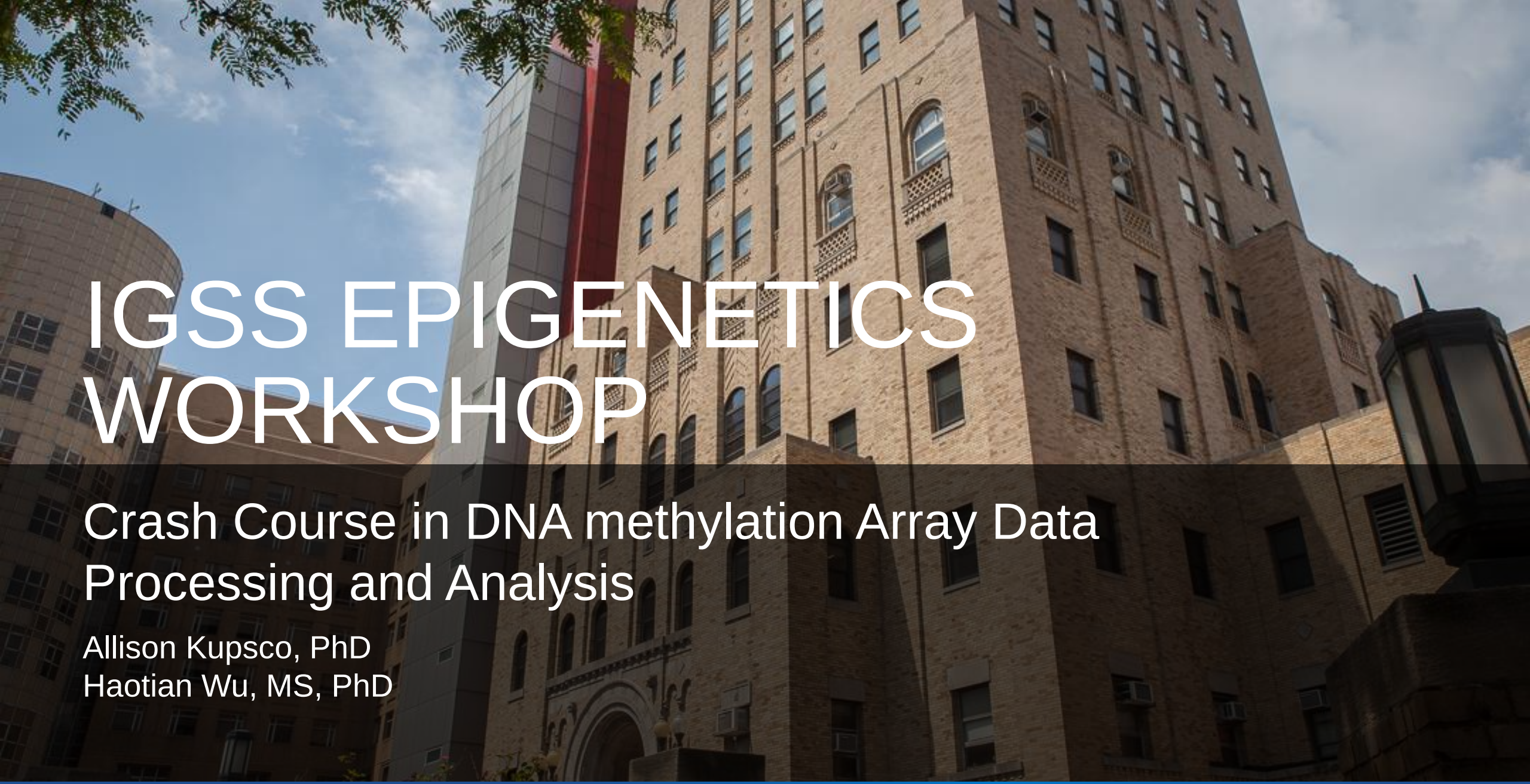


IGSS EPIGENETICS WORKSHOP

Crash Course in DNA methylation Array Data Processing and Analysis

Allison Kupsco, PhD
Haotian Wu, MS, PhD

The Presenters

Dr. Haotian (Howie) Wu

- Associate Research Scientist
- Environmental and Reproductive Epidemiologist
  - Maternal health and children's neurodevelopment
  - Molecular markers (DNAm, miRNA, metabolomics, mitochondrial) of reproduction and aging
- Big fan of the exposome
  - Reluctant fan of Canadian Rugby
- Emerging environmental pollutants and epigenetic biomarkers

Dr. Allison Kupsco

- Assistant Professor
- Environmental toxicologist and epidemiologist
  - Children's Metabolic Development
- Development of omics biomarkers for environmental health studies
  - Mitochondria
  - miRNAs
  - DNA methylation
  - New epigenetic markers

SHARP Trainings

Epigenetics Boot Camp: 2-days of concepts, techniques, and data analysis methods utilized in human epigenetics studies.

June 14-15, 2021 | Live-stream, virtual

Today's Training is a Condensed Version of a Condensed Workshop

- Hope to cover the basics
- Please feel free to reach out to us or talk to us after

Other Courses Offered:

- Microbiome, multi-omics, single cell seq, quantitative genomics, machine learning, exposome, mendelian randomization, and more!

Target Audience

- At least basic R experience
- Little to no experience processing DNAm array data
- Little to no experience with DNAm-wide modeling and subsequent applications (e.g. pathway or gene ontology)
- Interest in using DNAm data (and know what it is)

- ***Reminder – if you haven't, please start installing the packages (see our Rscripts)***

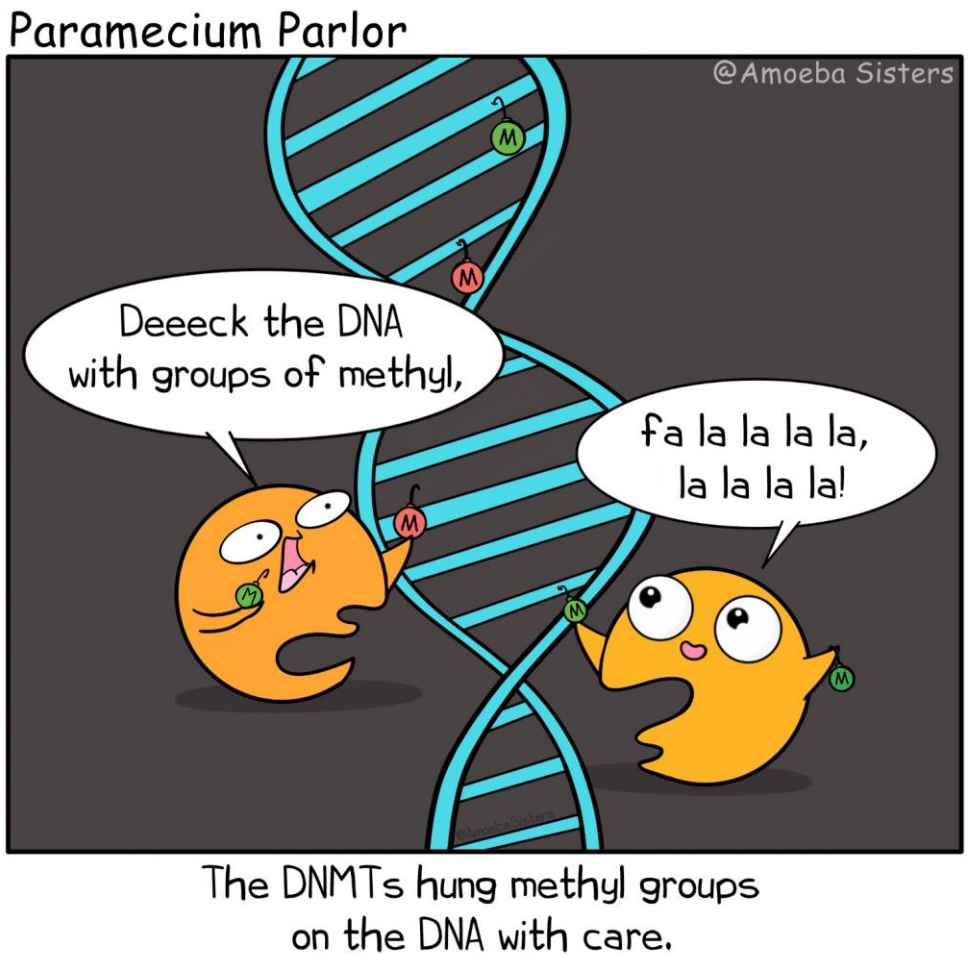
Content and Goals

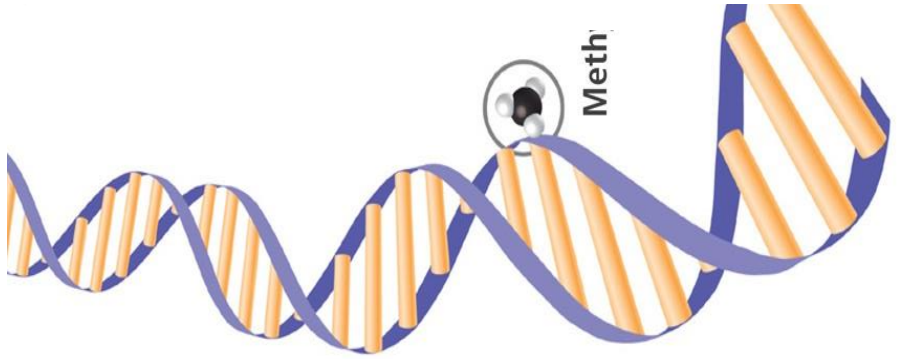
- Ability to
  - Process raw Illumina array data (code for 2 common pipelines)
  - Conduct common epigenome wide analyses
  - Conduct basic diagnostics of those models
  - Apply basic pathway analyses
  - Adapt the processed data for other downstream applications (e.g. methylation clocks)
- Understand the common challenges and decisions making processes

Introduction

DNA Methylation

- Genomic CpGs are most frequently studied
  - Do occur on other bases
  - Occurs on mitochondrial DNA as well
- Evolution of Technology
  - Global DNAm
  - Candidate Gene DNAm
  - -omic level assays
    - Arrays
    - Sequencing





Genetics and Epigenetics

Both screen for thousands to millions of loci:

- GWAS: Single nucleotide polymorphisms (SNPs)
- EWAS: CpG* sites

The EWAS/epigenetics field is relatively new

- Most methods are borrowed from genomics

Differences

- Genetics less susceptible to confounding and reverse causation
  - DNAm is (for example, by genetics!)
  - Mendelian Randomization works. DNAm randomization does not.

