Machine Learning in Genomics: Containerised tutorials demonstrating best practises, pitfalls, and reproducibility

Sach Nehal

2024-08-08

Contents

About

Applied machine learning utilising vast amounts of data has aided in pattern identification, predictive analytics, and solving complex problems across a multitude of fields. Solving these complex problems within these fields, researchers would find differing answers to the following questions; **what machine learning techniques can we apply to the problem, how do we apply the techniques in the context of this field, and why do we need to apply them in this way?** In any case, applied machine learning requires an interdisciplinary understanding of computing techniques and the field in question.

The aim of this project is to provide you with **a set of reproducible, containerized tutorials that include all necessary data, code, and descriptions to replicate key results, along with demonstrations of common pitfalls, in the field of genomics**. It is designed for users with knowledge of machine learning but little or no background in biology as a process to learn about applying machine learning techniques in genomics.

Part I

Introduction

Chapter 1

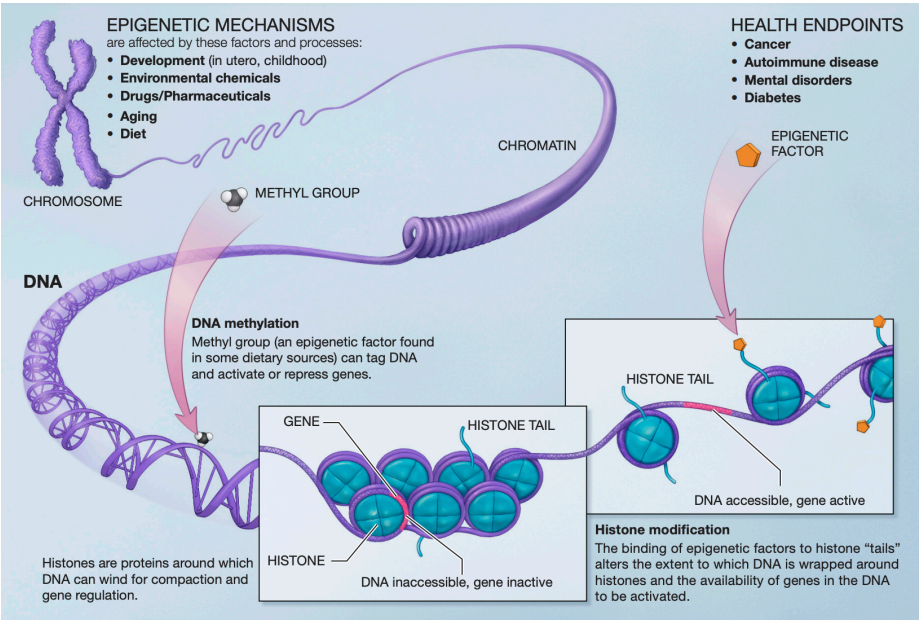
Epigenetic Data

1.1 What is epigenetic data?

As you may already know, typically all of the cells in your body contain the same DNA. How, then, do we have different cell types in our body? Your DNA contains a script that is able to produce the proteins required for each specific cell in your body. Which proteins, and subsequently which cells are made, depends on gene expression and regulation, i.e. "the way each cell deploys its genome."[1]

Epigenetic data arises from "the study of heritable and stable changes in gene expression that occur through alterations in the chromosome rather than in the DNA sequence."[2]

[1] Ralston and Shaw [2008]
[2] Al-Aboud et al. [2023]



commonfund.nih.gov

The image above shows quite simply the basics of genetic structures. Several more complex processes are involved during cell replication such as DNA transcription and translation in order to make proteins. A key takeaway in coming closer to understanding gene expression is that **Chromatin** is a complex structure made up of DNA wound around histone proteins, with some segments of DNA being accessible/inaccessible to further processes. **Euchromatin** refers to the accessible state, while **Heterochromatin** refers to a chromatin state in which DNA cannot be transcribed (inaccessible).[3] There are many different epigenetic modifications that affect chromatin accessibility.

