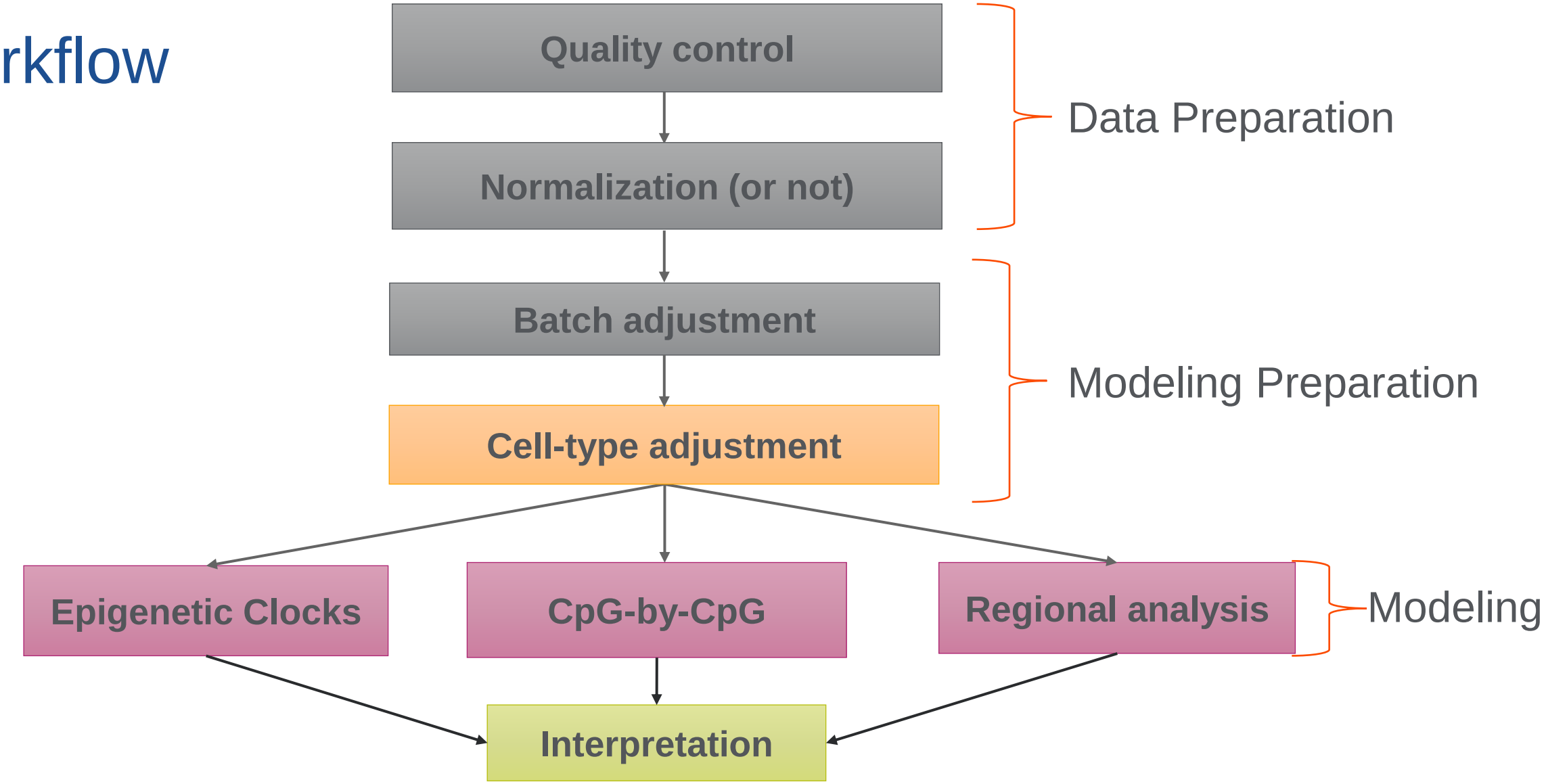
# Epigenome-wide association studies (EWAS)

IGSS 10/29/2021

Allison Kupsco, PhD
Assistant Professor of Environmental Health Sciences
Columbia University Mailman School of Public Health

# Workflow


Quality control
Normalization (or not)
Batch adjustment
Cell-type adjustment
Epigenetic Clocks
CpG-by-CpG
Regional analysis
Interpretation
Data Preparation
Modeling Preparation
Modeling

# So you're ready to begin your EWAS…

- Think critically about your research question.
- Generate your hypotheses.
- Consider your study design.
- Determine your confounders and covariates.
- Decide on your modeling strategy.