- Changes over time
  - GWAS: SNPs (almost) never change
  - EWAS: epigenetic marks change over time
    - Not just DNAm, but also histones, ncRNAs, etc.

Differences

- Type of Data
  - GWAS: SNP has fixed values
    - 0 (wt/wt); 1 (wt/var); 2 (var/var)
  - EWAS: measures are quantitative
    - Average % methylation
    - % cell with methylation
  - Epigenetic data can be both the independent and dependent variable

- Implications for interpretation
  - Common question: what is a “meaningful effect size”?

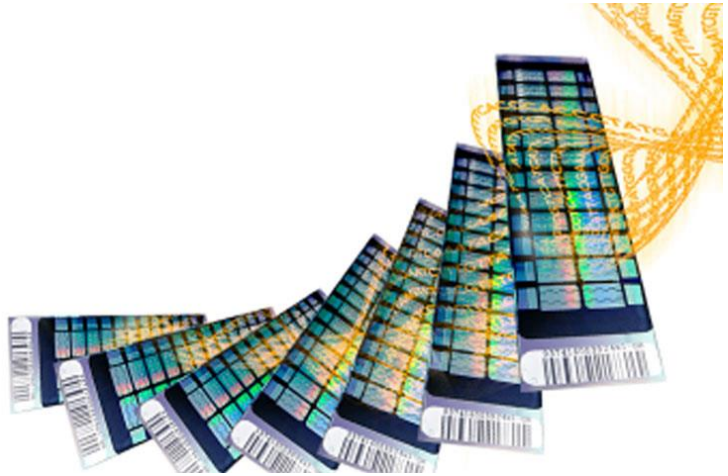
Tissue Specificity

- GWAS: SNPs are not tissue specific
- EWAS: epigenetic marks are tissue specific

- Need to be very cautious transporting results from commonly used biomatrices (e.g. blood, saliva) to actual targets of interest

- *Will address later*

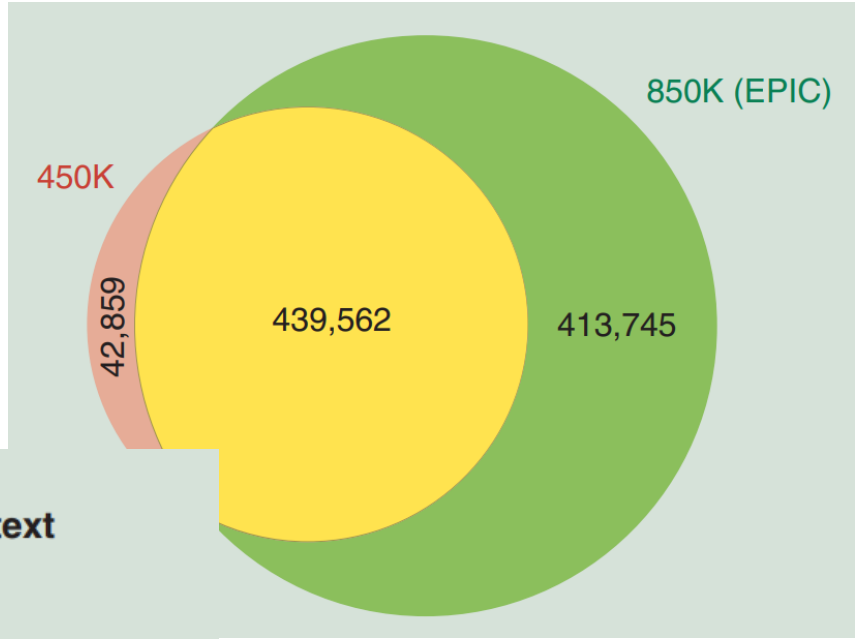
Platforms

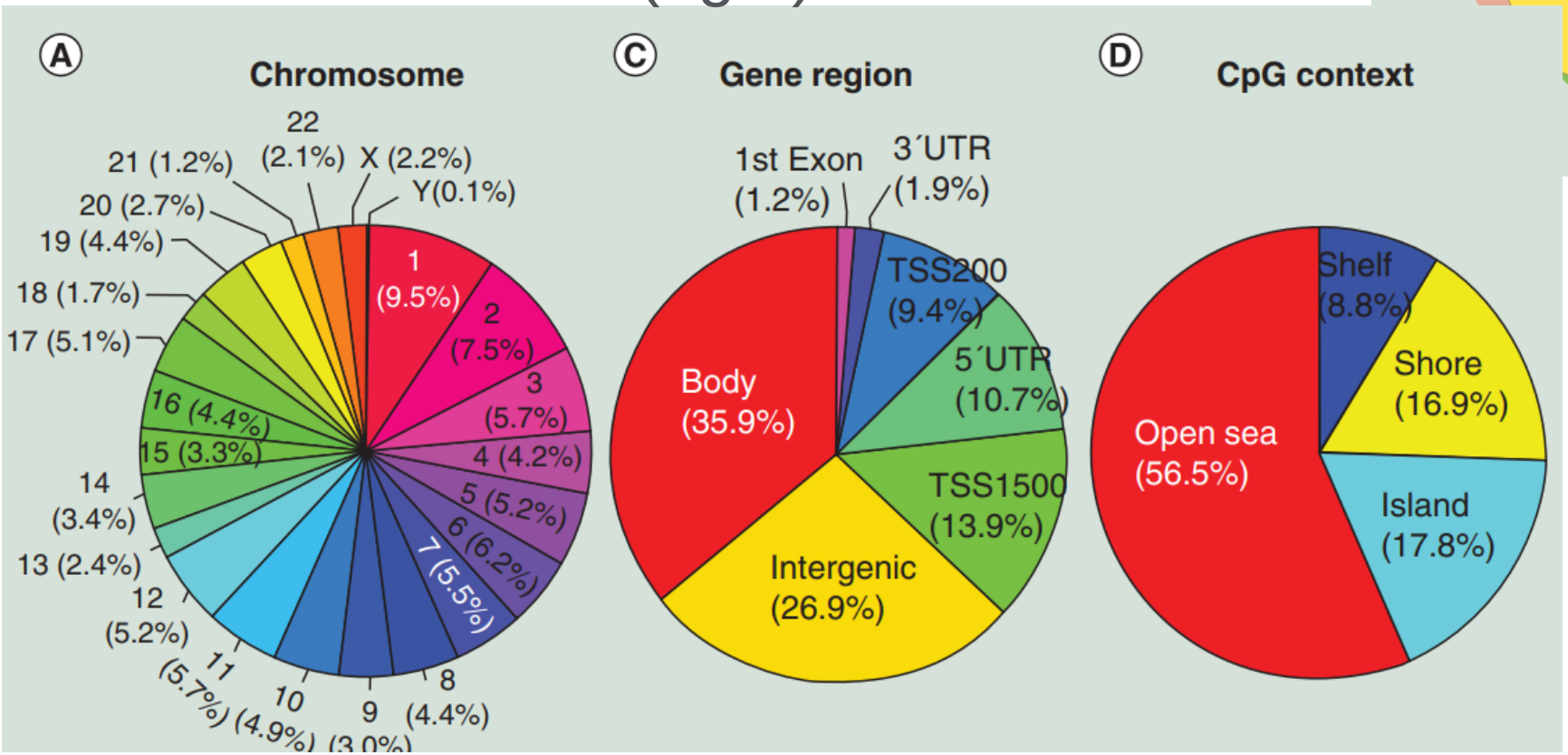


- Arrays vs. WGBS Sequencing
- Arrays (particularly 450K and EPIC) have emerged as the standard for most large studies
  - Limited in information + potential for discovery (<1mil sites)
  - Better reproducibility
  - Consistent information* ← VERY useful feature
  - Lower cost

450K vs. EPIC

Coverage of the EPIC Array (below)
450K vs. 850K/EPIC (right)





Source: Moran, Sebastian, Carles Arribas, and Manel Esteller. "Validation of a DNA methylation microarray for 850,000 CpG sites of the human genome enriched in enhancer sequences." Epigenomics 8.3 (2016): 389-399.