Some common epigenetic modifications include:

1. **DNA Methylation**: Addition of methyl groups to DNA, affecting gene expression regulation[4].
2. **Histone Modifications**: Chemical changes to histone proteins that DNA wraps around, including acetylation, methylation, or phosphorylation. These changes influence chromatin structure and gene accessibility.[5]
3. **Chromatin Accessibility**: Regions of open chromatin that are accessible to transcription factors (special types of proteins that bind to DNA sequences and regulate gene expression) further dictate which regions of DNA can be expressed[6].

[3] Shahid et al. [2023]
[4] Al-Aboud et al. [2023]
[5] T. [2007]
[6] Kappelmann-Fenzl [2021]

In studying gene expression and epigenetic modifications, we aim to more closely understand biological mechanisms that regulate development, disease, and how cells respond to epigenetic factors.

1.1.1 What Does DNA Look Like?

As illustrated in the image above, DNA is structured as a double helix, with two complementary strands intertwined to form the characteristic helical shape. DNA consists of an extremely long sequence composed of four types of nucleotides: Adenine (A), Cytosine (C), Thymine (T), and Guanine (G).

According to the National Cancer Institute (USA), nucleotides within the DNA double helix form complementary pairs—Adenine pairs with Thymine, and Guanine pairs with Cytosine[7]. These pairs are commonly referred to as base pairs (bps). For example, if one strand of the double helix has the sequence "ATCGG", the complementary strand will have the sequence "TAGCC".

Genes are sequences of DNA located at specific positions on chromosomes and can vary in length. Each gene encodes information necessary for producing proteins or RNA molecules, which are essential for the structure, function, and regulation of an organism[8]. The complete set of genetic material in an organism is known as its genome.

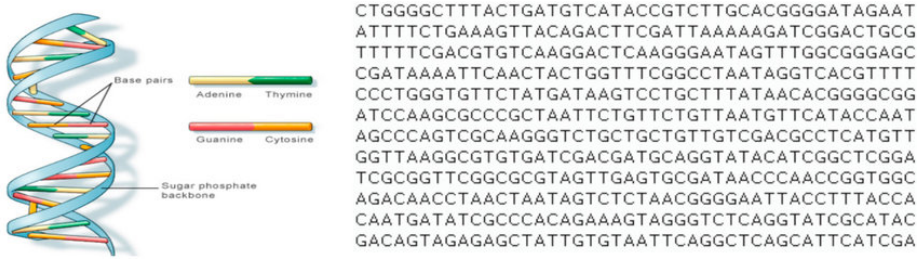


Image highlighting part of a dna sequence and base pairs.[9]

1.1.2 Common Epigenetic Sequencing Techniques:

1. ***ATAC-Seq*** (Assay for Transposase-Accessible Chromatin with Sequencing): o`Measures chromatin accessibility to identify open regions of the genome where transcription factors can bind.` o`Output: Peaks indicating accessible chromatin regions.`
2. ***ChIP-Seq*** (Chromatin Immunoprecipitation Sequencing): o`Used to identify DNA regions bound by specific proteins (e.g., transcription factors, histones with specific modifications).`

[7]Board
[8]Board
[9]Kanani and Padole [2020]

- Output: Peaks indicating binding sites or modification locations.

1.2 What does epigenetic data look like?

Epigenetic data can be represented in various forms, depending on the type of modification being studied and the methods used to gather the data. **ATAC-Seq** and **ChIP-Seq** are the common methods I will focus on, but there are others that may produce different forms of data, such as WGS (whole-genome sequencing) which produces nucleotide sequencing data, or Bisulfite conversion of DNA producing data on methylation levels across the genome.

1.2.1 Representing epigenetic data

Epigenetic data originates from sequencing methods such as ATAC-Seq or ChIP-Seq experiments. The initial experiments produce raw sequencing reads (fragments), which are then aligned to a reference genome. By aligning these sequences, we can aggregate the reads into regions where they 'pile up' to form peaks, indicating areas of significant biological activity or modification. This can be done per base across the genome, or per gene. Additionally, we could also examine mismatches where a read's base differs from the genome's base, and use them to identify SNPs (single nucleotide polymorphisms)[10]. This mismatch information can be recorded in a table showing the position, type of mismatch, and the number of reads supporting each mismatch.[11]

[10] Board
[11] Akalin [2020]

