Current to -Do

1. Make coverage plots for myotis samples
   1. Use them to determine coverage cutoffs
2. Deep dive into fastq files
   1. Use them to determine coverage cutoffs
3. Build ‘bridge’ script – python output to clock training
4. Build clock training script using myotis
   1. Add fake ages – point is to build up the data frame structure

Can we re-build Wilkinson’s clock using just our samples?

1. Cross reference exactly what sites of ours are those used in Wilkinson clock
   1. Only 62/169
2. Make metadata for our samples
   1. Clock training
   2. Clock using
3. Write script for clock using
4. Write script for clock training
5. Go through Manny’s steps to understand lab procedure
   1. We fragment the DNA (same as them)
   2. Repair the DNA ends and ligate a methylation-compatible universal Illumina adapter **(different)**
   3. Do the bisulfite conversion (same as them)
   4. PCR amplify the universal adapters with indexing primers for multiplexing **(different)**
   5. Size-select the libraries to a median size of 300-600bp **(different)**
   6. Pool libraries together based on molarity **(different)**
   7. Capture target loci using the pooled library **(different, but ideologically the same as their hybridization step)**
   8. Sequence the libraries **(different, similar to their scanning step)**.

Done

1. Go through this github, more details of using wilkinson’s method:
   1. <https://gitlab.com/WiDGeT_TrentU/graduate_theses/-/tree/master/Cossette/Epigenetics%20and%20island-mainland%20divergence%20in%20an%20insectivorous%20small%20mammal>
   2. https://onlinelibrary.wiley.com/doi/pdf/10.1111/mec.16735

Outstanding questions

1. Does bismark have some normalization routine comparable to sesame?
2. Do we need to ‘trim’ the clock to only the CpG sites we are looking at to use it for an age estimate?
   1. Yes, insofar as there is a filtering step
   2. Yes, if a clock site is not included in your dataset, you have to fill it with an ‘average methylation value from all other sites’
3. Are we only using CpG? Should we just ignore the CHG and CHH data we have from Manny?
   1. Use all three, since we have them
4. What filtering do we need to do to go from data output to clock input? (i.e. remove zeros, coverage statistics)
   1. Coverage statistics!
   2. From TIME-seq: since high-coverage rDNA loci have been shown to make better age prediction models, mouse rDNA methylation data was filtered to compromise only CpGs with high coverage (>200) in greater than 90% of samples at each CpG in the coverage matrix
   3. For other mouse and human clock CpG enrichment datasets, CpGs were filtered to have at least coverage 10 in 90%
5. How exactly is our sequencing method different from Wilkinsons?
   1. See Manny email and do more reading
6. How many known-age samples do we have? How many additional samples do we have?
   1. Need to look into age\_bat data for that
7. Are our P. rufus and E. dupreanum genomes high quality enough for post-sequencing processing?
8. How can we have less than 100% methylation? (bismark output)
   1. Still don’t know. But it’s a thing.
9. How can we have multiple sites per probe?
   1. CpG islands

GOAL: Create a methylation aging clock to use to reliably age Madagascar bats (*E. dupreanum, P. rufus, R. madagascariensis)* from wing punches instead of dentition. This project is half wet pipeline, half dry pipeline.

Background

Epigenetic clocks allow us to measure the age of tissue by looking at changes in DNA. This works using DNA methylation. Methyl groups attach to cytosines over time. You can study changes and patterns in DNA methylation over time to determine age.

Wilkinson et al 2021 – paper that uses bat dna methylation to age bats by building epigenetic clock

Input: methylation, confirmed age

Output: model

Methods: Creation of epigenetic clocks using penalized regression

We developed epigenetic clocks for bat wing tissue by regressing chronological age on all CpGs that map to at least one of the ten bat genomes. To improve linear fit we transformed chronological age to sqrt(age + 1). Penalized regression models were created in the R package glmnet[61](https://www.nature.com/articles/s41467-021-21900-2#ref-CR61). We investigated models produced by elastic net regression (alpha = 0.5). The optimal penalty parameters in all cases were determined automatically by using a tenfold internal cross-validation (cv.glmnet) on the training set. By definition, the alpha value for the elastic net regression was set to 0.5 (midpoint between Ridge and Lasso-type regression) and was not optimized for model performance. We performed two cross-validation schemes for arriving at unbiased estimates of the accuracy of the different DNAm based age estimators. One type consisted of leaving out a single sample (LOO) from the regression, predicting an age for that sample by regressing an elastic net on the methylation profiles of all other samples and iterating over all samples. We conducted LOO analyses using all samples from all species, using all samples from each species and using all samples from several species in the same genus. The second type consisted of leaving out a single species (LOSO) from the regression, thereby predicting the age of each sample using the data for all other species.