Quick Cost Breakdown

Pyrosequencing	<10 CpGs	~$20/sample
Targeted Bisulfite Sequencing	100s CpGs	~$100/sample
Illumina 450K microarray*	485K CpGs	~$300/sample
Illumina EPIC microarray	850K CpGs	~$330/sample
Reduced Representation Bis Seq	1M CpGs	~$300/sample
Whole Genome Bis Seq	28M CpGs	>$1000

**No longer commercially available*

Practical Considerations

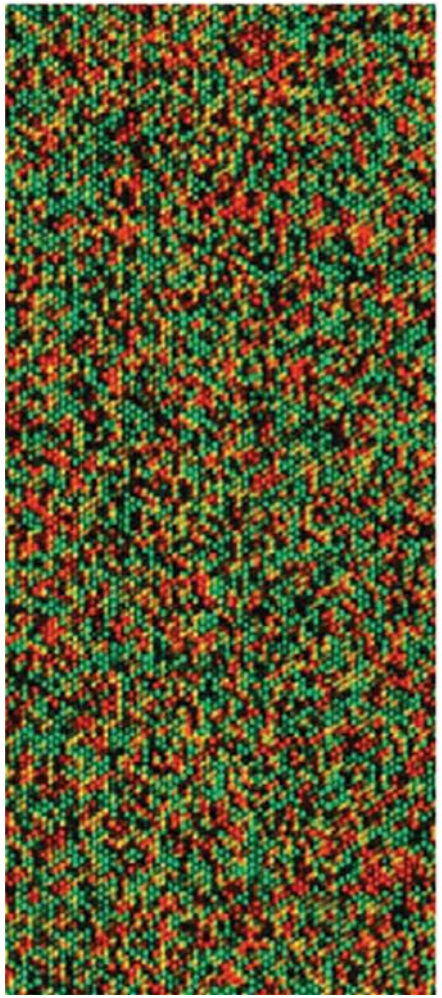
- No guarantee that you will end up with sites of interest unless there is **super** deep sequencing
  - Resolution can be low even with super deep sequencing
- Difficult for:
  - Replication
  - Existing algorithms (e.g. methylation clocks) that require specific CpG sites
- It depends on primary aim(s)

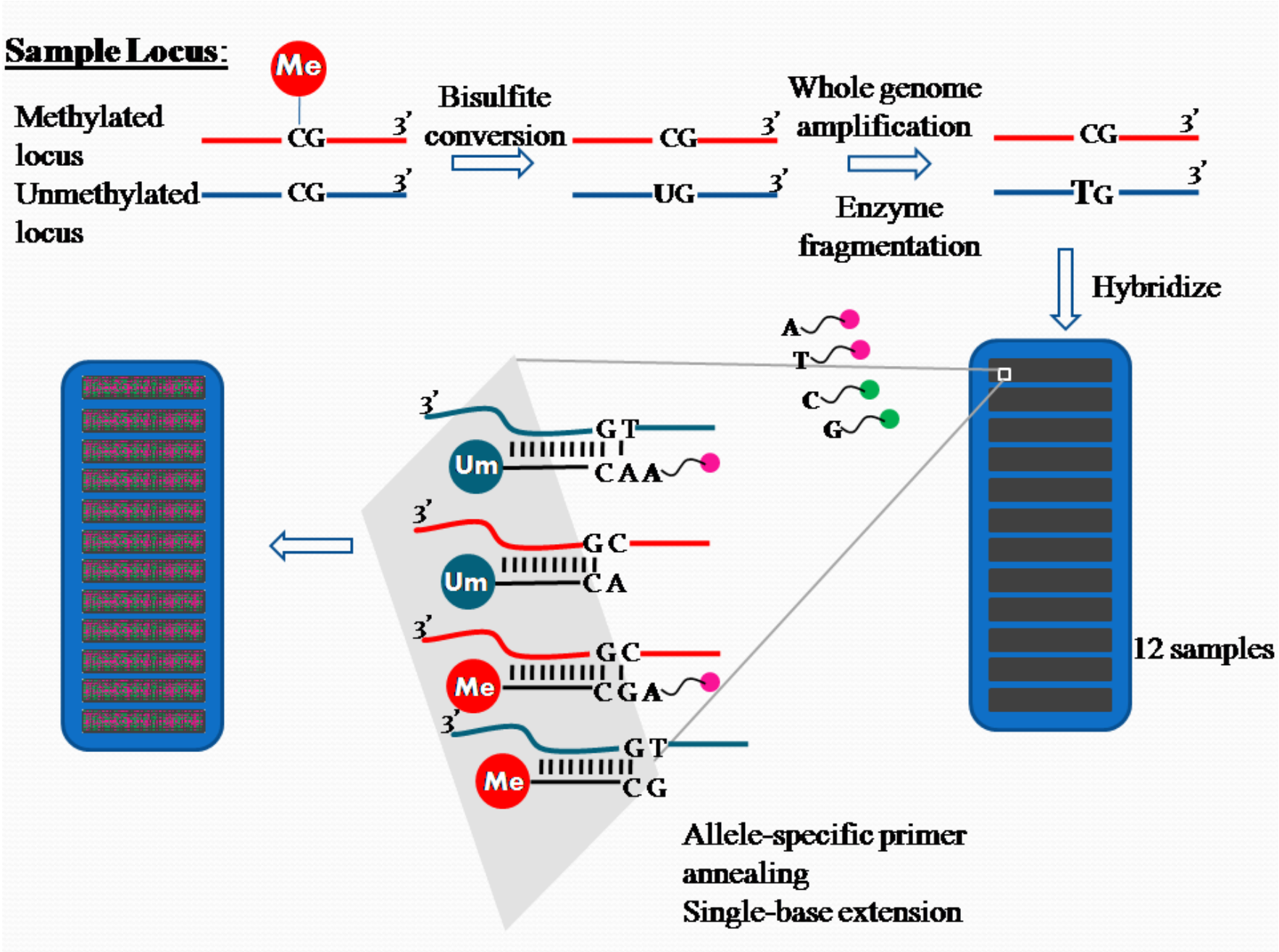
Illumina 450K/EPIC Array

Data Processing

Guide to Illumina Microarrays

- 450,000 - 850,000 “probes” fixed to a chip
  - Each probe is specific for a single site
- BS-DNA is added to the array
- Target sequences bind to probes
- If targets bind to probes, fluorescent signal is released
- Color and intensity of signals is translated into numerical methylation levels at each queried CpG **<- our current task**



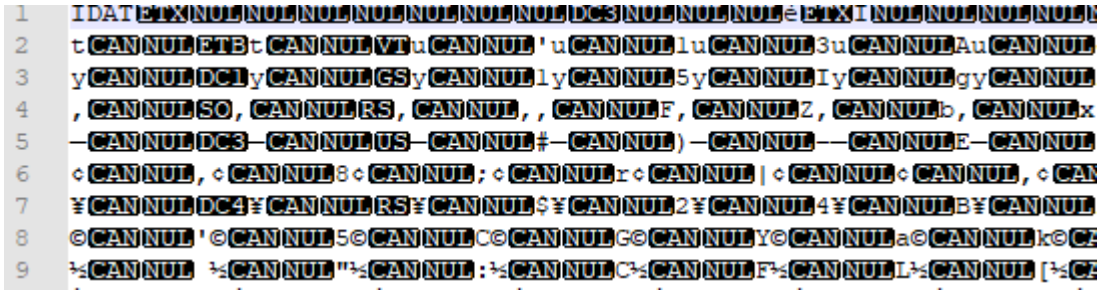


https://www.youtube.com/watch?v=lVG04dAAyvY

The .idats

- The raw files
  - 2 per sample
  - _Grn and _Red
- A “samplesheet” containing information about the assay

200383880006_R06C01_Red.idat	3/14/2017 12:52 PM	IDAT File	13,367 KB
200383880006_R08C01_Grn.idat	3/14/2017 12:52 PM	IDAT File	13,367 KB
200383880006_R08C01_Red.idat	3/14/2017 12:52 PM	IDAT File	13,367 KB
200383880020_R05C01_Grn.idat	3/14/2017 12:52 PM	IDAT File	13,367 KB
200383880020_R05C01_Red.idat	3/14/2017 12:52 PM	IDAT File	13,367 KB
200383880077_R02C01_Grn.idat	3/14/2017 12:52 PM	IDAT File	13,367 KB
200383880077_R02C01_Red.idat	3/14/2017 12:52 PM	IDAT File	13,367 KB
200383880080_R02C01_Grn.idat	3/14/2017 12:52 PM	IDAT File	13,367 KB
200383880080_R02C01_Red.idat	3/14/2017 12:52 PM	IDAT File	13,367 KB
200383880080_R07C01_Grn.idat	3/14/2017 12:52 PM	IDAT File	13,367 KB
200383880080_R07C01_Red.idat	3/14/2017 12:52 PM	IDAT File	13,367 KB
sample.info	6/20/2017 8:24 AM	Microsoft Excel C...	2 KB



IlmnID	cg00002837	cg00026222	cg00049400
S1	0.66324430	0.57973698	0.03822085
S10	0.62771809	0.54093474	0.03256397
S11	0.66466799	0.42062156	0.12967219

Starting the Labs

- Please make sure all packages are installed
- Open up Rscript:
  Preprocessing – ewastools
- Set working directory (line 19) to the correct path

- Run the code along with the presentation
  - Please ask for assistance if you are running into errors
- I will talk about the rationale and what you expect to see at each step

ewastools

- Developed by Drs. Jonathan Heiss (top) and Allan Just (bottom)
- Meant to be lightweight processing pipeline (i.e. no heavy manipulation of data)
- Originated from need to identify mislabeled and contaminated samples

Perspectives on Approach

- Want to be flexible to different uses of the data
  - Ensure consistency between analyses
- Ensure quality control
- Want to process the data as little as possible
  - Do not want to introduce artefacts into the data
- Numerous approaches on preprocessing (e.g. filtering and normalization)
  - No real consensus
  - Depends on application