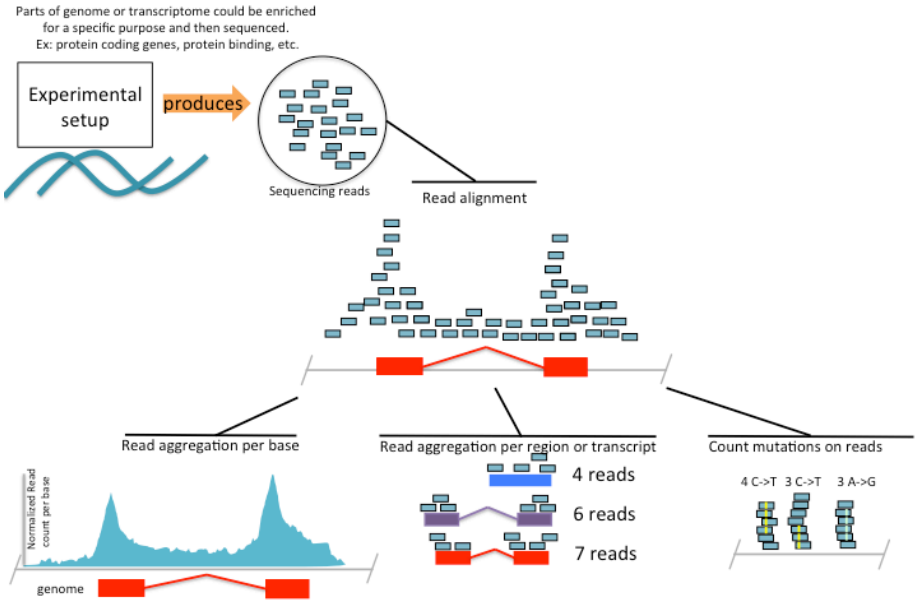


Image showing the sequencing pipeline from high-throughput sequencing methods[12].

1. ***Raw Sequence Reads:*** oThese are the basic output of sequencing experiments, such as those from ChIP-Seq or ATAC-Seq. oReads are processed and aligned to a reference genome before undergoing peak calling.

Lets look at what a few lines of raw sequence read data consists of: The data is taken from Encode Experiment ENCSR817LUF (chIP-Seq). The accession ID of the raw sequence read data is ENCFF397NRK. Genomic data comes in many file formats. This specific raw sequence read data is a compressed FASTQ file.

Note: The script I used involved streaming the data directly from a URL using the requests library. While files containing genomic data are generally quite large, for computational efficiency it is recommended that data be downloaded locally.

```
## ID: B091JABXX110402:1:2204:12975:184709
## Sequence: GTTAGGGTTAGGGTTAGGGTTAGGGTTAGGGTTAGG
## Quality Scores: [31, 30, 30, 32, 36, 37, 36, 32, 33, 32, 35, 36, 37, 33, 33, 34, 37, 37, 36, 3
##
## ID: B091JABXX110402:1:2205:18641:8399
## Sequence: GGTTAGGGTTAGGGTTAGGGTTAGGGTTAGGGTTAG
## Quality Scores: [22, 27, 31, 30, 30, 33, 31, 32, 31, 31, 24, 36, 37, 36, 33, 34, 33, 37, 37, 3
##
```

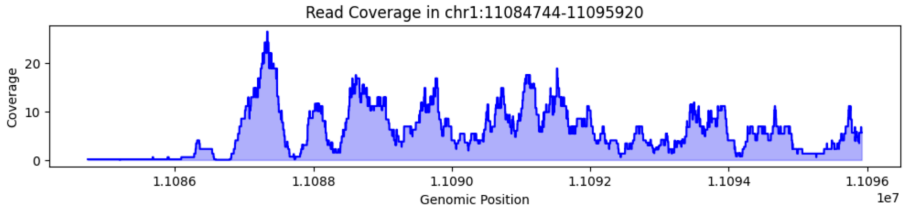
[12] Akalin [2020]

```
## ID: B091JABXX110402:1:1207:12202:100922
## Sequence: AGGGTTAGGGTTAGGGTTAGGGTTAGGGTTAGGGTT
## Quality Scores: [33, 33, 36, 37, 31, 34, 34, 36, 37, 36, 29, 33, 35, 36, 39, 37, 32
```