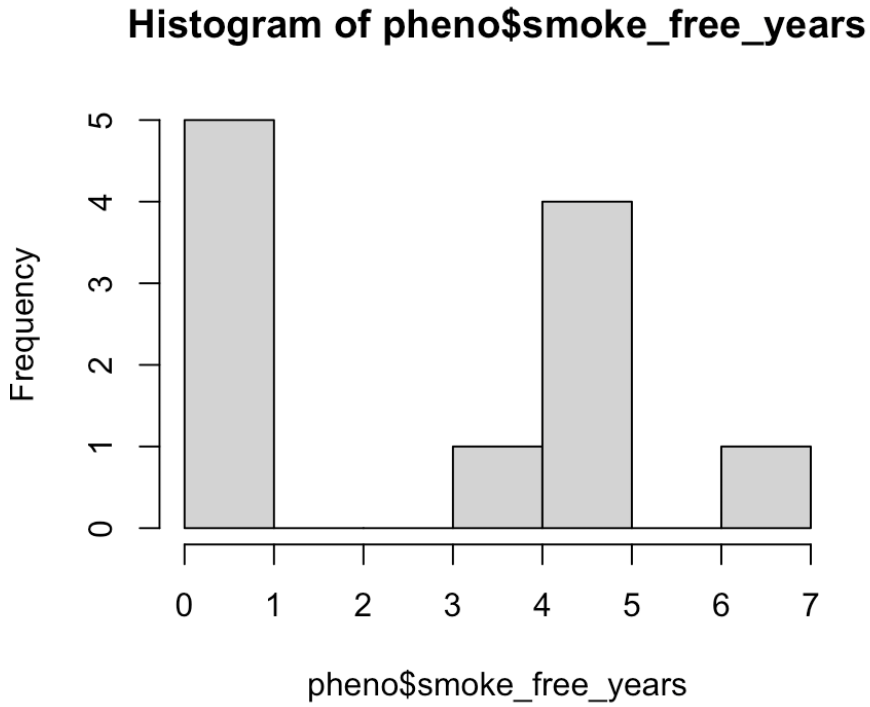
# Part 2:

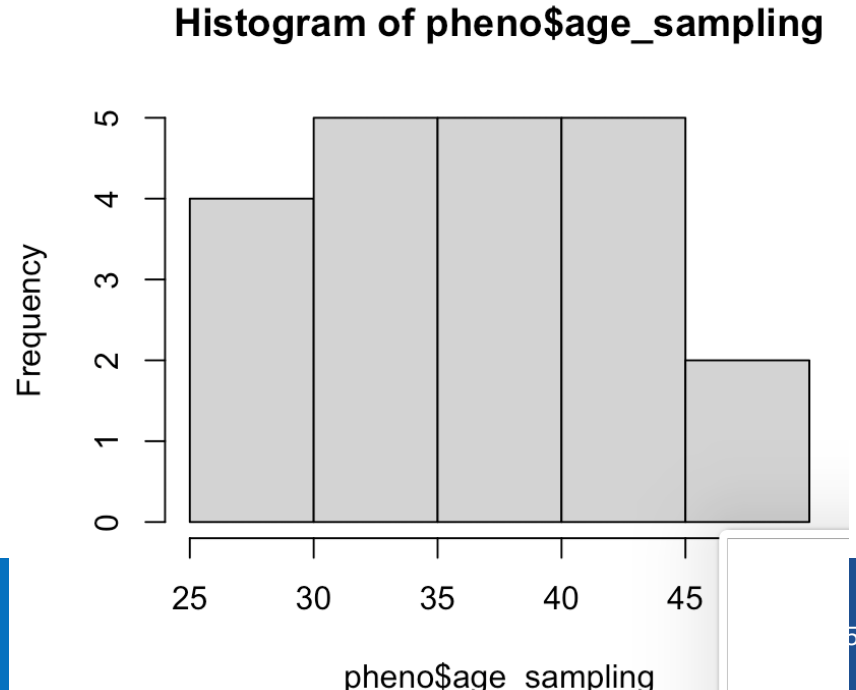
Restart R and open the “IGSS_2021_Batch_CellType_EWAS_Pipeline.R” script.