Step 0 – Importing/Reading Data

- Around lines 15-27

```
> setwd("C:/Users/haoti/Desktop/IGSS/IGSS Epigenetics Course 2021/")
> pheno <- fread("IGSS2021_Meta_data_for_GSE43976.csv")
> meth <- read_idats("GSE43976/" %s+% pheno$gsm, quiet=F)
[1] 622399
  |=================================================================================
=================================================================================| 100%
There were 45 warnings (use warnings() to see them)
```

- read_idats is the key function
  - Input is a list of filenames without the common suffixes (_Red.idat and _Grn.idat), including the filepath
  - Can take filled (.gz) files

Brief Gander at the Data

- Lines 24-25 shown below:

```
> #' Take a look at the dataset
> meth$platform  # The name of the platform (450K/EPIC)
[1] "450K"
> meth$manifest[1:10] #' A manifest with probe IDs, color channel, genomic coordinates and other important information
      probe_id addressU addressM channel next_base chr mapinfo strand probe_type index OOBi     Ui     Mi
 1: rs10796216 14622465 41635319     Red         A          NA                 rs     1    1  39329 302627
 2:   rs715359 18796328 48710462     Grn         C          NA                 rs     2    1  86374 372482
 3:  rs1040870 22687484 20663453     Red         A          NA                 rs     3    2 120242  99780
 4: rs10936224 34619331 30630453     Red         T          NA                 rs     4    3 233473 195590
 5:   rs213028 10622451 24684377     Red         T          NA                 rs     5    4   1025 139183
 6:  rs2385226 46691371 17623494     Grn         C          NA                 rs     6    2 351724  68598
 7: rs11034952 58692423 44652497     Grn         C          NA                 rs     7    3 467655 332692
 8:  rs9292570 48712372 14806497     Red         T          NA                 rs     8    5 372557  47772
 9:   rs654498 36707408 15665335     Red         T          NA                 rs     9    6 257232  50944
10:  rs1414097 23759362 39621311     Grn         C          NA                 rs    10    4 132980 282245
> 
```

Part 2

- Lines 26-27

```
> table(meth$manifest$probe_type) #' Not all probes are targeting CpG sites

    cg     ch     rs
482421   3091     65
> head(meth$controls) #' Similar manifest for the control probes
    address    group channel           name index      i
1: 21630339 STAINING     -99      DNP(20K)     1     NA
2: 27630314 STAINING     Red    DNP (High)     2 165943
3: 43603326 STAINING  Purple     DNP (Bkg)     3 320778
4: 41666334 STAINING   Green Biotin (High)     4 304106
5: 24669308 STAINING     -99    Biotin(5K)     5     NA
6: 34648333 STAINING    Blue  Biotin (Bkg)     6 234885
>
```

Typical Challenges / Errors

2 Common Stumbling Blocks

1. All samples/names in the samplesheet (typically a csv) need to be in the folder
   - So there should be 2x as many .idat files in the folder as there are samples. There will be an error if not
2. The names should be correct
   - Sometimes the .idat files are named after the chip # and position
   - Example - 6929689021_R02C01

Tip

- We called our sample sheet “pheno”
  - We will continuously add information to this sheet

- We have already pre-merged the samplesheet with the phenotypic data (or exposure, or another other data you might need)
- *In your analyses, it might make your life easier if you merge them now*
  - Unless you are processing for other people and/or have large datasets

Step 1 –Sample Failure Check

- Illumina contains 17 control metrics
  - Link to the detailed document
- Samples need to pass all 17 metrics
- Code is around lines 32-39

```
> pheno$failed <- sample_failure(control_metrics(meth))
> table(pheno$failed,useNA='always') #no samples failed, moving on

FALSE  <NA>
   22     0
> |
```

- You should see no failed samples

But What if Some Samples Failed?

Then it depends - How bad did it fail?

Did it fail 1 / 17 metrics by a little bit?

Did it fail 3 / 17 metrics by a lot?

There might be times where you might be inclined to keep samples if they narrowly fail one metric

But please be cautious

Use other steps to inform your decision!

Step 2 – Sex Check

- To check for sample contamination, we can use the X and Y chromosome probes to predict the sex
- Males – high in Y, medium in X
- Females – low in Y, higher in X
- Lines 43-58
  - Embarrassing mistake in the code – line 55 is supposed to list mismatches (M vs. F), but I forgot to change the sex variable from “male/female” to “m/f” so now it thinks it’s all mismatches. Oops.

```
> pheno[,c("X","Y"):=check_sex(meth)]
> pheno$predicted_sex <- predict_sex(pheno$X, pheno$Y, which(pheno$sex=="male"), which(pheno$sex=="female"))
>
> plot(Y~X,data=pheno,type="n")
> text(Y~X,labels=sex,col=ifelse(sex=="female",2,1),data=pheno)
> pheno[sex!=predicted_sex,.(gsm,sex,predicted_sex)]
          gsm    sex predicted_sex
1: GSM1075838 female            f
2: GSM1075839 female            f
3: GSM1075840 female            f
4: GSM1075843 female            f
```

Sex Chromosome Intensities by Sex

- Visualization of intensities
- Lone male sample pretty high in Y intensity
- All female samples low in Y intensity and high in X intensity
- Samples in the “danger zone” need to be flagged and checked

Quirks about This Function

- Does not work well if your samples are all male or all female
  - But you can get X and Y intensities and plot it out for your own sanity

Resolving Sex Mismatches

- For the purposes of our workshop, we will drop the lone male sample
- Can create flags for your own datasets
  - Removal is recommended

```
pheno[,sex_mismatch:=FALSE]
pheno[sex!=predicted_sex,sex_mismatch:=TRUE] # flag sample if there was a mismatch

#' We know there aren't supposed to be males, so let's drop that
meth <- drop_samples(meth, which(pheno$sex=="male"))
pheno <- pheno[-which(pheno$sex=="male"), ]
#Now we have 21 samples
```

Step 3 – Detection P-Values

- Some target probes might not have worked
  - Empty well
  - Dust on chip
  - Poor PCR
- So we want to look at total intensity relative to background “noise”
  - Derive a p-value and address situations where there is insufficient separation between probe intensity and background

Detection P-Values

- Lines 68-96
- Only 2 lines are actually necessary:
  - meth = ewastools::detectionP(meth)
  - meth = ewastools::mask(meth,0.01)
- The rest of the code given are there for when you have both sexes
  - Intent is to show differences in the # of detected probes on X / Y chromosomes

So How Many Samples/Probes Did this Affect?

- 0.124%
- This will vary across datasets and is related to the quality of the samples and assay

```
> meth$detP[-chrY,] %>% is_weakly_greater_than(0.01) %>% table(useNA="ifany")
.
   FALSE     TRUE     <NA>
10175284    12596      501
> round((12596/(12596+10175284))*100,3)
[1] 0.124
>
```

- Note – there is an additional QC step (which we do not show) where you can assess # of failed probes per sample

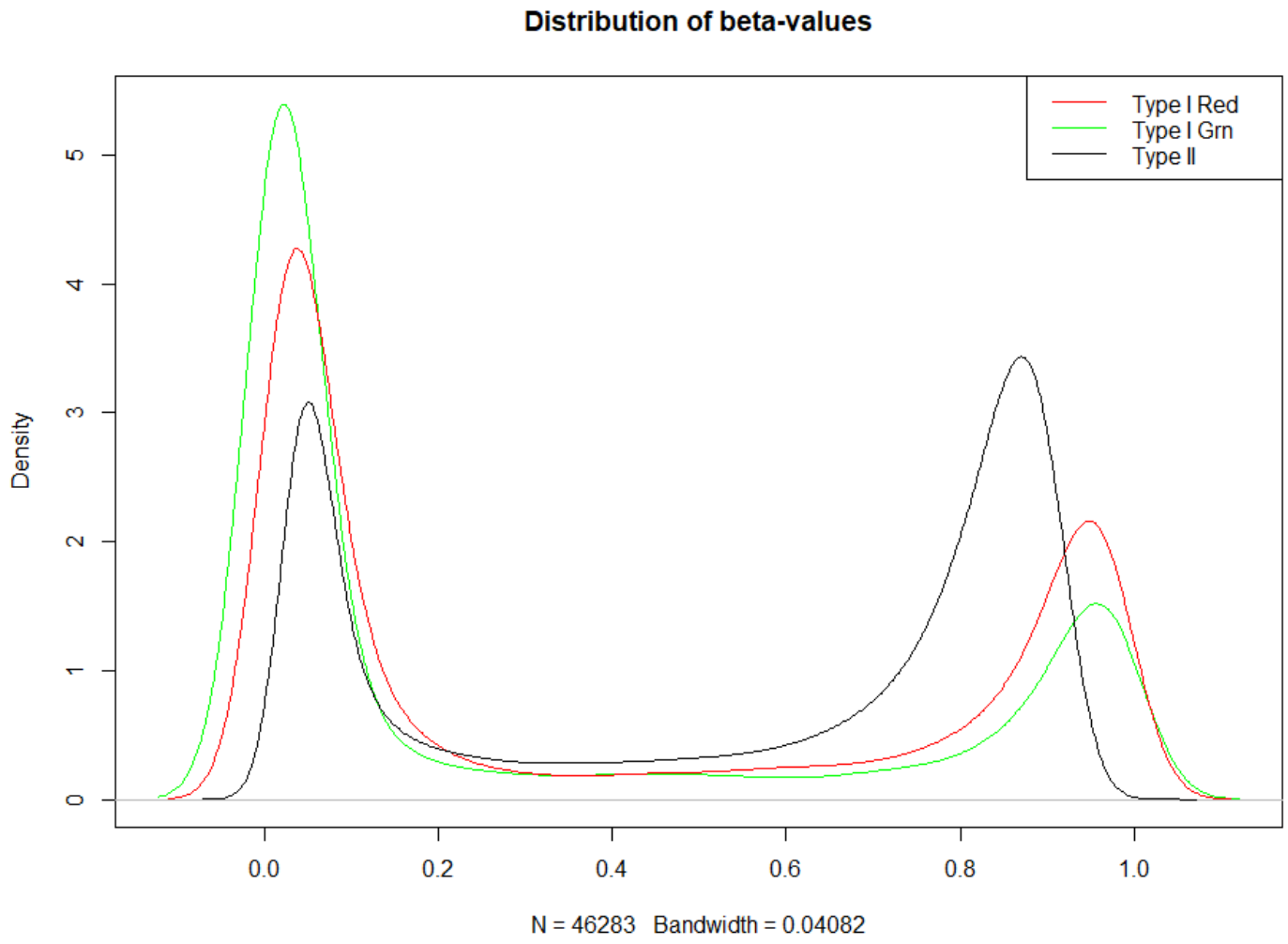
Masking vs. Dropping Probe

- Some pipelines will drop all of the probes that have any detection p-value higher than the threshold
  - Benefit – more stringent, easier to manage during batch corrections
  - Downside – lose all information about this site

