Assignment 5: Integrative epigenomics analysis

Due date: ~~Wednesday, 2/24 10am~~ Friday, 2/26 10am. Note: no late assignments will be accepted.

Introduction

CpG islands (CGIs) are important regulatory regions in the genome. What are their epigenetic profiles? Let's find out! We obtained several epigenomic datasets from the Roadmap Epigenome Project from a human brain germinal matrix sample:

- `BGM_WGBS.bed`: methylation data from whole genome bisulfite sequencing (WGBS)
- `BGM_MeDIP.bed`: methylation data from methylated DNA immunoprecipitation sequencing (MeDIP-seq)
- `BGM_MRE.bed`: methylation data from methyl-sensitive restriction enzyme sequencing (MRE-seq)

You are also provided with some annotation files:

- `CGI.bed`: locations of CGIs.
- `CpG.bed`: locations of CpGs.
- `refGene.bed`: locations of RefSeq genes.

Please see the appendix for a description of the files. All files are located in `/home/assignments/assignment5/`. All files are based on the hg19 reference genome assembly. To make things easier, we have decided to focus on chromosome 21. (The full datasets can be found in the Roadmap Epigenome Project flagship paper.)

From these data we want to obtain an epigenomic description of CGIs. Now you are ready to explore epigenomics data!

A few notes before you get started

- In this assignment, you will use the powerful command-line tool `bedtools` to help with some of the analysis. `bedtools` is already installed on the server. To print the `bedtools` help menu, type

```
$ bedtools --help
```

- All plots **must** have informative **axis labels, titles, etc**.
- In this assignment, you will be writing scripts that produce output files. Unlike previous assignments, you're going to run the same scripts on different inputs. In order to not overwrite the output file, your script **cannot hardcode** the output filename. Instead, you will use Python to dynamically name the output filenames based on the input file names. Here's some example code:

```
import os
basename = os.path.splitext(os.path.basename(<input_filename>))[0]
plotname = basename + "_methylation_distribution.png"
```

Part 1 — Analyzing DNA methylation data

Part 1.0

First, we are interested in answering some typical epigenomics questions.

Write a script called `analyze_WGBS_methylation.py` that takes a WGBS bed file and

1. Calculates the methylation level for each CpG using the formula:
$$CpG\ methylation\ level = \frac{\#\ C\ base\ calls}{\#\ C\ base\ calls + \#\ T\ base\ calls}$$
2. Writes a bed-like file of CpG methylation. The columns will be 1) chromosome, 2) start, 3) stop, and 4) methylation level. Do not output CpGs that have 0X coverage. Save the file as `<WGBS bed basename>_CpG_methylation.bed`.
3. Plots the distribution, e.g., with a histogram, of CpG methylation levels as `<WGBS bed basename>_methylation_distribution.png`.
4. Plots the distribution of read coverage for *all* CpGs for coverages between 0X and 100X as `<WGBS bed basename>_CpG_coverage_distribution.png`.
    a. ~~Note: `BGM_WGBS.bed` does **NOT** contain CpGs with 0X coverage. Use `bedtools intersect` with `CpG.bed` and `BGM_WGBS.bed` to find the CpGs with 0X coverage.~~
    b. `BGM_WGBS.bed` does contain CpGs with 0X so you do NOT have to use bedtools to find them.
5. Calculates and print the fraction of CpGs that have 0X coverage.

The usage of the script will be:

```
$ python3 analyze_WGBS_methylation.py <WGBS bed>
```

Run `analyze_WGBS_methylation.py` on the given WGBS data. Paste the command in your README. Copy your output files to your submissions directory.

Question 1

What does DNA methylation look like across chromosome 21?

Question 2

What does the CpG coverage look like across chromosome 21?

Question 2.1

What fraction of the CpGs have 0X coverage?

Part 1.1

Using `bedtools`, create a bed file with the average CpG methylation level in each CGI. This file should be in bed format (i.e., have the columns 1) chromosome, 2) start, 3) stop, 4) CGI name, and 5) average CpG methylation in CGI). Name your final file with the average CpG methylation

level in each CGI `WGBS_CGI_methylation.bed`. Paste the commands to generate this file in your README.

Hint: use the `bedtools intersect` and `bedtools groupby` subcommands.

Part 1.2

Write a script called `analyze_CGI_methylation.py` that takes the average CGI methylation bed file and plots the distribution of average CGI methylation levels. Save the plot as `<average CGI methylation bed basename>_distribution.png`.

The usage of the script will be:

```
$ python3 analyze_CGI_methylation.py <average CGI methylation bed>
```

Run `analyze_CGI_methylation.py` on `WGBS_CGI_methylation.bed`. Save the output figure as `WGBS_CGI_methylation_distribution.png`. Paste the command in your README.