As you can see, each data entry is a DNA sequence (read). While I'm only showing the first three entries, for each experiment there are millions of sequencing reads. The quality scores indicate the confidence of each base call in the sequence. Higher scores suggest higher confidence. The scores are in Phred format, where a score of 20 corresponds to a 99% base call accuracy, 30 corresponds to 99.9%, 40 corresponds to 99.99%, and so on.[13] 2. ***Peak Calling:*** oA method used to identify regions in the genome where there is significant enrichment of sequencing reads. This indicates the presence of DNA-protein interactions (e.g., transcription factor binding sites or accessible chromatin regions). oPeaks represent areas where epigenetic marks or chromatin accessibility are concentrated.

A common peak calling algorithm is MACS2. Essentially, aligned sequencing reads are aggregated into regions where they 'pile up' to form peaks as a read count per base. The outputs typically include signal p-values or fold change over control values, with the continuous -log10(p-value) often averaged over bins (e.g., 25 base pairs). While MACS2 is traditionally used in ChIP-seq experiments, it is also applied to ATAC-seq to identify significant peaks and assess their enrichment. In ChIP-seq, broad peaks often represent histone modifications covering entire gene bodies, while narrow peaks are indicative of transcription factor binding sites[14]. In ATAC-seq, the peaks primarily reflect regions of open chromatin.

Representing Peaks: o ***P-value or Fold-change:*** P-value: Indicates the statistical significance of the peak, helping to distinguish true peaks from background noise. Fold-change: Represents the difference in read density between treated and control samples, indicating the strength of the signal.



This image shows the signal p-value coverage over a small region (11,176bps) in chromosome one from Encode Experiment ENCSR817LUF (The same chIP-Seq experiment we saw the raw sequence reads from). For further context, experiment ENCSR817LUF targets the H3K36me3 histone modification in brain tissue. The experiment aims to map the locations where the H3K36me3

[13] Green and Ewing
[14] Wilbanks and Facciotti [2010]

histone modification is present along the genome. Therefore the peaks represent regions of the genome where the the H3K36me3 histone modification is enriched compared to the background or control. The accession ID of the signal p-value data is ENCFF601VTB. Genomic data comes in many file formats. This specific signal p-value data is a bigWig file. o ***Types of Peaks:*** Categorical Peaks: Simple yes/no indication of a peak's presence. Continuous Peaks: More nuanced representation that includes the intensity or enrichment level of the peak, often visualized as a signal track. Thresholded/Pseudoreplicated Peaks: Usually categorical, these peaks are of high confidence regions from multiple replicates (experiments) or pseudoreplicates (artificial data splits), to ensure reliability and reproducibility.