Reset your working directory and load the packages.

# Explore the phenotype data

```
table(pheno$smoking_evernever)

Ever Never
  11    10
> table(pheno$smoking_5years)

Before_5years  Never_Smoker Within_5years
            5            10             6

hist(pheno$smoke_free_years)
hist(pheno$age_sampling)
```



# Explore the meta data

```
> table(pheno$Sentrix_ID)

6929689021 6929689032 6929689045
         7          7          7
```

Sentrix ID is the chip name

```
> table(pheno$Sentrix_Position)

R01C01 R01C02 R02C01 R02C02 R03C01 R03C02 R04C01 R04C02 R05C01 R05C02 R06C01 R06C02
     2      1      3      1      3      2      2      2      1      1      2      1
```

Sentrix position is the row and column indicator

# Analysis Practice: Working with your cleaned data

This is the resulting file from processing

```
betas.clean <- readRDS("cleaned_betas.rds")
pheno <- read.csv("IGSS2021_Meta_data_for_GSE43976.csv", strip.white=T, stringsAsFactors=F) #

#remove the male participant
pheno <- pheno[pheno$sex != "male",]

#make sure the IDs in pheno match the column IDs in the betas and the order in the WB object
all.equal(pheno$gsm, colnames(betas.clean))
```

Always make sure your meta data order matches your betas

# PCA to explore variability and batch effects:

Required packages: sva

```
betas.clean2 = na.omit(betas.clean)
```

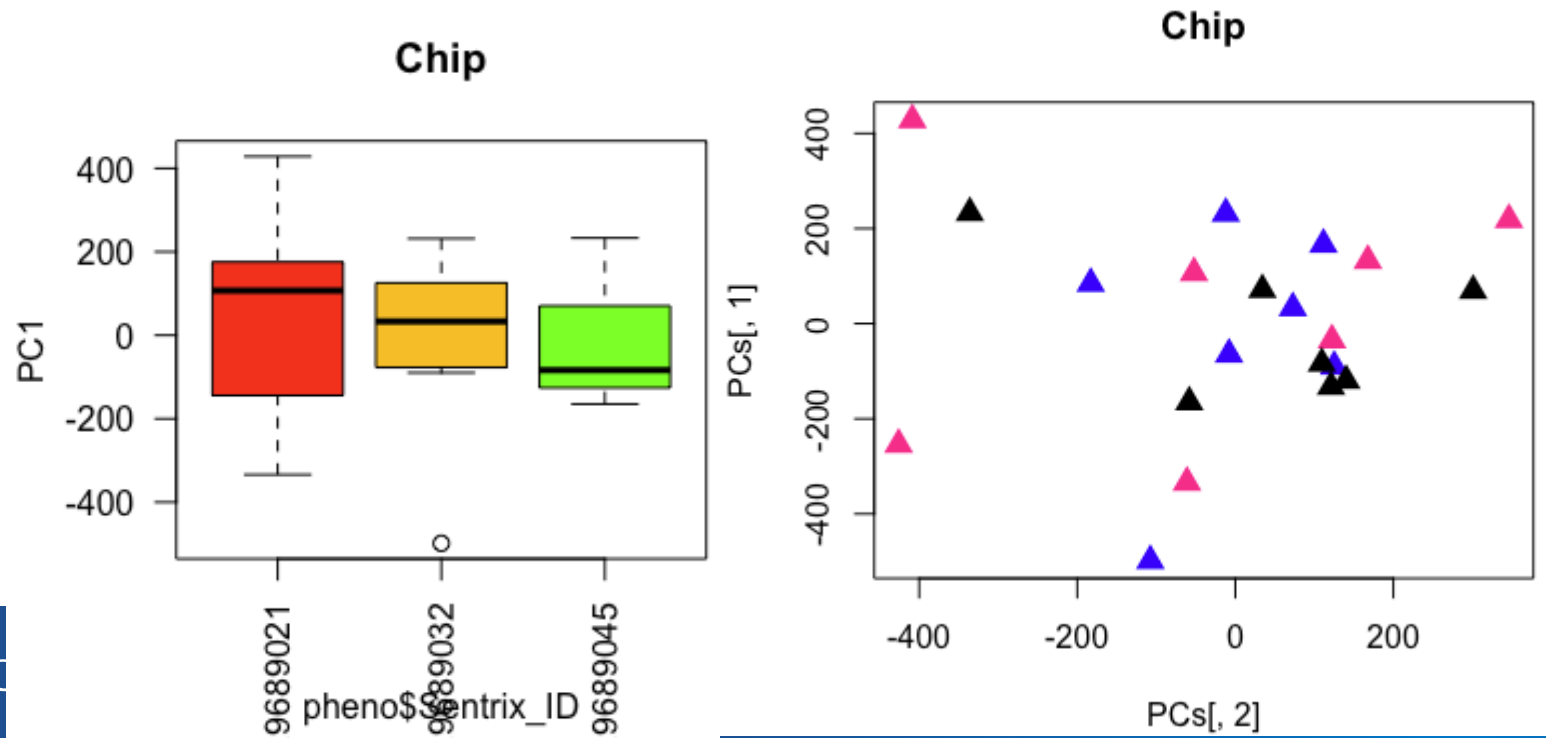
Cannot handle NAs

```
#' Calculate major sources of variability of DNA methylation using PCA
#' Need to transpose data so IDs are rows and CpGs are columns
PCobject <- prcomp(t(betas.clean2), retx = T, center = T, scale. = T)

#' Extract the Principal Components from SVD
PCs <- PCobject$x
#' Proportion of variance explained by each additional PC
cummvar <- summary(PCobject)$importance["Cumulative Proportion", 1:10]
knitr::kable(t(as.matrix(cummvar)),digits = 2)
```