Question 3

What does DNA methylation look like for CpGs in CGIs? How does it compare to all the CpGs on chromosome 21?

Part 1.3.0

Lastly, we want to explore the CpG methylation profiles in "promoter-CGIs" versus "non-promoter-CGIs."

Write a script called `generate_promoters.py` that takes a bed file of gene coordinates and creates a bed file of their promoters. The columns will be 1) chromosome, 2) start, 3) stop, 4) gene name, and 5) strand.

The usage of the script will be:

```
$ python3 generate_promoters.py <bed of gene coordinates>
```

Run `generate_promoters.py` on `refGene.bed`. Save the output file as `refGene_promoters.bed`. In your README, justify how you defined promoter and paste your command for creating this file.

Generate a bed file of promoter-CGIs called `promoter_CGI.bed` and a bed file of non-promoter-CGIs called `non_promoter_CGI.bed`. Promoter-CGIs are defined as CGIs that overlap with a promoter. In your README, justify your overlapping criteria and paste your commands for creating these files.

Hint: use `bedtools intersect`.

Calculate the average CpG methylation level for each promoter-CGI and non-promoter-CGI. Save these files as `average_promoter_CGI_methylation.bed` and `average_non_promoter_CGI_methylation.bed`, respectively. Paste your commands for creating these files in your README.

Hint: see your commands for part 1.1.

Run `analyze_CGI_methylation.py` on `average_promoter_CGI_methylation.bed` and `average_non_promoter_CGI_methylation.bed`. Save the output figures as `average_promoter_CGI_methylation.png` and `average_non_promoter_CGI_methylation.png`, respectively. Paste your commands for creating these files in your README.

Question 4
How do the DNA methylation profiles of promoter-CGIs and non-promoter-CGIs differ?

Part 1.3.1

Use this modified nuc count script (`/home/assignment5/nuc_count_multisequence_fasta.py`) to calculate the frequency of CpGs in promoter-CGI and non-promoter-CGIs fastas. In your README, paste the CpG frequency for both the promoter-CGIs and non-promoter-CGIs and your commands to calculate the frequencies.

Hint: use `bedtools getfasta` to convert a bed file to a fasta file. Use this chromosome 21 fasta file: `/home/assignments/assignment5/hg19_chr21.fa`.

Question 5
What is a possible biological explanation for the difference in CpG frequencies? Interpret your results from parts 1.4.0 and 1.4.1: what are the "simple rules" for describing regulation by DNA methylation in promoters?

Part 2 — Comparing CGI MeDIP-seq, MRE-seq, and WGBS methylation level.

The script `bed_reads_RKPM.pl` has already been written for you. This script takes a bed file of features, e.g., CGIs, and a bed file of reads and calculates the RKPM methylation score. The RPKM methylation score is defined as:

$$RPKM\ methylation\ score = \frac{\frac{\#\ of\ reads\ that\ overlap\ feature}{size\ of\ feature\ in\ kb}}{\#\ of\ mapped\ reads\ in\ millions}$$

The output is a bed-like file:

Column	Description	Example
0	Chromosome	chr21
1	Start coordinate of feature (0-based, inclusive)	9437272
2	End coordinate of feature (0-based, exclusive)	9439473
3	RPK (reads per kb) methylation score	2.2717

4	RPKM methylation score	12.8020

Usage:
```
$ perl bed_reads_RPKM.pl <feature bed> <reads bed> > <RPKM methylation levels>
```
Note: this is a script written in the Perl programming language, so it is invoked by typing "`perl <perl script>`."

Use `bed_reads_RKPM.pl` to calculate the CGI RPKM methylation scores using the MeDIP-seq and MRE-seq data. Save the output files as `MeDIP_CGI_RPKM.bed` and `MRE_CGI_RPKM.bed`, respectively. Paste the commands in your README.

Write a script called `compare_methylome_technologies.py` that

1. Creates the following scatter plots
    a. MeDIP-seq RPKM vs. MRE-seq RPKM. Save the plot as `<MeDIP-seq RPKM bed basename>_vs_<MRE-seq RPKM bed basename>.png`.
    b. MeDIP-seq RPKM vs. WGBS average DNA methylation level. Save the plot as `<MeDIP-seq RPKM bed basename>_vs_<WGBS methylation level bed basename>.png`.
    c. MRE-seq RPKM vs. WGBS average DNA methylation level. Save the plot as `<MRE-seq RPKM bed basename>_vs_<WGBS methylation level bed basename>.png`.
2. Calculate the correlation for each comparison. Print the correlation to stdout. Hint: use `scipy.stats`.

