Jumpstart your genomics pipelines with genomepy

This manuscript (permalink) was automatically generated from vanheeringen-lab/genomepy_manuscript@ad1f078 on November 20, 2020.

Authors

- **Siebren Frolich**
  0000-0001-6925-8446 · siebrenf
  Department of Molecular Developmental Biology, Radboud University

- **Maarten van der Sande**
  0000-0001-7803-1526 · Maarten-vd-Sande · MaartenvdSande
  Department of Molecular Developmental Biology, Radboud University

- **Simon van Heeringen**
  0000-0002-0411-3219 · simonvh · svheeringen
  Department of Molecular Developmental Biology, Radboud University

Abstract

Analyzing genomics data, including RNA-, ATAC- and ChIP-sequencing, requires multiple types of support data such as genome sequence and gene annotations. Many of these resources can be retrieved from different organizations, exist in multiple versions and may be generated by different methods. What datasets to use depends on the context of the research, such as collaboration partners, data reuse or dataset quality. As such, making an informed decision is essential. While many analysis pipelines are available, these mostly require manual downloading and management of genome-related resources. This can be tedious and error-prone and does not allow for full computational reproducibility.

Here we present genomepy, a command line tool and Python API that can navigate the genome databases of Ensembl, UCSC and NCBI. Genomepy can search and install genome sequences and gene annotation data from these providers in a consistent, reproducible and documented manner. The search function retrieves genomes related to the search term for all of the available providers, allowing the user to make an informed decision. The installation of the genome can be modified to obtain a soft-, hard- or unmasked version, or specific chromosomes or scaffolds filtered by regular expressions. Gene annotation data can be downloaded in addition to the genome sequence, which genomepy checks for compatibility with other bioinformatic tools. Finally, genomepy can automatically create indexes for commonly used aligners.

To summarize, genomepy is a straightforward tool to find, download and index genomes. It can be used to obtain genomes, gene annotations and additional support files in a consistent and automatic fashion. Genomepy is freely available at https://github.com/vanheeringen-lab/genomepy and can be installed using Bioconda and Pip.

The maze of genomes

Its big, its vague, and you just want to run your pipeline yesterday, right?

Table 1: Caption for this example table.

Database	Fun aspect
Ensembl	Updates infrequently, chromosome names don’t play nice
UCSC	GTFs labelled incorrectly
NCBI	Different pipelines, looks like Ensembl

What is genomepy

search, download, sensible defaults, reproducible, automatable. about those defaults...

How does genomepy fit in the ecosystem

Similar tools

its not like refgenie, but they could work nicely with eachother!

Within workflows

automatically download genomes, gene annotations, generate genome metadata and (splice-aware) genome indexes. Reference usage in seq2science.

How genomepy works

Install via conda, pip or git. Run via CLI or API.

CLI

basic steps.

API

same steps, but in API.

Acknowledgements

This manuscript was writtin with Manubot 1.

Introduction

Its big, its vague, and you just want to run your pipeline yesterday, right?

Table 2: Genome providers.

Database	Fun aspect
Ensembl	Generally seen as standard, updates infrequently, incompatible chromosome names
UCSC	multiple GTF formats, GTFs labelled incorrectly
NCBI	Different pipelines, looks like Ensembl, updates frequently

Related Work

- its not like refgenie, but they could work nicely with eachother!
- its missing in most workflows

there's a need for something that does the first step. genomepy fill that need.

genomepy

search, download, sensible defaults, reproducible, automatable. about those defaults...

Install via conda, pip or git.

basic steps in CLI

Repeat steps in API Extended steps, link to seq2science implementation?

Conclusions

- need for reproducibility
- standadization as the key to collaborations
- role for genomepy in this
- application in automated workflows
  - seq2science 2

Acknowledgements

We thank the Department of Molecular (Developmental) Biology, out github contributors, and issue posters for their patience, feedback and insight. We thank black, pytest, CodeCoverage and TravisCI for enduring our abuse and teaching us patience. And finally, we thank Manubot 1 for assisting with this manuscript.

Code availability

Genomepy can be installed using Bioconda and Pip. The code is available at https://github.com/vanheeringen-lab/genomepy.

References

1. **Open collaborative writing with Manubot**
   Daniel S. Himmelstein, Vincent Rubinetti, David R. Slochower, Dongbo Hu, Venkat S. Malladi, Casey S. Greene, Anthony Gitter
   PLOS Computational Biology (2019-06-24) https://doi.org/c7np
   DOI: 10.1371/journal.pcbi.1007128 · PMID: 31233491 · PMCID: PMC6611653

2. **seq2science**
   Maarten Van Der Sande, Siebren Frölich, Jos Smits, Simon Van Heeringen
   Zenodo (2020-11-05) https://doi.org/ghktg7
   DOI: 10.5281/zenodo.3921913