| PC1 | PC2 | PC3 | PC4 | PC5 | PC6 | PC7 | PC8 | PC9 | PC10 |
|---|---|---|---|---|---|---|---|---|---|
| 0.11 | 0.21 | 0.27 | 0.34 | 0.39 | 0.44 | 0.49 | 0.54 | 0.58 | 0.62 |

# Visually explore the variance in the data

```
#' Is the major source of variability associated with chip?
par(mfrow = c(1, 1))
boxplot(PCs[, 1] ~ pheno$Sentrix_ID,
        ylab = "PC1",las=2, main="Chip",col=rainbow(8))
plot(PCs[,1]~PCs[,2],cex=1.5, pch=17,col=c("deeppink","blue", "black")
     [factor(pheno$Sentrix_ID)], main = "Chip")
```

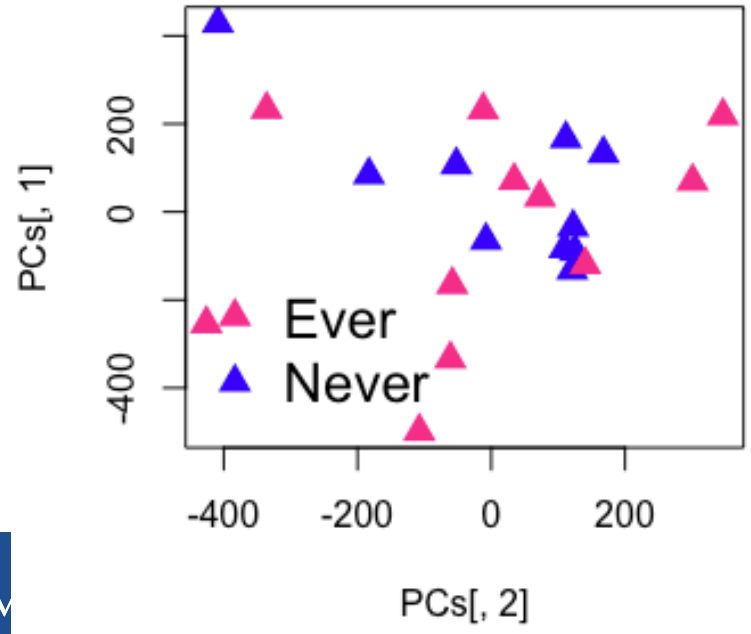


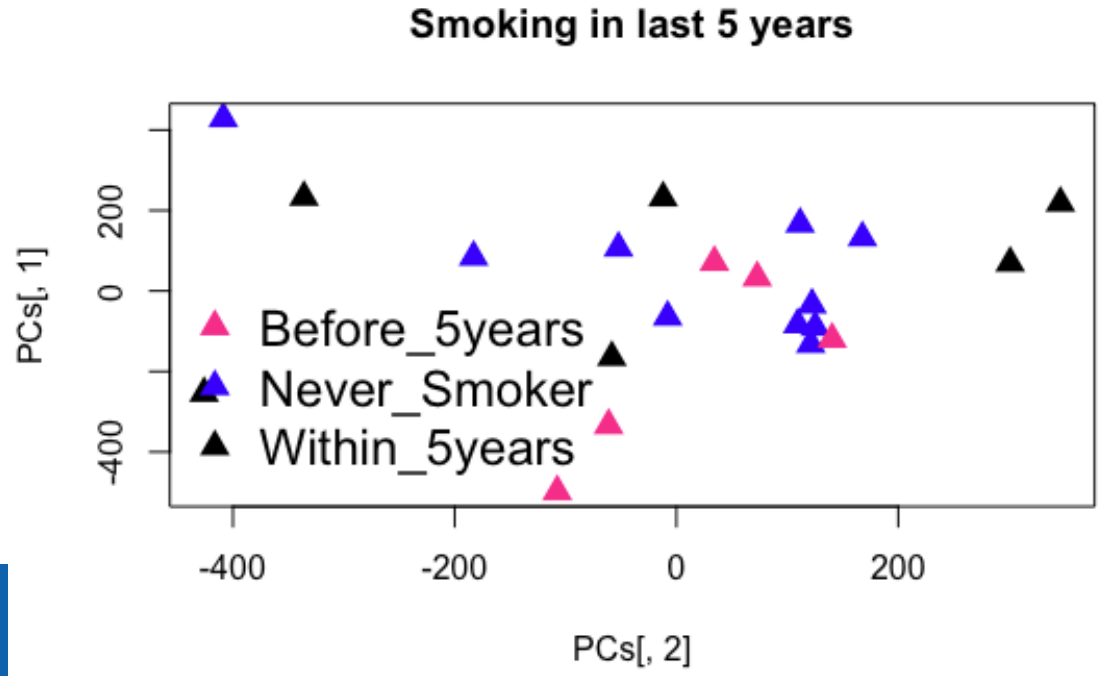

Chip does not appear to be associated with PCs 1 or 2

# Is the variability associated with smoking?

```
plot(PCs[,1]~PCs[,2], pch=17,col=c("deeppink","blue")[factor(pheno$smoking_evernever)],
     cex=1.5, main = "Smoking")
legend("bottomleft", legend=levels(factor(pheno$smoking_evernever)),bty='n',
       cex=1.5,pch=17,col=c("deeppink","blue"))
```

```
plot(PCs[,1]~PCs[,2],cex=1.5, pch=17,col=c("deeppink","blue", "black")
     [factor(pheno$smoking_5years)], main = "Smoking in last 5 years")
legend("bottomleft", legend=levels(factor(pheno$smoking_5years)),bty='n',
       cex=1.5,pch=17,col=c("deeppink","blue", "black"))
```



# Explore the variability with regression

```
#' What are the major sources of variability?
#' Run linear models with the first 10 PCs as outcomes
pheno$Sentrix_ID = as.factor(pheno$Sentrix_ID)

variables = c("sample_year", "smoking_evernever",
              "smoking_5years", "pack_years", "Sentrix_ID")

res_all = data.frame()
for (i in 1:10) {
  for (j in variables) {
    res = tidy(lm(PCs[,i]~pheno[,j]))
    res$PC = i
    res$variable = j
    res_all = rbind(res_all, res)
  }
}
```