The usage of the script will be:
```
$ python3 compare_methylome_technologies.py <MeDIP-seq RPKM bed> <MRE-seq RPKM bed> <WGBS methylation level bed>
```

Run `compare_methylome_technologies.py` on `MeDIP_CGI_RPKM.bed`, `MRE_CGI_RPKM.bed`, and `WGBS_CGI_methylation.bed`. Save the output figures as `MeDIP_CGI_RPKM_vs_MRE_CGI_RPKM.png`, `MeDIP_CGI_RPKM_vs_WGBS_CGI_methylation.png`, and `MRE_CGI_RPKM_vs_WGBS_CGI_methylation.png`. Paste the command to generate the figures, the correlations, and justify which correlation statistic you used in your README.

Question 6
How do MeDIP-seq and methylation correlate? How do MRE-seq and methylation correlate? How do MeDIP-seq and MRE-seq correlate?

There is at least one outlier. In your README, list their locations and explain the potential cause(s) for the outlier(s). (Hint: look at the CGIs in a genome browser.) Explain why (or why not) the outlier(s) should be removed. If you removed them, recreate the scatter plots and recalculate correlations. Paste the correlations in your README. Save the output figures as `MeDIP_CGI_RPKM_vs_MRE_CGI_RPKM_outliers_removed.png`,

`MeDIP_CGI_RPKM_vs_WGBS_CGI_methylation_outliers_removed.png`, and `MRE_CGI_RPKM_vs_WGBS_CGI_methylation_outliers_removed.png`

What to turn in

- All scripts that you wrote
  - `analyze_WGBS_methylation.py`
  - `analyze_CGI_methylation.py`
  - `generate_promoters.py`
  - `compare_methylome_technologies.py`
- All output files
  - `BGM_WGBS_CpG_methylation.bed`
  - `BGM_WGBS_methylation_distribution.png`
  - `BGM_WGBS_CpG_coverage_distribution.png`
  - `WGBS_CGI_methylation.bed`
  - `WGBS_CGI_methylation_distribution.png`
  - `refGene_promoters.bed`
  - `promoter_CGI.bed`
  - `non_promoter_CGI.bed`
  - `average_promoter_CGI_methylation.bed`
  - `average_non_promoter_CGI_methylation.bed`
  - `average_promoter_CGI_methylation.png`
  - `average_non_promoter_CGI_methylation.png`
  - `MeDIP_CGI_RPKM.bed`
  - `MRE_CGI_RPKM.bed`
  - `MeDIP_CGI_RPKM_vs_MRE_CGI_RPKM.png` (and possibly `MeDIP_CGI_RPKM_vs_MRE_CGI_RPKM_outliers_removed.png`)
  - `MeDIP_CGI_RPKM_vs_WGBS_CGI_methylation.png` (and possibly `MeDIP_CGI_RPKM_vs_WGBS_CGI_methylation_outliers_removed.png`)
  - `MRE_CGI_RPKM_vs_WGBS_CGI_methylation.png` (and possibly `MRE_CGI_RPKM_vs_WGBS_CGI_methylation_outliers_removed.png`)
- Turn in a `README.txt` file with your commands and answers the above questions.

These files should be in your `assignment5/submission` folder.

Extra credit — Going the distance (again!)

Examine H3K4me3, the histone mark for promoters.
The H3K4me4 ChIP-seq dataset is located here:
`/home/assignments/assignment5/BGM_H3K4me3.bed`.

Use `bed_reads_RKPM.pl` to calculate the ChIP-seq RPKM score for promoter-CGIs and non-promoter-CGIs using the H3K4me4 ChIP-seq data. Save the output files as `H3K4me3_RPKM_promoter_CGI.bed` and `H3K4me3_RPKM_non_promoter_CGI.bed`, respectively. Paste the commands in your README.

Write a script called `analyze_H3K4me3_scores.py` that creates one figure with a boxplot of the H3K4me3 RPKM score distribution for promoter-CGIs and a boxplot of the H3K4me3 RPKM score distribution for non-promoter-CGIs. Save the plot as `<basename of first bed of RPKM scores>_and_<basename of second bed of RPKM scores>.png`.

The usage of the script will be:

```
$ python3 analyze_H3K4me3_scores.py <first bed of RPKM scores> <second bed of RPKM scores>
```

Run `analyze_H3K4me3_scores.py` on `H3K4me3_RPKM_promoter_CGI.bed` and `H3K4me3_RPKM_non_promoter_CGI.bed`. Save the output figure as `H3K4me3_RPKM_promoter_CGI_and_H3K4me3_RPKM_non_promoter_CGI.png`. Paste the commands in your README.

Question EC.1
How does the H3K4me3 signal differ in promoter-CGIs and non-promoter-CGIs?

Question EC.2
What are some better alternatives to model MeDIP-seq data and MRE-seq data instead of using RPKM? Explain.

Question EC.3
What would be a better way to compare H3K4me3 values instead of using boxplots? Explain.

