



SOFTWARE TOOL ARTICLE

DNAContentViewer a BioJS component to visualise GC/AT Content [version 1; peer review: 2 approved with reservations]

Anil S. Thanki, Shabhonam Caim, Manuel Corpas , Robert P. Davey 

The Genome Analysis Centre, Norwich Research Park, Norwich, NR4 7UH, UK

V1 First published: 13 Feb 2014, 3:54
<https://doi.org/10.12688/f1000research.3-54.v1>
 Latest published: 13 Feb 2014, 3:54
<https://doi.org/10.12688/f1000research.3-54.v1>

Abstract

Summary: Compositional GC/AT content of DNA sequences is a useful feature in genome analysis. GC/AT content provides useful information about evolution, structure and function of genomes, giving clues about their biological function and organisation. We have developed *DNAContentViewer*, a BioJS component for visualisation of compositional GC/AT content in raw sequences. *DNAContentViewer* has been integrated in the BioJS project as part of the BioJS registry of components. *DNAContentViewer* requires a simple configuration and installation. Its design allows potential interactions with other components via predefined events.

Availability: <http://github.com/biojs/biojs>; doi: [10.5281/zenodo.7722](https://doi.org/10.5281/zenodo.7722).

Keywords

DNA



This article is included in the **Bioinformatics** gateway.




This article is included in the **BioJS** collection.

Open Peer Review

Approval Status ? ?

	1	2
version 1 13 Feb 2014	? view	? view

1. **Dan Schwartz**, University of Connecticut, Storrs, CT, USA
Joseph O'Shea, University of Connecticut, Storrs, CT, USA
2. **Matúš Kalaš** , University of Bergen, Bergen, Norway

Any reports and responses or comments on the article can be found at the end of the article.

Corresponding author: Anil S. Thanki (Anil.Thanki@tgac.ac.uk)

Competing interests: No competing interests were disclosed.

Grant information: AT, SC, MC and RD were supported by the UK Biotechnology and Biological Sciences Research Council (BBSRC) National Capability Grant (BB/J010375/1) at TGAC.

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2014 Thanki AS *et al.* This is an open access article distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. Data associated with the article are available under the terms of the [Creative Commons Zero "No rights reserved" data waiver](#) (CC0 1.0 Public domain dedication).

How to cite this article: Thanki AS, Caim S, Corpas M and Davey RP. *DNAContentViewer a BioJS component to visualise GC/AT Content* [version 1; peer review: 2 approved with reservations] F1000Research 2014, 3:54 <https://doi.org/10.12688/f1000research.3-54.v1>

First published: 13 Feb 2014, 3:54 <https://doi.org/10.12688/f1000research.3-54.v1>

Introduction

General compositional features of DNA sequences (e.g., GC/AT content) are important to understand the evolution, structure and function of genomes¹. Genes are often characterised by having a higher GC content than the rest of the genome. It has also been shown that the length of the coding sequence is directly proportional to a higher GC content². Stop codons, however, have a bias towards A and T nucleotides and are thus indicative of shorter gene sequences³.

GC content is usually expressed as a percentage value and sometimes as a ratio (GC ratio). The AT and GC content percentages and ratios can be measured in different ways (e.g., simple arithmetic or using a free online AT/GC calculator⁴).

To calculate the GC content percentage the following formula can be used⁵:

$$\frac{G+C}{A+T+G+C} \times 100$$

The AT content percentage can be calculated in a similar manner:

$$\frac{A+T}{A+T+G+C} \times 100$$

The *DNAContentViewer* component

DNAContentViewer has been developed as a part of the BioJS project⁶. It is available via GitHub or the BioJS registry⁷, a centralised repository of BioJS components hosted at the European Bioinformatics Institute (EBI).

DNAContentViewer uses the JavaScript library D3.js⁸ for generating graphs. It reads the DNA sequence in raw format and plots it as a Scalable Vector Graphics (SVG) object. To our knowledge, this

is the first client-side modular BioJS component that visualises GC/AT content directly from a raw sequence.

The way *DNAContentViewer* works is by calculating GC/AT content for dynamically generated slices of the sequence. This dynamic feature makes it fully scalable, working with sequences of any length. *DNAContentViewer* represents AT and GC content in separate graphs, allowing toggling of both to show and hide.