Example Data Pipeline

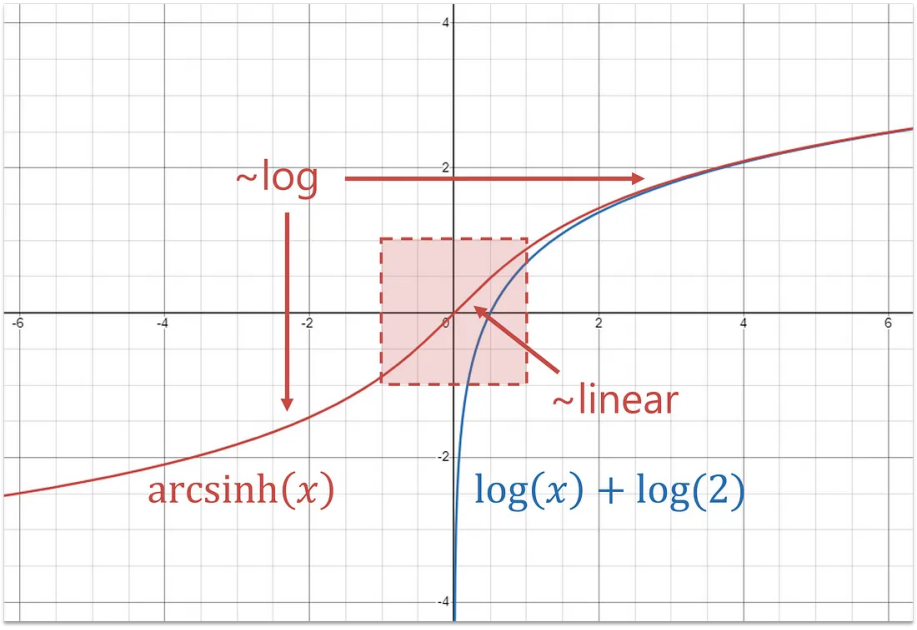
encodeproject.org

This is an example data pipeline from Encode Experiment ENCSR817LUF, the same chIP-Seq experiment we saw the raw sequence reads, and signal p-value coverage from. The yellow bubbles represent downloadable data sets of different types, while the blue boxes represent step types (e.g. peak calling). In the left column are multiple data sets of raw sequence reads, which then undergo data quality steps before being aligned (first blue box) to the reference human genome GRCh38 (denoted by ENCFF110MCL below the reads). The next steps include Peak calling (categorical peaks) and signal generation (continuous peaks) to produce the data we normally use in our machine learning models. This data pipeline process aids in normalisation, noise reduction, and dimensionality reduction of the data.

1.2.2 Transformations to stop extreme p-values

When utlising genomic data which incorporates p-values, it is important to consider and deal with extreme p-values. One way this is done is through using an Arcsinh-transformation (inverse hyperbolic sine). $\text{arsinh}(x) = \ln\left(x + \sqrt{x^2 + 1}\right)$

The arcsinh-transformation as a logarithmic function helps in reducing the significance of outliers and sequencing depth while maintaining variance by compressing the range of the data. This transformation can be used in the data preprocessing stage. The graph below visualises how the transformation works. While extreme values are transformed logarithmically, the smaller values are barely transformed as the function for smaller values is more linear in nature.



Plot of Arcsinh Transformation compared to a log function, made with Desmos available under CC BY-SA 4.0. Text, arrows, and box shape added to image.

1.3 Sources of epigenetic data

There are numerous public data banks which contain genomic datasets ready to be downloaded. Blueprint Blueprint's genomic datasets are focused on gene expression in healthy and diseased cells mostly relating to haematopoietic cells (cells which develop into different types of blood cells).

Roadmap The National Institute of Health's Roadmap Epigenomics Project contains sample datasets from multiple experiements as well as reference and mapping datasets.

Encode The Encode Project contains a large amount of publicly available genomic data easily filtered and downloaded. The genomic data used in this markdown book is sourced from Encode.

The largest genomic data bank is the UK Biobank, however they require that you apply for access to their datasets.

1.4 UCSC'S Genome Browser

The UCSC Genome Browser is a powerful and versatile tool that allows the visualisation and exploration of many sets of genomic data, especially bigWig files. It offers an extensive collection of genome assemblies, annotation tracks, and

functional data, enabling users to examine gene structures, regulatory elements, and genetic variations. With its user-friendly interface and customisable display options, the UCSC Genome Browser facilitates detailed genomic analyses and supports a wide range of applications in genomics and bioinformatics. Whether you're investigating gene functions, exploring genetic variants, or studying comparative genomics, the UCSC Genome Browser serves as an essential resource for understanding complex genomic information. It is also possible to load and visualise genomic data from other sources such as Encode. While the visualisations are extensive, as you can explore below, the browser can be quite overwhelming for first time users.

The following is an example of what the same chIP-Seq data targeting the H3K36me3 histone modification in brain tissue looks like using UCSC's Genome Browser. The pseudoreplicated peaks represent categorically, the significant locations along the genome where the H3K36me3 histone modification is present.

UCSC Genome Browser

The following is an example of ATAC-Seq data from an experiment on T-helper 17 cells (a type of immune system cell). Recall that the ATAC-Seq method aims to find chromatin regions that are accessible for transcription factor binding. The p-value and fold change graphs show continuous peaks, while the IDR thresholded peaks and pseudoreplicated peaks represent the significant locations of accessible chromatin along the genome.