- Prefer to just drop individual observations (making them missing) than the entire probe/site for the whole population
  - People can always exclude them later if necessary

Step 4 – Dye Bias Correction

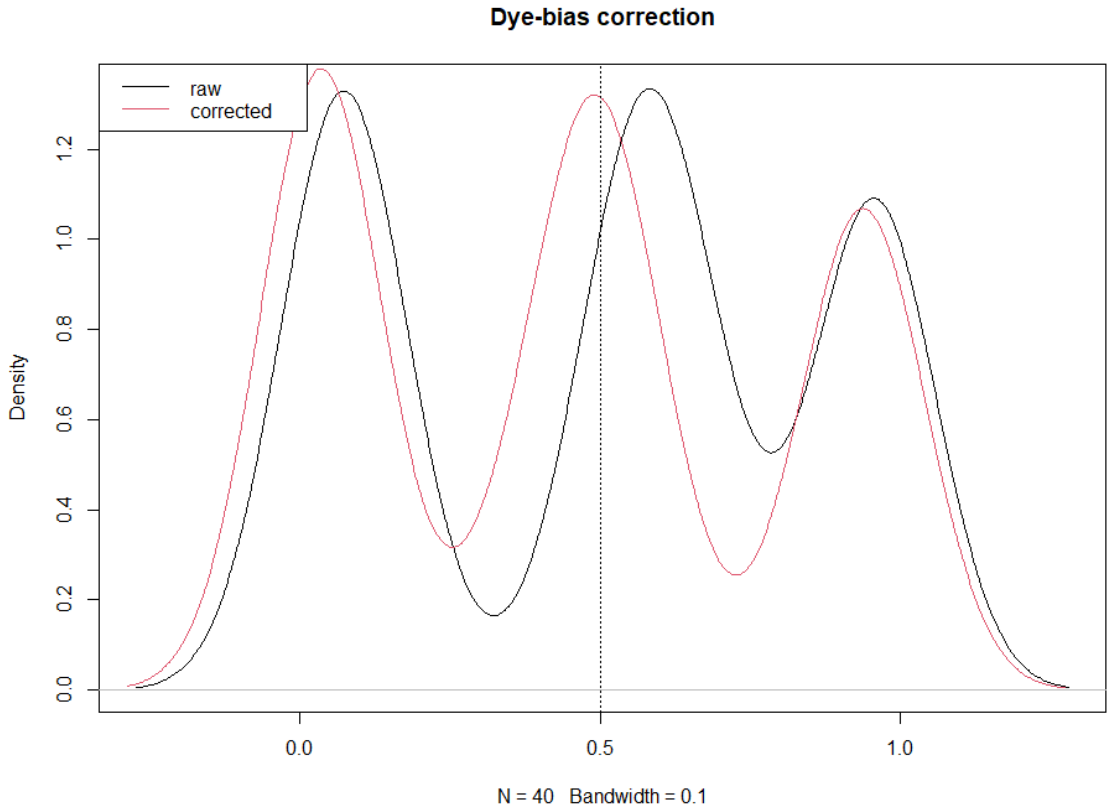
- Differences in red and green channels can impact resulting methylation values
- Correction will improve technical replicate reproducibility
- Lines 103-129



See the Changes

```
> #
> color_bias = meth %>% dont_normalize #without dye bias correction
> beta = meth %>% correct_dye_bias %>% dont_normalize #with dye bias correction
> #' We can look at a few methylation values on the fly and see whether dye-bias correction changed them
> meth$manifest$channel[201:203] # One probe for each type/color channel
[1] "Both" "Red"  "Grn"
> color_bias[201:203,1:3] %>% round(4)
           GSM1075838 GSM1075839 GSM1075840
cg27487046     0.0435     0.0576     0.0746
cg06091566     0.1052     0.1132     0.1094
cg03735847     0.0102     0.0221     0.0145
> beta      [201:203,1:3] %>% round(4)
           GSM1075838 GSM1075839 GSM1075840
cg27487046     0.0192     0.0232     0.0329
cg06091566     0.1052     0.1132     0.1094
cg03735847     0.0052     0.0112     0.0069
```

- Figure on the right shows the % methylation values generated from raw data for heterozygous SNPs
  - It should be at 0.5
  - Correction brings that to 0.5



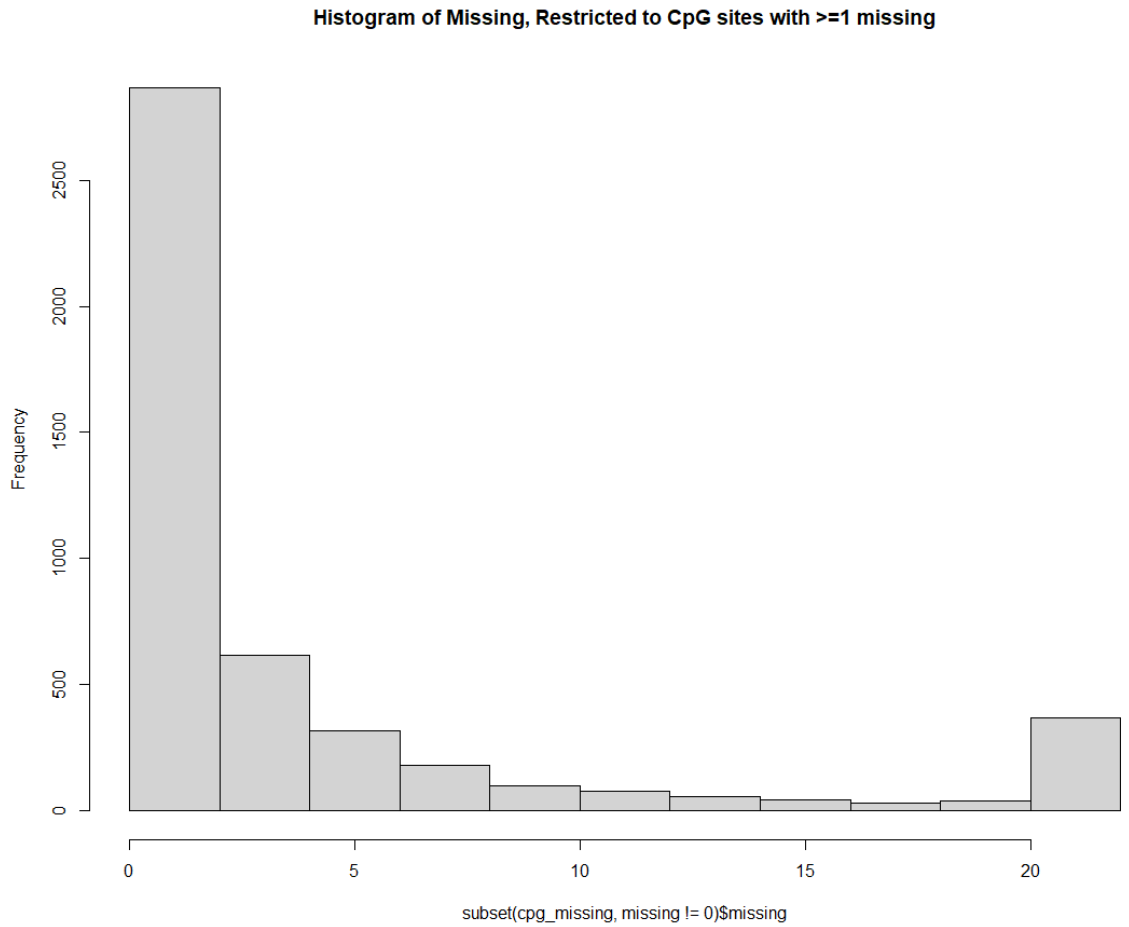
Step 5 – Drop “Bad” Probes

- Optional step, but useful
- A trade off – stringency vs. more data
  - Decision may depend on research question
  - Example – for DNAm clock, you might want to keep as probes as you can
- Drop probes if too many samples failed to overcome the background noise for any given probe (as determined by the detection p-values)
- 10% is a commonly used threshold