DNAContentViewer shows a percentage marker, which can be set on the left (by default), in the middle or on the right side of the graph. To adjust the visual aspect of the DNA content, *DNAContentViewer* contains simple controls for zooming and panning. *DNAContentViewer* can interact with other components via events and methods. Both events and methods are demonstrated through the component's page in the BioJS registry. Following the BioJS standard recommendations, documentation is available giving more details on how to implement the extension of the component via the registry.

The code below shows how to initialize *DNAContentViewer* with some minimal input. The name given to *YourOwnDivId* constitutes the component container identifier and *file* is defined absolute or relative path to the file containing the raw DNA sequence.

```
var instance = new Biojs.DNAContentViewer({
  target: "YourOwnDivId",
  dataSet: "file"
});
```

Example

An example is shown in [Figure 1](#), depicting GC/AT content from a genomic sequence in scaffold 1 of an assembly of the Chinese Hamster Ovary (CHO) genome. The GC/AT density data calculated from the raw sequence file are presented as bins, where the Y-axis indicates the percentage of GC/AT content for each bin. The overall AT content is

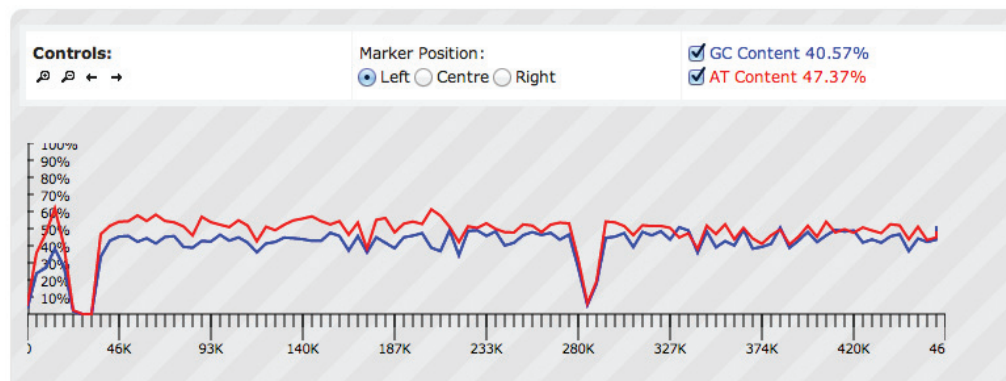


Figure 1. *DNAContentViewer* view of the Chinese Hamster Ovary (CHO) genome, scaffold 1. The top controls are designed to scroll sideways and to zoom; radio buttons can be used to set the position of the marker to the left (default), centre or to the right. Checkboxes can be used to toggle the GC/AT graph. Peaks show the GC/AT percentage density of dynamically generated bins. Differences can be observed around the 13K mark, with greater density of AT content, indicative of a stop codon and a poly A tail.

higher than the GC content. The 13K region shows a much higher AT content than average. This may be indicative of a stop codon and a poly A tail present in the region. The figure also shows a dramatic change in the GC/AT content in the 23K-32K and 284K regions, showing that in these locations GC/AT content does not sum up to 100% of the total base content. This is indicative of regions that contain unidentified nucleotides (X or N bases) instead of A, T, G or C.

Conclusion

The *DNAContentViewer* component provides the functionality to visualise GC/AT content of DNA sequences within the BioJS framework. *DNAContentViewer* can be easily integrated with other BioJS web components or extended to provide new functionality. Like any other BioJS component, *DNAContentViewer* requires little technical knowledge for its reutilisation.

Software availability

Zenodo: DNAContentViewer: a BioJS component to visualise DNA Content, doi: [10.5281/zenodo.77229](https://doi.org/10.5281/zenodo.77229).

GitHub: BioJS, <http://github.com/biojs/biojs>.

Author contributions

AT developed the code for *DNAContentViewer* and AT, MC and RD wrote the paper. SC designed the algorithm and wrote the initial version.

Competing interests

No competing interests were disclosed.