UCSC Genome Browser

1.5 Handling bigWig files

BigWig files can be quite tricky to deal with. However, libraries such as pyBigWig enable easier access of data. In order to understand how to handle the data pre-processing stage, I have created a jupyter notebook tutorial. The tutorial begins using UCSC's programs to quickly understand the genomic data within BigWigs, before using the pyBigWig library to simply extract BigWig data.

The final part of the tutorial uses the pyBigWig library to load, filter, and split BigWig data into training, validation, and test sets. The data consists of signal p-values from ChIP-seq experiments, processed using the MACS2 tool, which outputs signals averaged over 25 base pair bins. We will re-average these signals to a resolution of 32 base pairs. Additionally, we will implement threshold-based filtering and consistent data splits to understand how to ready data for a model.

Tutorial 1: Dealing with bigWigs (interactive)

Tutorial 1: Dealing with bigWigs (nbviewer)

1.6 Data loaders and simplifying pre-processing

Data loaders are scripts/functions to load batches of data into your model. They are crucial in machine learning because they simplify how data is fed into models, making the whole process smoother and more efficient. This becomes especially important with the large datasets used in genomic studies, where managing and processing data manually would be cumbersome. By automating these tasks, data loaders help ensure that data is processed efficiently, allowing for faster and more effective model training. While there are existing github repositories with data loaders, such as "Kipoi Dataloader", and "Dataloader for BigWig files".[15], depending on the data used and model you build, they won't cover all of the use cases.

1.7 Dealing with missing data (oversampling, undersampling, weighting)

In genomics, class imbalance is a frequent challenge, often necessitating the use of statistical methods to validate the few positive instances amid vast amounts of data. This is particularly evident in tasks such as alignment queries, GWAS projects, and motif scanning, where conservative significance thresholds are essential to control false positives due to the low frequency of true positives across the genome. Researchers tend to address these imbalances by either oversampling the minority class, undersampling the majority class or by employing weighting.[16] Imbalanced data will be further discussed in the pitfalls section.

Methods for dealing with class imbalances:

- Scikit-learn's 'imbalance-learn' package (Oversampling, Undersampling and Weighting) "Imbalanced-learn (imported as imblearn) is an open source, MIT-licensed library relying on scikit-learn (imported as sklearn) and provides tools when dealing with classification with imbalanced classes."
- SMOTE (Oversampling) "a random example from the minority class is first chosen. Then k of the nearest neighbors for that example are found (typically k=5). A randomly selected neighbor is chosen and a synthetic example is created at a randomly selected point between the two examples in feature space."
- ADASYN (Oversampling) "similar to SMOTE but it generates different number of samples depending on an estimate of the local distribution of the class to be oversampled."

[15] Retel et al. [2024]
[16] Whalen et al. [2022]

Part II

Training models with DNA input

Chapter 2

Loss functions, and peak metrics

Chapter 3

Base pair averaging

Chapter 4

Training tricks

Chapter 5

Choosing which genomic regions to train on

Chapter 6

Effect of differences in sequencing depths

Chapter 7

Reproducibility of machine learning models

7.1 Seeding

7.2 Dashboarding

Chapter 8

Testing

Part III

Software libraries for model building

Chapter 9

gReLU

Chapter 10

Kipoi

Chapter 11

Weights and Biases

Part IV

ML pitfalls in genomics

Chapter 12

Pitfalls overview

12.1 Distributional differences

12.2 Dependent examples

12.3 Confounding

12.4 Leaky pre-processing

12.5 Unbalanced classes

In genomics, imbalance is a frequent challenge, often necessitating the use of statistical methods to validate the few positive instances amid vast amounts of data. In these cases, models can threaten to over-learn the majority class and under-learn the minority class. This is particularly evident in tasks such as alignment queries, GWAS projects, and motif scanning, where conservative significance thresholds are essential to control false positives due to the low frequency of true positives across the genome. The extensive size of the human genome exacerbates this issue, especially in problems related to non-coding variants, such as predicting chromatin states, gene expression, and disease status from sequence data.

