Unlocking scalable and efficient multiomic analysis of 5- and 6-base genomes

biomodal

Nicholas Harding, Michael Wilson, Jean Teyssandier, David Currie, Casper Lumby, William Stark, **Mark S. Hill**, Páidí Creed

1. Introduction

Analysing methylation data is challenging, many existing analysis tools are difficult to work with and do not scale well as the number of samples increases. This lack of scalability means that standard analyses, such as identifying differentially methylated regions (DMRs), or summarising methylation fractions over genomic regions, require substantial time and memory – typically necessitating large scale compute infrastructure (e.g. compute clusters, cloud).

duet multiomics solution evoC is a new sequencing technology, that simultaneously derives all four genetic bases without ambiguity in C or T calls, alongside distinguishing 5-methylcytosine and 5 hydroxymethylcytosine (**6-base** data) in a single read from a single DNA molecule [1] (**Figure 1**). The technology consists of pre-sequencing library prep and post-sequencing analysis pipeline, providing single-base resolution of genetics and epigenetics at high accuracy.

This expansion of biological signal that can be generated in a single sequencing experiment further increases the scale and complexity of downstream analysis, necessitating the development of more efficient analysis software. To address this challenge, we present ***modality***, a fast and scalable array-based python package for the analysis of 5 and 6-base genomes (genetics, 5-mC and 5-hmC).

2. duet multiomics solution evoC

1 - Strand synthesis - creates a single molecule with a direct copy of the original information tethered together with a hairpin. The copy strand is without cytosine modifications initially, but importantly, utilises a high fidelity methyltransferase to specifically copy 5mC from the original to the copy strand.

2 - Paired-end read sequencing - generates sequence information after protection of cytosine modifications followed by deamination of all remaining cytosines to uracils, read as thymine in SBS.

3 - Read resolution - aligns original and copy strands to correctly call all 4 canonical bases in addition to 5mC and 5hmC.

4 - Aligned (4 base) reads with 5mC & 5hmC are tagged (**6-Base information**)

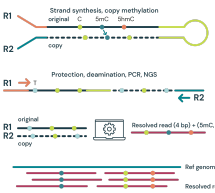
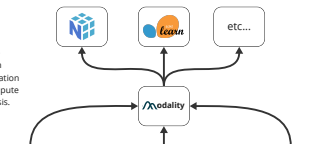


Figure 1 | duet multiomics solution evoC is a 6-base calling technology that reads all four canonical bases plus 5mC and 5hmC.

3. *modality* core concepts

modality is built around three core python packages: *zarr*, *xarray*, and *dask*. These are powerful modern data science packages which collectively allow *modality* to deal with larger-than-memory data arrays in an expressive and paralelised fashion. This foundation means that analyses that would previously require long run times and extensive compute infrastructure can now run quickly on one's laptop - speeding up iterative data analysis.

Figure 2 | a view of the modality ContigDataset
The core data structure used by *modality* is the ContigDataset. This contains arrays that represent methylation counts as well as accompanying arrays which encode the coordinates. This object also provides a set of easy-to-use and efficient methods for working with them. Each of the arrays are chunked *Dask* arrays allowing extremely efficient computation.



zarr storage backend
modality interacts with *zarr*, leveraging chunked and compressed sets of n-dimensional arrays, allowing analyses to efficiently scale to many samples (>100) even with very limited RAM.

Multi-dimensional Data Interaction
modality enables seemless interaction with datasets across multiple dimensions using *xarray*.

Parallel processing and memory efficiency
modality leverages the parallel processing capabilities of *dask* for efficient computation.

Efficient and rapid tooling for analysis

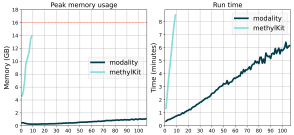


Figure 3 | Performance benchmarking of *modality*
One common operation that provides useful insight into the relationship between samples, in terms of their methylation profiles, is to compute a pairwise Pearson correlation matrix (see **Figure 4**). This is a computationally expensive operation so we used this to benchmark against an existing tool - *methylkit*. We used a machine with 16Gb of RAM to match what we might expect to be available on a typical laptop and computed the matrices for varying numbers of samples (>=1<=110) using both *modality* and *methylkit*. We were not able to generate the matrices using *methylkit* for datasets >10 samples, this would cause an out-of-memory error and crash. In contrast, we were able to generate these matrices for 110 samples in modality, using <1Gb RAM and running in ~6 minutes.

4. *modality* in action

As well as focussing on performance, *modality* was also built with the idea of providing a very user-friendly experience with a simple and intuitive API.

For example, calculating pairwise Pearson correlations between samples, for a given variable, and plotting the matrix can be done with the following lines of code:

```
dataset.plot_pearson_matrix(
  numerator="num_modc",
  denominator="num_total_c",
  min_coverage=10,
)
```

See **Figure 4** for result.

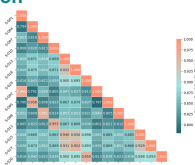


Figure 4 | Pearson matrix plot for Genome in a bottle data
The above plot is the output of the code block (**left**). It was produced using a duet +modC dataset of genome in a bottle (GIAB) samples and highlights the similarity between samples in terms of the fraction of modC calls relative to total C calls. This dataset is distributed with the *modality* package.

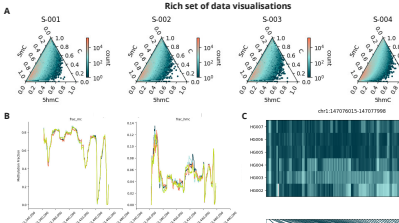


Figure 5 | Extensive array of visualisations and analyses for 5- and 6-base genomes
modality offers an extensive range of visualisations and analyses to help understand genome-wide methylation profiles. **A**, 6-base ternary plot showing density of data over bins of C:mC:hmC methylation fractions in mouse Ese14 cells. **B**, Methylation profile plots over genomic regions. **C**, Tile plot from differentially methylation regions (DMR) analysis, showing a genomic region with consistently different methylation fractions between two sets of samples.

5. Conclusion

To address the difficulties of analysing methylation data, we present *modality*, an efficient and scalable analysis package for 5- and 6-base genomes.

- The package is built on a core set of performant data science libraries and roots the user into a powerful ecosystem for data analysis in Python.
- *modality* has an intuitive API providing powerful analysis and visualisation methods.
- Moving forward, the underlying data structure used by *modality* is very extensible, allowing other data modalities to be incorporated and analysed alongside the methylation data modalities shown here.

biomodal

6. References

[1] Simultaneous sequencing of genetic and epigenetic bases in DNA, Füllgrabe and Gosal et al., Nature Biotechnology (2023) (duet multiomics solution technology paper)