Grant information

AT, SC, MC and RD were supported by the UK Biotechnology and Biological Sciences Research Council (BBSRC) National Capability Grant (BB/J010375/1) at TGAC.

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Acknowledgements

We are grateful to all BioJS developers who have contributed their work under an open source license.

References

- Gao F, Zhang CT: **GC-Profile: a web-based tool for visualizing and analyzing the variation of GC content in genomic sequences.** *Nucleic Acids Res.* 2006; **34**(Web Server issue): W686–W691.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Pozzoli U, Menozzi G, Fumagalli M, et al.: **Both selective and neutral processes drive GC content evolution in the human genome.** *BMC Evol Biol.* 2008; **8**(1): 99.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Wuitschick JD, Karrer KM: **Analysis of genomic G + C content, codon usage, initiator codon context and translation termination sites in *Tetrahymena thermophila*.** *J Eukaryot Microbiol.* 1999; **46**(3): 239–247.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Wikipedia. GC-content.** [Online; accessed 27–Dec-2013].
[Reference Source](#)
- Madigan MT, Martinko JM, Parker J: **Brock, Biología de Los Microorganismos (Spanish Edition).** Pearson Educacion. 2004.
[Reference Source](#)
- Gómez J, García LJ, Salazar GA, et al.: **BioJS: an open source JavaScript framework for biological data visualization.** *Bioinformatics.* 2013; **29**(8): 1103–1104.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- BioJS: registry.** <http://www.ebi.ac.uk/Tools/biojs/registry/>, 2013.
[Reference Source](#)
- D3.js data-driven documents.** <http://d3js.org>, 2012.
[Reference Source](#)
- Thanki AS, Caim S, Corpas M, et al.: **DNAContentViewer: a BioJS component to visualise DNA content.** *Zenodo.* 2014.
[Data Source](#)

Open Peer Review

Current Peer Review Status: ? ?

Version 1

Reviewer Report 13 May 2014

<https://doi.org/10.5256/f1000research.3795.r4001>

© 2014 Kalaš M. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

? **Matúš Kalaš** 

Center for Computational Science, University of Bergen, Bergen, Norway

Here, the authors publish the BioJS component *DNAContentViewer* JavaScript widget for visualizing the GC and AT content of an input DNA sequence. This widget is part of the BioJS community effort which I strongly support.

Article

The title and the abstract are appropriate. The rest of the article is short, what I certainly appreciate. It has, however, both some major and some minor issues.

1. "raw sequence(s)" is an unnecessary expression, and possibly misleading.
2. A more appropriate reference than *Gao et al. 2008* for the importance of GC content should be included.
3. Reference number 4 is inappropriate, not because it is Wikipedia, but because it does not point to the mentioned calculator or to a proper Wikipedia article.
4. Reference number 5 is unnecessary and not particularly useful.
5. The window for the content calculations is not described.
6. The algorithm is not described.
7. The events and methods of *DNAContentViewer* are listed in the BioJS registry, but I am not convinced that they are demonstrated as is claimed.
8. I am not convinced that details on how to implement an extension of this component are given in its documentation.
9. "This is indicative of regions that contain unidentified nucleotides (X or N bases) instead of A, T, G or C." from a practical point of view is a somewhat ridiculous sentence. The purpose of the visualization is not to enable users to hypothesize about unidentified nucleotides which are obvious from the input sequence. Instead, the unidentified nucleotides should be visualized within the graph, in order to enable users to focus on the other, relevant indications.
10. "*DNAContentViewer* can be easily integrated with other BioJS web components or extended to provide new functionality" isn't really justified. The following sentence may also not be.
11. It would be nice if the *DNAContentViewer* were used in production, and in that case a

production usage example were mentioned in the article. Production usage would serve crucial feedback to the design of the component. On the other hand, I understand that despite its usefulness, production usage is not essential for designing a good component.