Researchers address this imbalance by employing balancing algorithms that oversample the negative class and undersample the majority class. For instance, in training models to predict functional peaks from ChIP-seq or chromatin accessibility data, an approach might involve using all identified peaks along with a matching number of negative regions, thus effectively undersampling the majority class. For datasets with no such negative regions, researchers have to construct their own. While such imbalances are commonly discussed in classi-

fication, they also pose challenges in regression models predicting quantitative outcomes, where performance may be compromised in regions with sparse data, such as genomic areas or genes with low read counts in single-cell genomics studies. ## Balancing the proportion of peaks / no-peaks in validation sets

Part V

Model interpretability

Chapter 13

Creating and visualising a simple model

Chapter 14

TF mo-Disco

Part VI

Using existing models

Chapter 15

Using the gReLU model zoo

Chapter 16

Fine tuning of Enformer

Part VII

Predicting in novel cell types

Chapter 17

Incorporating ATAC-seq info

Chapter 18

Use of cell type averages

Part VIII

More complex models

Chapter 19

Training multi-headed models

Chapter 20

Training siamese twin models

Bibliography

Altuna Akalin. *Computational Genomics with R*. Github Pages, 2020. URL https://compgenomr.github.io/book/.

Nora M. Al-Aboud, Connor Tupper, and Ishwarlal Jialal. *Genetics, Epigenetic Mechanism*. National Library of Medicine, 2023. URL https://www.ncbi.nlm.nih.gov/books/NBK532999/#article-22137.r1.

PDQ® Cancer Genetics Editorial Board. *genetics-dictionary*. National Cancer Institute USA. URL https://www.cancer.gov/publications/dictionaries/genetics-dictionary.

Phil Green and Brent Ewing. *Phred - Quality Base Calling*. CodonCode Corporation. URL https://www.phrap.com/phred/#:~:text=Phred%20is%20a%20base%2Dcalling,%22)%20to%20each%20base%20call.

Pratik Kanani and Mamta Padole. *Improving Pattern Matching performance in Genome sequences using Run Length Encoding in Distributed Raspberry Pi Clustering Environment*. Procedia Computer Science, 2020. URL https://www.sciencedirect.com/science/article/pii/S1877050920311601?via%3Dihub. https://doi.org/10.1016/j.procs.2020.04.179.

Melanie Kappelmann-Fenzl. *Design and Analysis of Epigenetics and ChIP-Sequencing Data*. Springer, 2021. URL https://doi.org/10.1007/978-3-030-62490-3_12. ISBN 978-3-030-62490-3.

Amy Ralston and Kenna Shaw. *Gene Expression Regulates Cell Differentiation*. Nature Education, 2008. URL https://www.nature.com/scitable/topicpage/gene-expression-regulates-cell-differentiation-931/#:~:text=All%20of%20the%20cells%20within,each%20cell%20deploys%20its%20genome. Nature Education 1(1):127.

Joren Sebastian Retel, Andreas Poehlmann, Josh Chiou, Andreas Steffen, and Djork-Arné Clevert. A fast machine learning dataloader for epigenetic tracks from BigWig files. *Bioinformatics*, 40(1):btad767, January 2024. ISSN 1367-4811. doi: 10.1093/bioinformatics/btad767. URL https://doi.org/10.1093/bioinformatics/btad767.

Zainab Shahid, Brittany Simpson, Kathleen H. Miao, and Gurdeep Singh.

Genetics, Histone Code. StatPearls Publishing LLC, 2023. URL https://www.ncbi.nlm.nih.gov/books/NBK538477/. PMID: 30860712.

Kouzarides T. *Chromatin modifications and their function*. National Library of Medicine, 2007. URL https://doi.org/10.1016/j.cell.2007.02.005. PMID: 17320507.

Sean Whalen, Jacob Schreiber, William Noble, and Katherine Pollard. Navigating the pitfalls of applying machine learning in genomics. 2022. URL https://www.nature.com/articles/s41576-021-00434-9. https://doi.org/10.1038/s41576-021-00434-9.

Elizabeth Wilbanks and Marc Facciotti. *Evaluation of Algorithm Performance in ChIP-Seq Peak Detection*. Plos One Journal, 2010. URL https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0011471. https://doi.org/10.1371/journal.pone.0011471.