PIPELINE – Building the clock

1. DNA extraction from wing punches
   1. In: wing punch of known-age bats (from dentition)
   2. Out: DNA
2. Methylome sequencing
   1. In: DNA
   2. Out: methylation levels (DNAm, beta values) at targeted CPG sites
   3. Using custom Illumina methylation array
3. Python processing
   1. In: fastq, genome
   2. Out: % methylation level in bismark.cov files
4. Elastic net regression with known age data
   1. In: bismark.cov files, known age data, metadata?
   2. Function: some coverage filtering, training and testing
   3. Out: coefficients of clock and statistics

PIPELINE – Using the built clock

1. DNA extraction from wing punches
2. Methylome sequencing
3. Python Processing
4. Use with known

VALIDATION – WORK IN PROGRESS

1. Confirm that the Myotis samples can be processed A-Z
   1. Lab work troubleshooting – Manny/Sarah
      1. Sarah determined probe sequences based on Wilkinson et al—see /methylationbackground\_Guth
   2. There’s some python something happening here – what is it and who is going to do it in the future
   3. Dry pipeline
      1. Process raw reads (“Manny’s samples”)—modify into format to be used in a clock estimate based on Wilkinson et al
      2. Try to use a finished clock (Wilkinson et al clock) and see if I can get age estimates
         1. What is the clock, how do I use it
2. Make sure I know how to do the post sequencing python processing (see Python below)
   1. What comes out of sequencing?
      1. Fastq?
   2. What is our final product?
      1. 0-1 beta values (to be normalized). I’m not sure his pipeline gets us this far.
3. Test Manny’s pipeline on known age individuals to make sure we can get reasonable age estimates. Wilkinson is sending Manny samples—‘’the key part here is not so much the species as to get as wide and even a range of ages as possible for testing the predictive power of our approach”
   1. Wet: “for the Wilkinson samples, I’m going to be conservative and only do two Wilkinson samples for now, plus two other known-age controls from cell lines that Sarah extracted” – Manny
      1. Others will be completed in Berkeley
      2. These are already DNA, and we are going to learn how to do the methylation process
   2. Dry: predict age using wilkinson’s clock

DATASETS

**Manny’s samples:**

5 Myotis species with no known ages. “The goal for these runs was to show that the library prep, sequencing, and mapping/analysis pipeline works from A-Z. Since we didn’t want to burn through precious samples, we used as input either wing punches I could spare or cell lines that we had extracted DNA from already. The DNA from the cell lines was a control to make sure that the non-DNA carryover from wing punches didn’t inhibit or cause any issues downstream. Also, since these were the cell lines used to make the reference genomes for these species, it is the best mapping control you can ask for.

https://github.com/docmanny/methylationAging

Quick facts

5 species

* Myotis californicus
* Myotis evotis
* Myotis lucifugus
* Myotis thysanodes
* Myotis Volans

File structure

Output/methylation\_extracted

Table

Description automatically generated

**Our samples**

* infoAllSamp = Sample metadata (SampleID, Age, Tissue, Sex, Species, confidence in data, etc.)
* datAllSamp = Sample methylation data (SampleID, many many cg probes)
* anAge = Database of taxonomic classification & life history traits for species we have analyzed (Entire taxonomic name, Age of Sexual Maturity, Gestational Period, Maximum Lifespan, etc.)
  + Used if we use special age transformations in the predictive model (for bats, only used in defining "Relative Age")
* probe\_amin\_table = Table of probes x species saying which CpG probes map to something on a given species' genome (based on a species' genome annotation generated by another lab member)
  + Used if we pre-filter the CpG probes prior to building the clock (for bats, we did this for all the clocks in the paper)

anAge : https://genomics.senescence.info/species/index.html

DRY PIPELINE – PYTHON (3 inputs: trimmed.f)

1. Snakefile
   1. Trimmomatic
      1. In: trimmed.fq.gz, samples.tsv
      2. Out: \_trimmed\_R1\_unpaired.gq.gz, \_trimmed\_R1.fq.gz
   2. bismark
      1. In: bisulfite treated sequence reads (fastq), genome of interest
      2. Function: bisulfite mapping, methylation calling
      3. Out: many files summarizing methylation levels
   3. Fastqc
      1. In: samples.tsv, fastq.gz
      2. Out: fastqc.html, fastqc.zip
   4. Multiqc
      1. In: samples.tsv, fastqc.html
      2. Out: multiqc.html
2. Bismark to % methylation per chromosome.py
   1. In: bismark.cov files from step 2
   2. Function: simple counting/adding
   3. Out: file with chromosome/methylated/unmethylated/%methylated

DRY PIPELINE – NORMALIZATION?

I investigated Wilkinson et al to see how they processed their raw methylation reads before using them in their clock to estimate ages. There is substantial documentation that they first normalized them using an R package called SeSAme. My first objective is therefore to figure out how to use SeSame and to write a pipeline to normalize some of the Myotis samples.

Based on the information above from Manny’s email, I think that I need to work with a SUB file.

Checking to make sure that Manny didn’t normalize anything yet. But I don’t think that he did, because according to Wilkinson’s shared data, this is our target (https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSM4997095)



SESAME - This function is a simple wrapper of noob + nonlinear dye bias correction + pOOBAH masking

<https://rdrr.io/bioc/sesame/man/openSesame.html>

https://bioc.ism.ac.jp/packages/3.11/bioc/manuals/sesame/man/sesame.pdf

Trying to figure out what file type it needs. Steve’s tutorial notes a sample file .csv, but doesn’t show the example file. Digging into online documentation to see what I can find.