Widget

DNAContentViewer is a very simple JavaScript component for visualisation of GC and AT content of an input DNA sequence. I can imagine scenarios where its simplicity would be appreciated. In its current state, I would consider it a prototype, as I am not sure it is completely ready for publication. To consider the *DNAContentViewer* ready for production, I suggest addressing a substantial part of the following points:

1. Despite its simplicity, it is quite slow when zooming or panning.
2. Zooming in is not functional below max resolution but still requires all the clicks to zoom out.
3. Unnecessary animation when changing the y axis position, while zooming and panning is far from convenient.
4. It would be much more usable with mouse wheel to zoom (and centralize), and mouse drag to pan. See e.g. <http://mbostock.github.io/d3/talk/20111018/area-gradient.html>.
5. Reasonable positions on the x axis would be more useful, instead of random-looking ones like now, and not only in kbp range. They would ideally only extend and scroll when zooming or panning, but not change otherwise.
6. Maybe show the sequence when zoomed-in enough.
7. The 'nnnnn' bits of the sequence should be shown in the view.
8. An option for inverting the y axis upside-down for one of the GC or AT would be useful.
9. Selection of a subsequence by dragging mouse over a different part of the widget than for panning (e.g. at the x axis area while above the axis would pan - or vice versa - in any case indicated by cursor change).
10. Enable zooming in to see the subsequence selected with mouse dragging.
11. Showing the used sliding window size as a scale, and maybe even allowing customizing it if that would be useful.
12. The sharp edges of the graphs - or the sparsity of points with values if that is the case - look a bit suspicious to me. What are the 'bins' and where do they come from? And could they be shown in the visualization, e.g. like in <http://mbostock.github.io/d3/talk/20111018/area-gradient.html>?
13. Details of the input format are not documented (e.g. character case and allowed characters).
14. Enabling more sequence formats, including a JS string without the need of a file.
15. I don't understand the 'selection' options. Which selected text do they concern, one on the axes or in the "menu" (why is that useful to do)?
16. Example of integration with another BioJS widget, naturally for example BioJS [Sequence](#) or BioJS [FeatureViewer](#).
17. It would be nice to enable at least testing with another sequence, without a need for local installation of Apache etc.

BioJS registry

I don't know how involved the authors are in the design and the future evolution of the BioJS

registry and BioJS itself, and although this may thus not be a suitable place, I have a couple of general suggestions related to the BioJS registry:

- I haven't found an easier way to view the source code of the components, that would be without the need for copying & pasting pieces of the URL, or navigating through GitHub.
- I find pagination at <http://www.ebi.ac.uk/Tools/biojs/registry/components.html> both useless and inconvenient. In addition it didn't always work well for me with sorting by a column versus the pagination.
- Examples of integration of multiple components, such as the [Sequence](#) and [Protein3D](#) example that is only described in the spec - or the even fancier example at http://cdn.f1000r.com.s3.amazonaws.com/manuscripts/3850/22cef3a8-64b5-4189-8c1c-71b545883363_figure1.gif - would be both useful, and justifying the interoperability claims of BioJS. It would be awesome to have an integrated example for each BioJS component easily accessible from the registry.
- I'd love being able to test out at least such simple components as this [Biojs.DNAContentViewer](#) in my own HTML, without having to install Apache with PHP locally. Is that possible? If it isn't due to the BioJS itself, at least some interpreter like http://www.w3schools.com/js/tryit.asp?filename=tryjs_lib_jquery would be very nice to have.

Competing Interests: No competing interests were disclosed.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Reviewer Report 24 March 2014

<https://doi.org/10.5256/f1000research.3795.r4051>

© 2014 Schwartz D et al. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



Dan Schwartz

Department of Physiology and Neurobiology, University of Connecticut, Storrs, CT, USA

Joseph O'Shea

University of Connecticut, Storrs, CT, USA

Thanki *et al.* describe *DNAContentViewer*, a component of the BioJS collection aimed at visualizing GC and AT content in a string of nucleotides. While we are wholly supportive of the BioJS project, we have some important concerns with the *DNAContentViewer* manuscript/tool as it presently stands.

Manuscript

The manuscript starts with an introduction that is vague and difficult to read. The first sentence sets up the importance of GC content, but references an alternate web tool for visualizing GC content (a primary reference should be used). It appears that the authors took the remainder of

their introduction directly from an inaccurate and poorly worded Wikipedia entry (complete with the identical references).

