, J., Dyer, P. S., and Grigoriev, I. V. (2017).
131 Comparative genomics reveals high biological diversity and specific adaptations in the industrially and medically
132 important fungal genus *Aspergillus*. *Genome Biol*, **18**(1), 28.

133 Doroghazi, J. R. and Metcalf, W. W. (2013). Comparative genomics of actinomycetes with a focus on natural
134 product biosynthetic genes. *BMC Genomics*, **14**(1), 611.

135 Garcia, P. S., Jauffrit, F., Grangeasse, C., and Brochier-Armanet, C. (2019). GeneSpy, a user-friendly and flexible
136 genomic context visualizer. *Bioinformatics*, **35**(2), 329–331.

137 Gilchrist, C. L. M. (2020). gamcil/cblaster: cblaster v1.2.2.

138 Gilchrist, C. L. M., Li, H., and Chooi, Y.-H. (2018). Panning for gold in mould: Can we increase the odds for
139 fungal genome mining? *Org Biomol Chem*, **16**(10), 1620–1626.

140 Guy, L., Roat Kultima, J., and Andersson, S. G. E. (2010). genoPlotR: Comparative gene and genome visualization
141 in R. *Bioinformatics*, **26**(18), 2334–2335.

142 Harrison, K. J., de Crécy-Lagard, V., and Zallot, R. (2018). Gene Graphics: A genomic neighborhood data
143 visualization web application. *Bioinformatics*, **34**(8), 1406–1408.

144 Li, H., Gilchrist, C. L. M., Lacey, H. J., Crombie, A., Vuong, D., Pitt, J. I., Lacey, E., Chooi, Y.-H., and Piggott,
145 A. M. (2019). Discovery and Heterologous Biosynthesis of the Burnettramic Acids: Rare PKS-NRPS-Derived
146 Bolaamphiphilic Pyrrolizidinediones from an Australian Fungus, *Aspergillus burnettii*. *Org Lett*, **21**(5), 1287–
147 1291.

148 Medema, M. H., Takano, E., and Breitling, R. (2013). Detecting Sequence Homology at the Gene Cluster Level
149 with MultiGeneBlast. *Mol Biol Evol*, **30**(5), 1218–1223.

150 Nielsen, J. C., Grijsseels, S., Prigent, S., Ji, B., Dainat, J., Nielsen, K. F., Frisvad, J. C., Workman, M., and
151 Nielsen, J. (2017). Global analysis of biosynthetic gene clusters reveals vast potential of secondary metabolite
152 production in *Penicillium* species. *Nat Microbiol*, **2**(6), 17044.

153 Pritchard, L., White, J. A., Birch, P. R. J., and Toth, I. K. (2006). GenomeDiagram: A python package for the
154 visualization of large-scale genomic data. *Bioinformatics*, **22**(5), 616–617.

155 Sullivan, M. J., Petty, N. K., and Beatson, S. A. (2011). Easyfig: A genome comparison visualizer. *Bioinformatics*,
156 **27**(7), 1009–1010.

157 Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson,
158 P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson,
159 A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J.,
160 Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro,
161 A. H., Pedregosa, F., van Mulbregt, P., Vijaykumar, A., Bardelli, A. P., Rothberg, A., Hilboll, A., Kloeckner,
162 A., Scopatz, A., Lee, A., Rokem, A., Woods, C. N., Fulton, C., Masson, C., Häggström, C., Fitzgerald, C.,
163 Nicholson, D. A., Hagen, D. R., Pasechnik, D. V., Olivetti, E., Martin, E., Wieser, E., Silva, F., Lenders, F.,
164 Wilhelm, F., Young, G., Price, G. A., Ingold, G.-L., Allen, G. E., Lee, G. R., Audren, H., Probst, I., Dietrich,
165 J. P., Silterra, J., Webber, J. T., Slavič, J., Nothman, J., Buchner, J., Kulick, J., Schönberger, J. L., de Miranda
166 Cardoso, J. V., Reimer, J., Harrington, J., Rodríguez, J. L. C., Nunez-Iglesias, J., Kuczynski, J., Tritz, K.,
167 Thoma, M., Newville, M., Kümmerer, M., Bolingbroke, M., Tartre, M., Pak, M., Smith, N. J., Nowaczyk, N.,
168 Shebanov, N., Pavlyk, O., Brodtkorb, P. A., Lee, P., McGibbon, R. T., Feldbauer, R., Lewis, S., Tygier, S.,
169 Sievert, S., Vigna, S., Peterson, S., More, S., Pudlik, T., Oshima, T., Pingel, T. J., Robitaille, T. P., Spura, T.,

- 170 Jones, T. R., Cera, T., Leslie, T., Zito, T., Krauss, T., Upadhyay, U., Halchenko, Y. O., Vázquez-Baeza, Y.,
171 and SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nat*
172 *Methods*, **17**(3), 261–272.
- 173 Ziemert, N., Alanjary, M., and Weber, T. (2016). The evolution of genome mining in microbes – a review. *Nat*
174 *Prod Rep*, **33**(8), 988–1005.