Vast Majority are Missing <10%

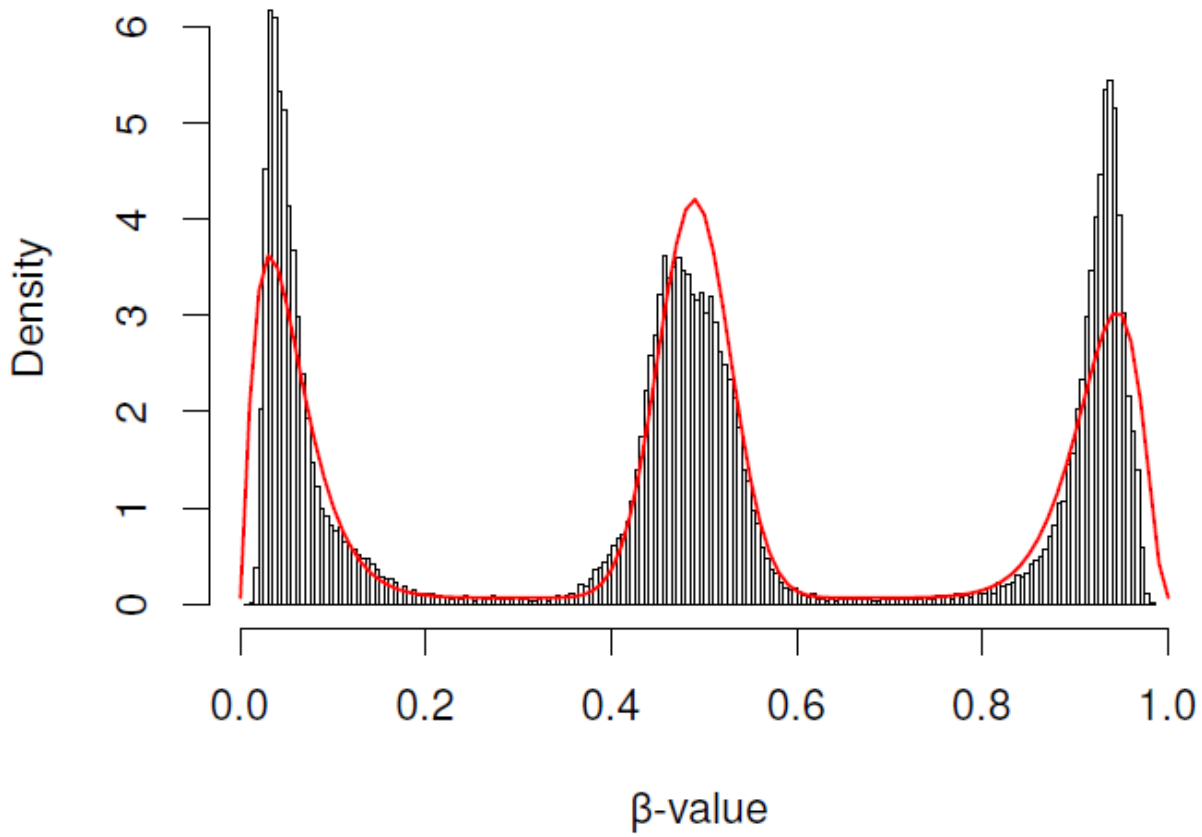
- Lines 135-144
- Removed 1816 CpGs

- Note – keeping probes that have large % missing will cause errors with downstream analyses



Step 6 – SNP Outliers

- 450K + EPIC are SNP microarrays
  - SNPs being artificially generated through bisulfite conversion of unmethylated Cs
  - Some probes do target real SNPs
- Beta values represent genotypes (trimodal distribution)



SNP Outliers

- Some SNPs cannot be assigned to one of the genotypes
  - Fall in between the three peaks

- Model outliers by adding uniform distribution component to mixture model. Compute average log odds of being an outliers across all SNP probes

- Indicate either poorly performing arrays or degraded or contaminated samples

Step 7 – Remove More Probes

- Cross-hybridizing
  - Probes that are cross-reactive to similar, but not target, sequences
- SNP-related probes
  - CpG sites where methylation values are driven by SNPs

- For simplicity and ease, we will use a single function (rmSNPandCH)
- Lines 172-176

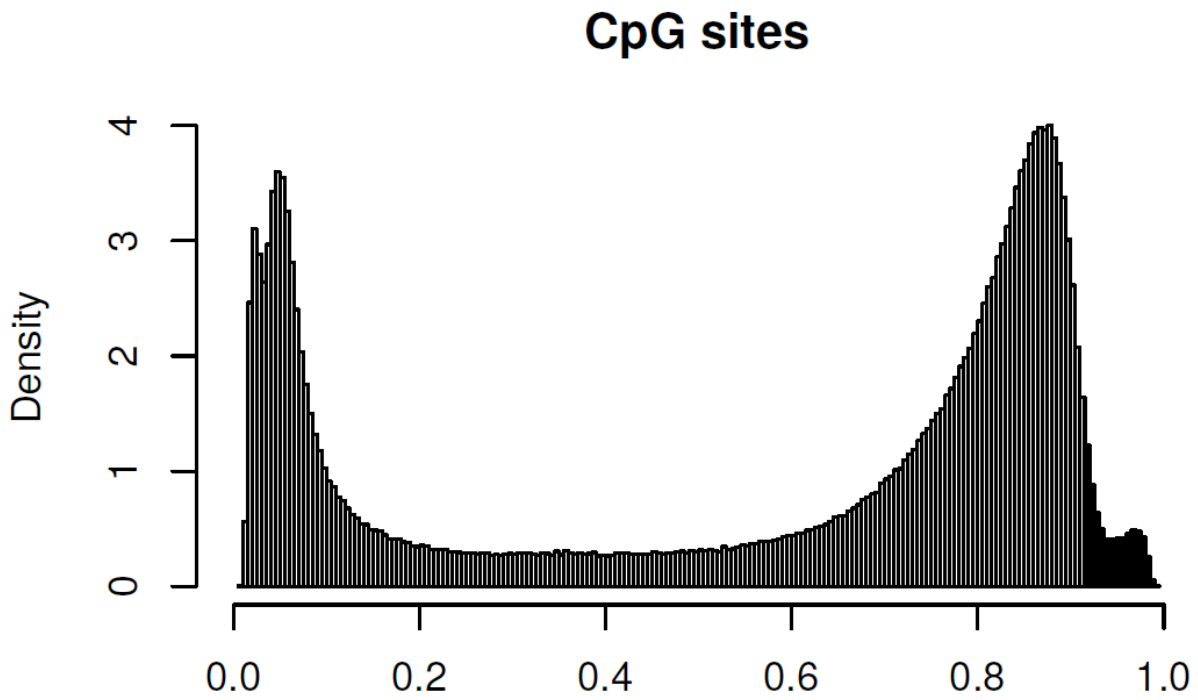
More about CH + SNPs

- There are many other resources to remove CH and SNP-related probes
- For examples and resources, see:
  - https://pubmed.ncbi.nlm.nih.gov/27717381/
  - https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5331917/
  - https://pubmed.ncbi.nlm.nih.gov/31861999/

- Another good practice is to visually check distributions of significant clusters found by analysis to ensure it is not confounded by genetics

What Does the Data Look Like?

	GSM1075838	GSM1075839	GSM1075840	GSM1075843
rs10796216	0.06353591	0.05926677	0.90563380	0.50598291
rs715359	0.97218212	0.52176281	0.53078342	0.98243898
rs1040870	0.08163867	0.09487410	0.09511785	0.89518497
rs10936224	0.90458580	0.49008811	0.46022392	0.46158445
rs213028	0.51340088	0.03310522	0.49241809	0.02756594
rs2385226	0.40266315	0.38304873	0.36318737	0.37678500
rs11034952	0.48304757	0.03148816	0.45518077	0.52297222
rs9292570	0.51127370	0.49603935	0.47750050	0.51329056
rs654498	0.50506024	0.88681592	0.51436782	0.11824462
rs1414097	0.93703373	0.03179402	0.49985804	0.40845019
rs13369115	0.91523734	0.92284372	0.51751972	0.06992543
rs10033147	0.48863636	0.90751945	0.88516969	0.42378559
rs3936238	0.03796233	0.04020333	0.04094196	0.44737374
rs1520670	0.54430811	0.55664770	0.91342894	0.05345455