Compare the Wikipedia entry for "GC-content":

"Within a long region of genomic sequence, genes are often characterised by having a higher GC-content in contrast to the background GC-content for the entire genome. Evidence of GC ratio with that of length of the coding region of a gene has shown that the length of the coding sequence is directly proportional to higher G+C content (Pozzoli et al., 2008). This has been pointed to the fact that the stop codon has a bias towards A and T nucleotides, and, thus, the shorter the sequence the higher the AT bias (Wuischick et al., 1999)".

to the *DNAContentViewer* introduction:

"Genes are often characterised by having a higher GC content than the rest of the genome. It has also been shown that the length of the coding sequence is directly proportional to a higher GC content (Pozzoli et al., 2008). Stop codons, however, have a bias towards A and T nucleotides and are thus indicative of shorter gene sequences (Wuischick et al., 1999)."

The authors then go on to cite Wikipedia itself in the following sentence:

"The AT and GC content percentages and ratios can be measured in different ways (e.g., simple arithmetic or using a free online AT/GC calculator (improper reference to Wikipedia "Talk" tab))."

Wikipedia version:

"the GC-content can be accurately calculated by simple arithmetic or by using the free online GC calculator (web link to <http://www.basic.northwestern.edu/biotools/oligocalc.html>)."

Finally, for reference 5 in the manuscript (the reference to the GC content formula), the authors again use the same reference as noted in the Wikipedia entry (Brock Biology of Microorganisms, 2003), but this time deciding to go with the Spanish Edition (Brock, Biología de Los Microorganismos, 2004).

Other concerns with the manuscript:

1. Despite the limited computation necessary for creating the visualization, the authors have omitted some key information that is essential to understanding how the tool works. For example, the manuscript does not describe how the window size for the calculation is determined. This is particularly important given that it is not a parameter of the existing tool. How was the bin size chosen? How will shorter sequences be affected (i.e., is there a recommended lower bound for use)? Why were bins used rather than a sliding average? While we were able to partially answer some of these questions by digging through the code base, this is important information for users and should be described in the manuscript.
2. The example given in Figure 1 suggests that the 13K mark increase in AT content "may be indicative of a stop codon and a poly A tail". First, it is impossible to discern 2-3 nucleotides in a bin size of 4,600 (the authors should note that the 3' UTR typically sits between the stop codon and poly(A) tail in eukaryotes). Second, is there a poly(A) tail in this region? The authors should either definitively answer this question, or preferably use a real example that highlights the utility of the tool in answering a real biological question (i.e., how does this visualization allow us to see something that is difficult to detect otherwise).

Code/Tool

The *DNAContentViewer* component code is overall well-organized and well-documented. We were able to get a working demo up and running on a local server in roughly 20 minutes. That said, several issues should be corrected:

1. If one accesses the demo page at <https://www.ebi.ac.uk/Tools/biojs/registry/Biojs.DNAContentViewer.html> via HTTPS (which seems to be the default if you get there via Google) the demo will not load due to D3.js being loaded via HTTP and therefore blocked on HTTPS pages.
2. The provided `<script>` and `<link>` tags in the installation instructions (<http://www.ebi.ac.uk/Tools/biojs/registry/Biojs.DNAContentViewer.html#>) contain relative URLs. Changing these to absolute URLs would make installation quicker and simpler.
3. The provided `<script>` tags include jQuery UI before including jQuery itself, causing a "ReferenceError: jQuery is not defined" error in the JavaScript console of both Chrome and Firefox. Although it seems this error does not prevent the component from working, there is no need for it to be there. Simply switching the order of the `<script>` tags removed this error.
4. When using the viewer, two errors are thrown in the console whenever a user clicks on the left control arrow. The component still appears to work normally despite these errors and the graph will still shift to the left and render (assuming the user is not already as far left as the graph can go).
5. At most zoom levels, panning left and right actually changes the resultant GC and AT lines (i.e., they do not stay fixed). This seems to be an issue with how the system chooses to bin the data upon panning, and should be corrected. Additionally, a more fluid panning and zooming capability would be preferable (e.g., panning by dragging and zooming with a slider).

Competing Interests: No competing interests were disclosed.

We confirm that we have read this submission and believe that we have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however we have significant reservations, as outlined above.

The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com

F1000Research