What to to turn in for extra credit

- All scripts that you wrote
  - `analyze_H3K4me3_scores.py`
- All output files
  - `H3K4me3_RPKM_promoter_CGI.bed`
  - `H3K4me3_RPKM_non_promoter_CGI.bed`
  - `H3K4me3_RPKM_promoter_CGI_and_H3K4me3_RPKM_non_promoter_CGI.png`
- Add your commands and answers to the extra credit questions in the `README.txt`.

Changes

V2 (2/17 1pm): Due date extended
V3 (2/22 3pm): Removed incorrect statement about BGM_WGBS.bed not containing CpGs with 0X coverage. Fixed section reference in a hint on p. 3.

Appendix: Description of input files

This appendix describes the data files in `/home/assignments/assignment5/`. All coordinates are based on the hg19 reference genome. The data has been reduced to chromosome 21.

Fetal brain sample files:

- `BGM_WGBS.bed`: contains methylation data from WGBS in bed format. Each row contains the coordinate of a CpG with at least 1X coverage, the number of times the C was sequenced as "C" and the number of times the C was sequenced as "T". (Note: Reads that mapped to either of the two strands in the CpG were combined.) From these numbers, one can estimate the level of methylation for each CpG, as well as its coverage. The file is in a bed-variant format:

Column	Description	Example
0	Chromosome	chr21
1	Start coordinate of CpG (0-based, inclusive)	9411551
2	End coordinate of CpG (0-based, exclusive)	9411553
3	# of C basecalls for CpG	12
4	# of T basecalls for CpG	9

- `BGM_MeDIP.bed`: contains methylation data from MeDIP-seq in bed format. Each row contains the coordinate of a mapped MeDIP-seq read, the read, the mapping quality, and the strand the read aligned to:

Column	Description	Example
0	Chromosome	chr21
1	Start coordinate of MeDIP-seq read (0-based, inclusive)	9411609
2	End coordinate of MeDIP-seq read (0-based, exclusive)	9411759
3	Sequence of MeDIP-seq read	CAGAACTTGAGTGTGTAAGCTCCCAGAAAGAAGAGAAACACATTGAAGTGATTTCAACCTTCTCCACAGCCTTTC
4	Mapping quality	23
5	Strand that read mapped to	+

- `BGM_MRE.bed`: contains methylation data from MRE-seq in bed format. Each row contains the coordinate of a mapped MRE-seq read, the read, the mapping quality, and the strand the read aligned to.

Column	Description	Example
0	Chromosome	chr21
1	Start coordinate of MRE-seq read (0-based, inclusive)	9421490
2	End coordinate of MRE-seq read (0-based, exclusive)	9421562
3	Sequence of ChIP-seq read	CATCTCGGCTCACTGCGAGCTCAGCCTCCTGGCTTCGTGCCATTCTCCTGCCTCAGCCTCTCTAGTAGCGGG
4	Mapping quality	37
5	Strand that read mapped to	+

- `BGM_H3K4me3.bed` (for extra credit): contains H3K4me3 ChIP-seq data in bed format. Each row contains the coordinate of a mapped ChIP-seq read, the read, the mapping quality, and the strand the read aligned to:

Column	Description	Example
0	Chromosome	chr21
1	Start coordinate of ChIP-seq read (0-based, inclusive)	9411232
2	End coordinate of ChIP-seq read (0-based, exclusive)	9411382
3	Sequence of ChIP-seq read	GAGGTAGATCATCTTGGTCCAATCAGACTGAAATGCCTTGAGGCTAGATTTCAGTCTTTGTGGCAGGTGGGGGAA
4	Mapping quality	25
5	Strand that read mapped to	+

Annotation files:

- `CGI.bed`: contains locations of CGIs in bed format. Each row contains the coordinate and name of a CGI:

Column	Description	Example
0	Chromosome	chr21
1	Start coordinate of CGI (0-based, inclusive)	9437272
2	End coordinate of CGI (0-based, exclusive)	9439473
3	CGI name (derived from the # of CpGs in CGI) (**not** unique)	CpG: 285
4	Number of CpGs in the CGI	285

- `CpG.bed`: contains locations of CpGs in bed format. Each row contains the coordinate of a CpG:

Column	Description	Example
0	Chromosome	chr21
1	Start coordinate of CpG (0-based, inclusive)	9411551
2	End coordinate of CpG (0-based, exclusive)	9411553

- `refGene.bed`: contains locations of refSeq genes in bed format. Each row contains the coordinate of a gene, the gene name, the number of exons in the gene, and the strand of the gene:

Column	Description	Example
0	Chromosome	chr21
1	Start coordinate, i.e., leftmost coordinate, of gene (0-based, inclusive)	9825831
2	End coordinate, i.e., rightmost coordinate, of gene read (0-based, exclusive)	9826011
3	Gene name	MIR364
4	Number of exons in gene	1
5	Strand	+