Other Normalization Steps

- Background correction
- Probe type correction
- Batch correction

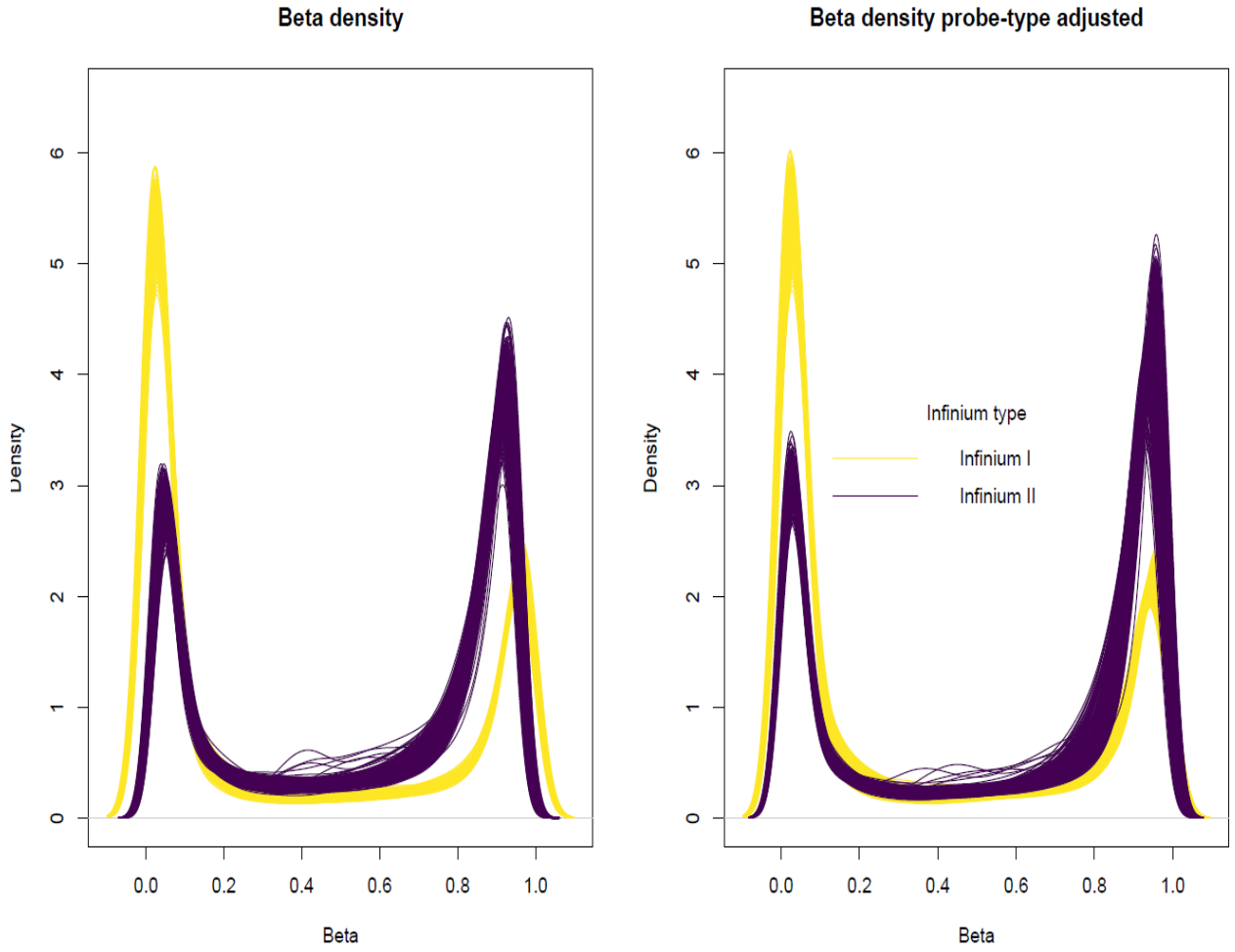
Normalization

- General term that refers to removal of unwanted variations in the data
- Already have dye-bias correction, but could normalize for background and technical variations

- Could correct for background noise
  - preprocessNoob (background + dye bias)
  - preprocessFunnorm (where it uses Noob)

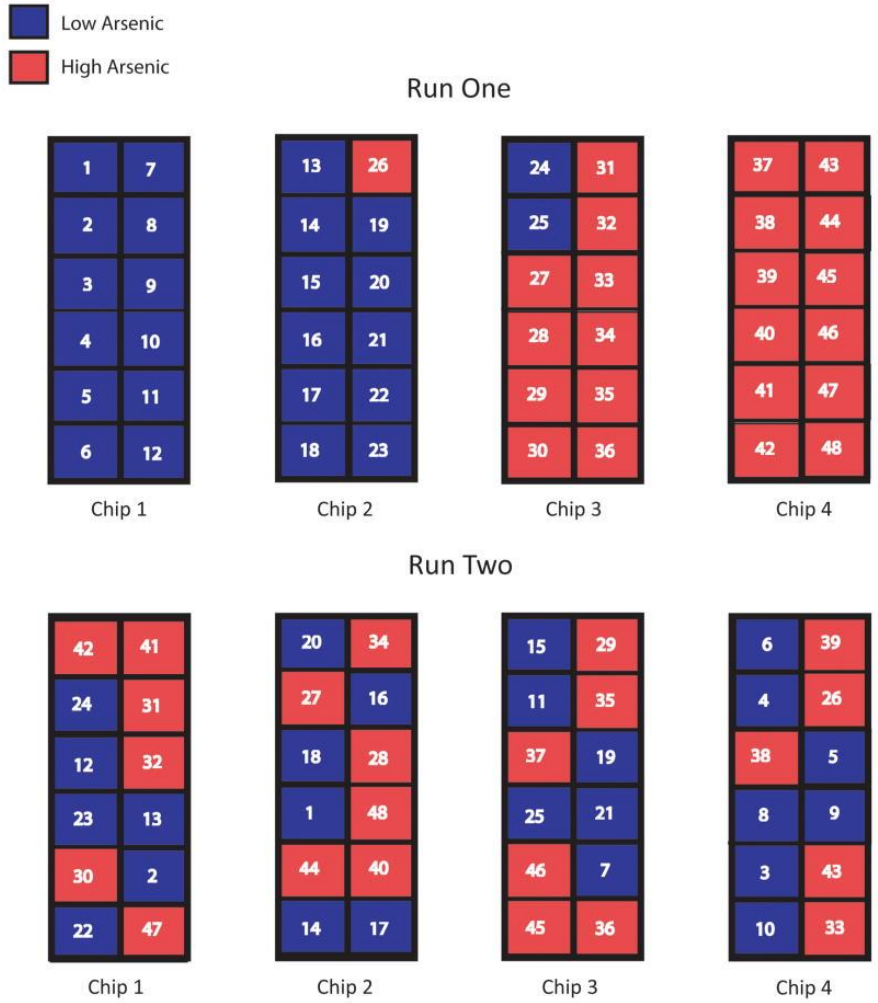
Probe Type Adjustment

- 2 different probe types
  - Incorporate green and red channels differently
  - rcp function from ENmix package
  - Partially addressed by background and dye bias correction
- Will not matter if analysis is CpG by CpG
- May induce noise in regional/cluster analyses



Batch Effects

- Historically a mandatory step in most microarrays
- In extreme cases (see figure), it is a necessity
- Recently, some question of its necessity when the samples are appropriately randomized



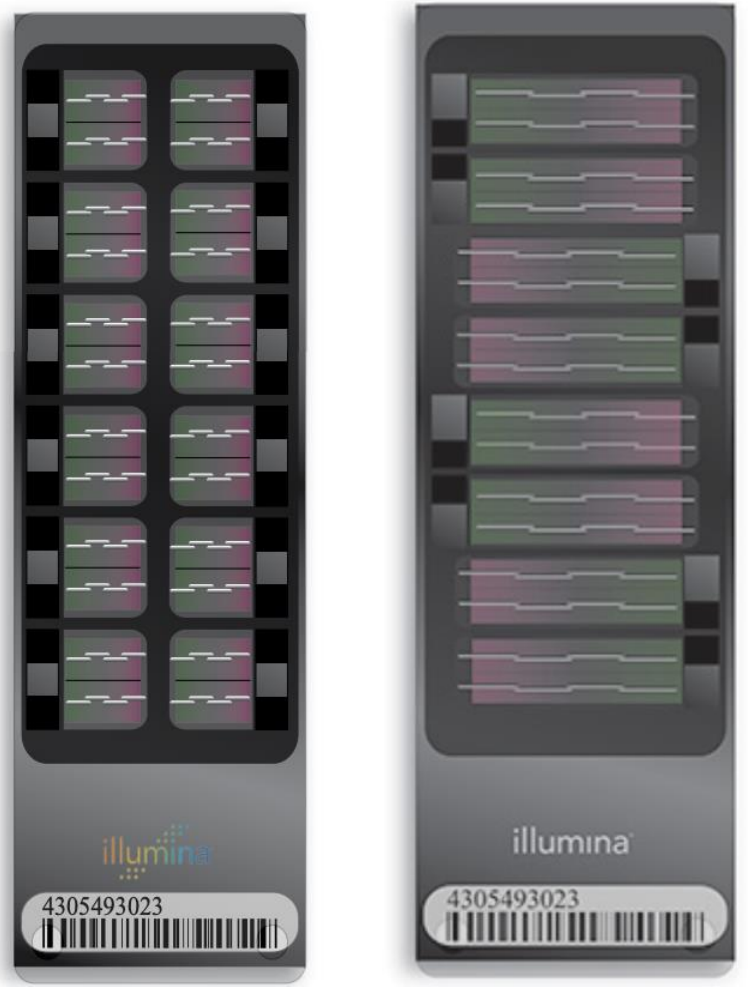
Harper et al., 2013. 10.1158/1055-9965.EPI-13-0114

Technical Batches

- Actual batch effects can arise from a variety of sources:
  - Assay itself - batch effects in Illumina arrays are observed between chips and chip positions.
  - Bisulfite conversion plates
  - Operator, reagent lot
    - Major differences can arise from types of kits used
  - Machine drift
- Many are worth checking