Arguments: SigSet(s), IDAT prefix(es), minfi GenomicRatioSet(s), or RGChannelSet(s)

Seems to be sample data available on the github:

<https://github.com/zwdzwd/sesameData/blob/master/vignettes/sesameData.Rmd>

\*\* looks like you need the data and a ‘sample-sheet’

SigSet

Text

Description automatically generated

https://bioconductor.org/packages/release//bioc/vignettes/sesame/inst/doc/sesame.html

IDATS – intensity data files!! The things that come directly out of Illumina.

\*I not have access to those files. I think Manny’s pipeline was intended to normalize of sorts, so maybe we are going to do that instead of SeSaMe anyways. I will wait for his response.

… so sesame is supposed to give normalized betas. What does that really mean?

Beta: continuous 0-1, representing the ratio of the intensity of the methylated bead type to the combined locus intensity.

I think we can just convert his beta values to 0-1. But I’m not sure about normalization. Also it might be easier to just use the Wilkinson method that goes directly from raw reads (I think)

GAGATAAAGAAAGGAATGTAAAACACTATGGAAACCAACCTCTGGAGACTTAATATTCTGATCAGCTCACCTCCGAACAT

DRY PIPELINE – USE CLOCK – Lu et al

A paper by Lu et al created an all mammal methylation clock, and had fairly explicit instructions on how to use the clock (I hope). I’m going to work my way through how to use their clock, then figure out how to apply that to the bat clock (since their documentation isn’t as great). They provide all code and data so it should be relatively straight forward (famous last words).

Main questions: what is the format of the input? How does the trained clock get taken in and used? What other metadata do I need?

Takes in: mydata.RDS (#includes species characters and CpGs), clocks as CSV

Step 1: convert age data from intput file to format for clocks (logs, relative, etc)

3 clocks:

Basic clock: log-transformed chronological age

Universal relative age clock: defines individual age relative to maximum spp lifespan

Universal log-linear transformed age clock: includes age at sex maturity and gestation time

Step 2: use clocks

Glmnet with input and trained clock

The trained clocks: lu and wilkinson

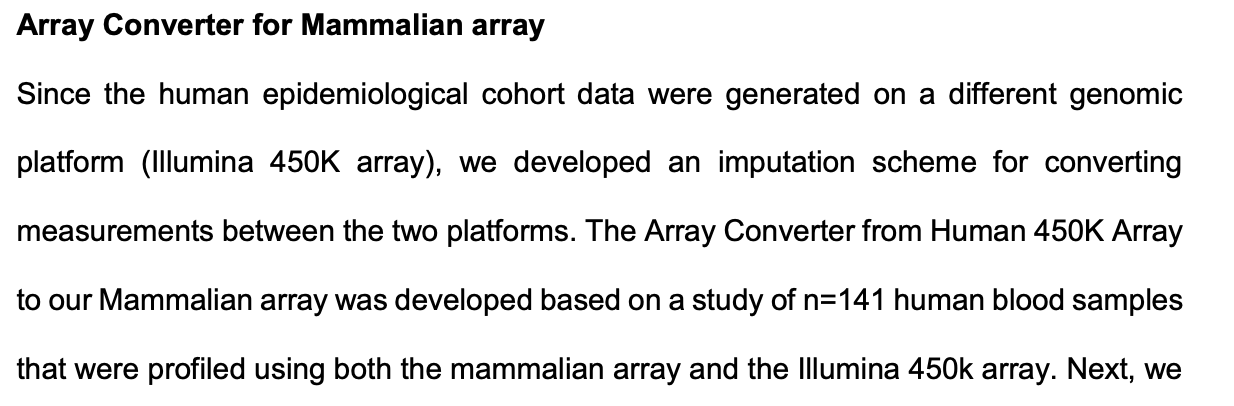
Table

Description automatically generated Table

Description automatically generated

Woo hoo they’re the same! Then I should be able to easily adapt Lu’s code to use the Wilkinson clock

RED FLAG – from Lu paper. We are also using Illumina, not the array. How do you make an array converter?



DRY PIPELINE – BUILD CLOCK – Wilkinson

General elastic net regression ML theory

Target variable: age

Independent variable: dna methylation levels

\*use methylation to predict age

1. Split into X and Y
   1. X = independent
   2. Y = target only
2. Split into train and test
   1. X\_train =
   2. X\_test =
   3. Y\_train =
   4. Y\_test =
3. Set up hyperparameters in elastic net
   1. Elastic net(alpha = 0.5, L1\_ratio = 0.5)
4. Fit model
   1. Enet.fit(X\_train, Y\_train)
5. Make predictions
   1. Y\_pred = Enet.pred(X\_test)
6. Evaluate
   1. MSE (y\_test, y\_pred)