Loop for regressions over PCs and variables

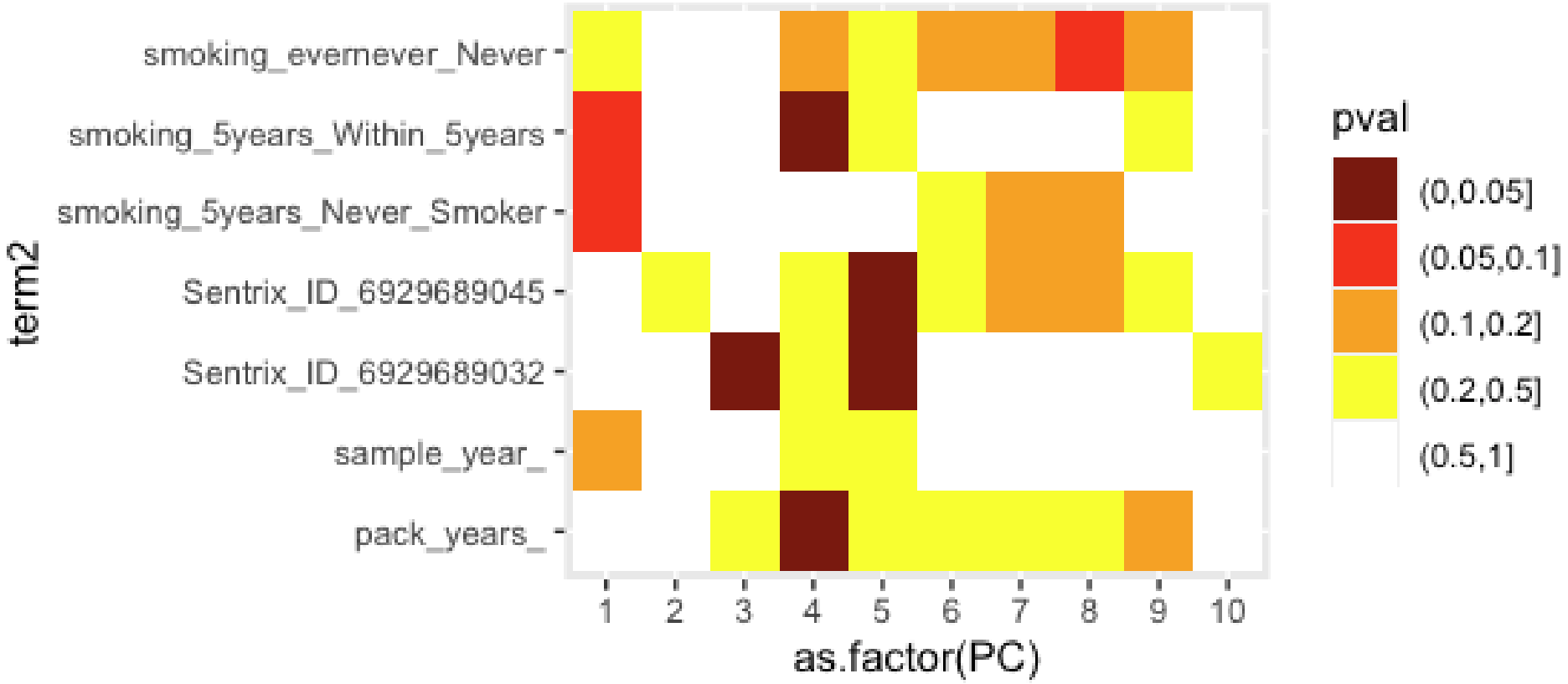
```
res_all = subset(res_all, term != "(Intercept)")
res_all$pval = cut(res_all$p.value, breaks = c(0, 0.05, 0.1, 0.2, 0.5,1))
res_all$term2 = paste(res_all$variable, gsub("pheno[, j]", "",
                                             res_all$term, fixed = TRUE), sep = "_")
```