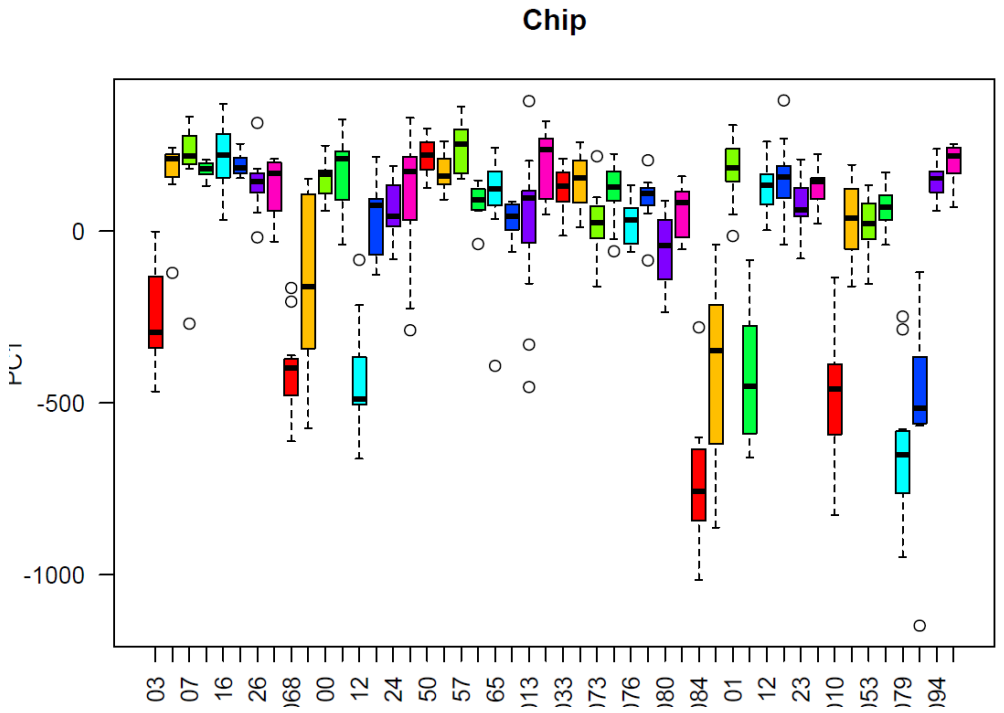
2 columns x 6 rows

1 column x 8 rows



Not Always an Easy Decision

- Each 850K chip has 8 samples, should we expect every 8 samples to look exactly alike?
  - What if one or two are truly biologically different?
    It would appear different
  - What happens to this variability when we batch correct?
- Figure right shows real example



General Set of Steps

- Run PCA
  - Extract the first few principal components
- Examine (visually and statistically) if batches are associated with PCs
- Run ComBat or some other normalization
  - Popular function to do batch correction (sva package)
  - Uses an empirical Bayes framework to adjust for batch effects
- Note – transform your data to M-values before and then transform it back if you want beta-values!

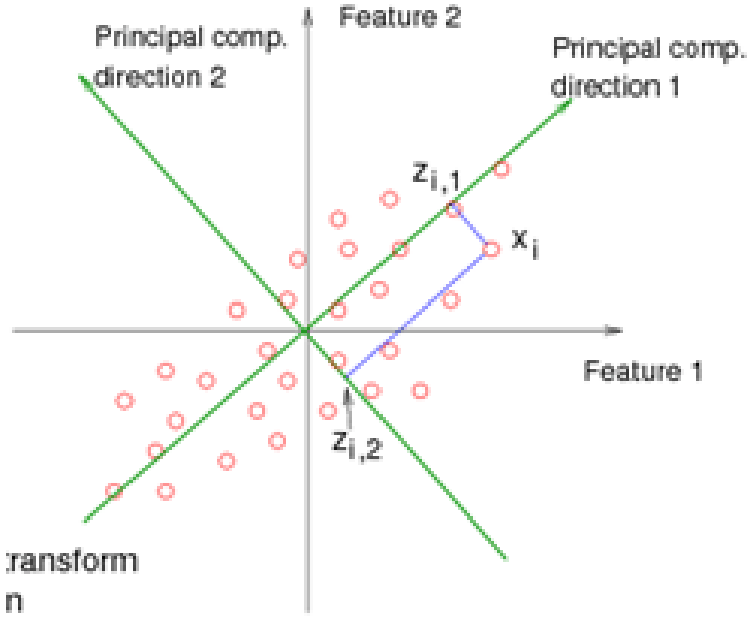
Alternative to Batch Correction

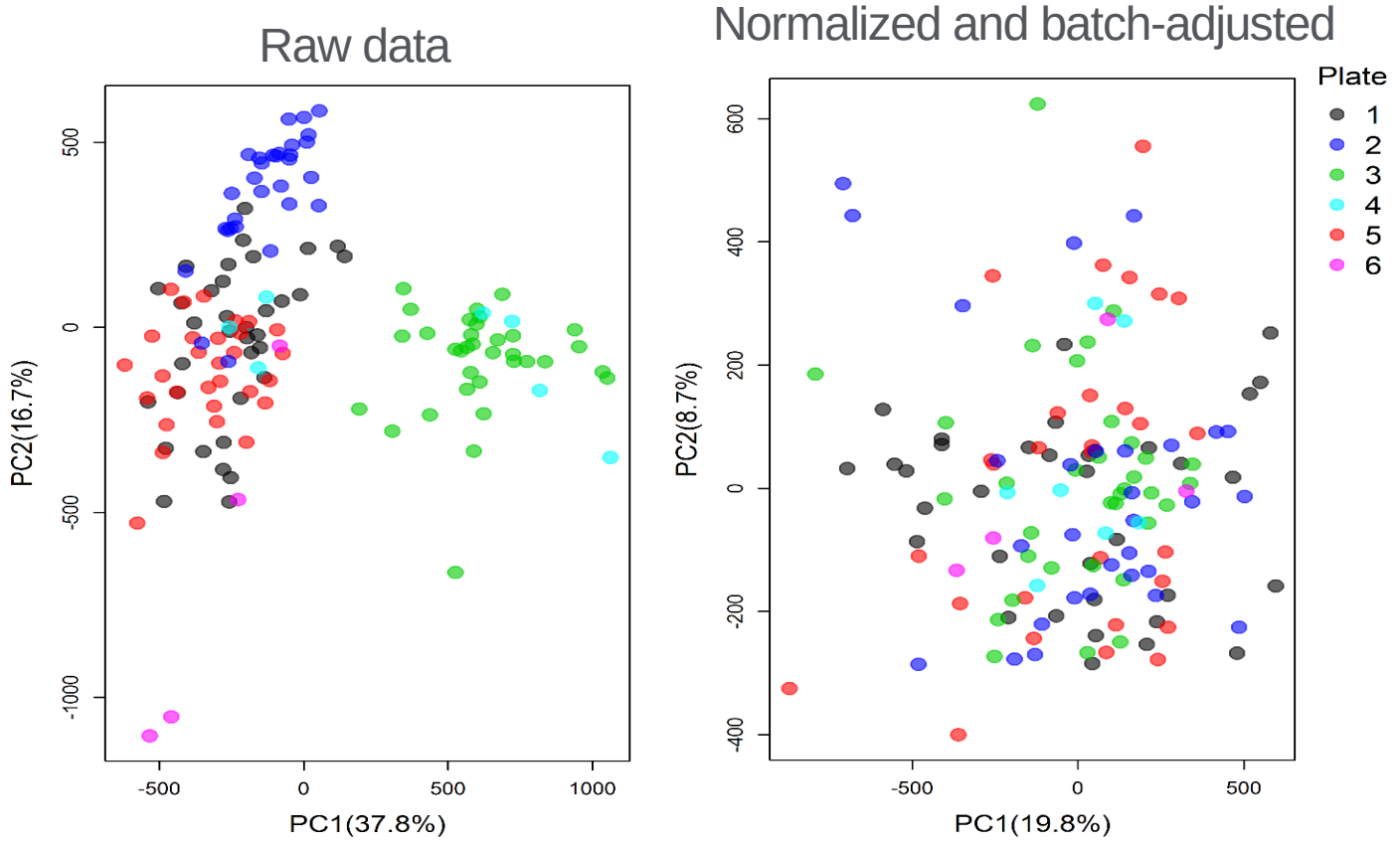
- Can always include batch variable(s) in the modeling stage
- If DNAm is the outcome
  - If there is a batch effect, could increase precision
  - If there is no batch effect, takes away a few degrees of freedom
- If DNAm is the independent variable
  - Done properly, batch should not be associated with outcome.
  - Most likely scenario is that you lose a few degrees of freedom

Principal Components Analysis (PCA)

Dimensionality reduction technique.

Allows us to capture the major sources of variability in the data.





	Raw PCA	Normalized + ComBat Adjusted
Principal Component	Variance explained	Variance explained
PC_1	37.8%	19.8%
PC_2	16.7%	8.7%
PC_3	6.2%	4.5%

Alternative Pipelines

- Another popular pipeline is based around the package “minfi”
  Preprocessing – minfi
- More traditional, less “flexible”
- Major differences in code:
  - Normalization is the first step (common options: NOOB, SWAN, BMIQ)
  - Detection p-value calculated differently
  - Probe type adjustment (via rcp)

There are Other Steps

- Lightweight and simple approach shown today
- There are other QCs one can do. Examples –
  - Apply PCA to the data and look for outliers and weird samples
  - Use agreement of SNPs to identify mislabeled samples (in twins studies or repeated measures from same individual)
  - Plot out beta density of each sample to see if samples look odd
- Many labs' pipelines will involve extra steps

Publicly Available Data

- Often pre-processed for you. No access to idats.

- Most important thing…
  Make sure you have documentation and know whatever it is people did

Flexible Approaches

- When processing data from human studies, it might be used for different analyses
- Want flexibility for most scenarios
  - Not dropping all non-detected probes (think detection p-value stage)
    - What if you need a probe for DNAm clocks?
  - No batch correction
  - Keep data consistent across analyses because processing matters

Memory Requirements

- Access to virtual machines or clusters are helpful, although not strictly necessary for smaller studies
- RAM dependent
  - Can be intensive for large sample sizes
- Data processing takes the most memory
  - ~200 EPIC samples can be done on 16GB of RAM
  - Most subsequent analyses require less RAM
- Aggressive management of R environment

Contact Information

Dr. Allison Kupsco

- ak4181@cumc.columbia.edu

Dr. Howie Wu

- hw2694@cumc.columbia.edu

Happy to help!