Categorize the p-values and clean up for plotting

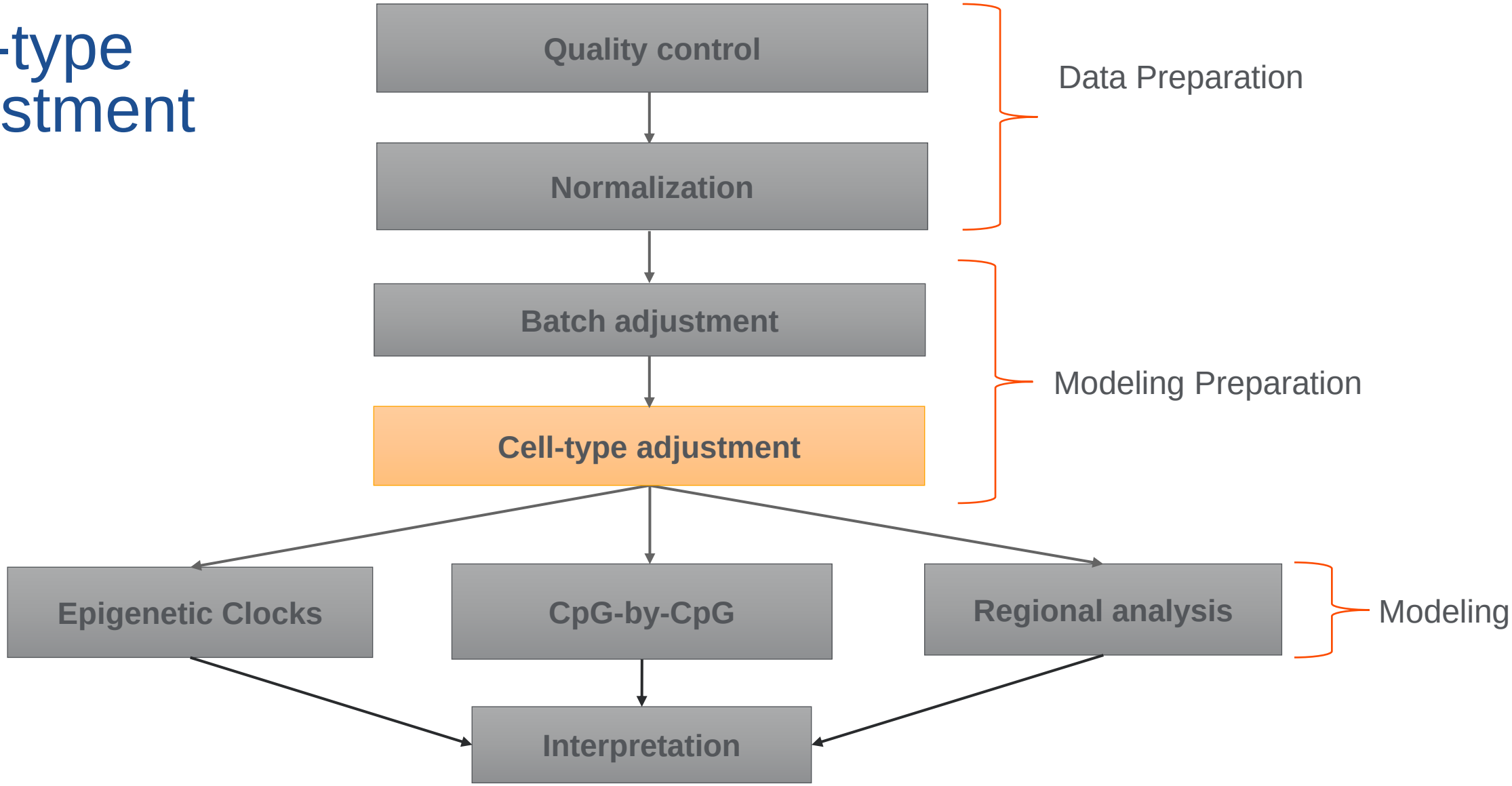
# Plotting the regression results

```
ggplot(res_all, aes(x = as.factor(PC), y = term2, fill = pval)) +
  geom_tile()+
  scale_fill_manual(values = c("darkred", "red", "orange", "yellow", "white"))
```

We can see that chip is a significant source of variability in PCs 3 and 4



# Cell-type adjustment



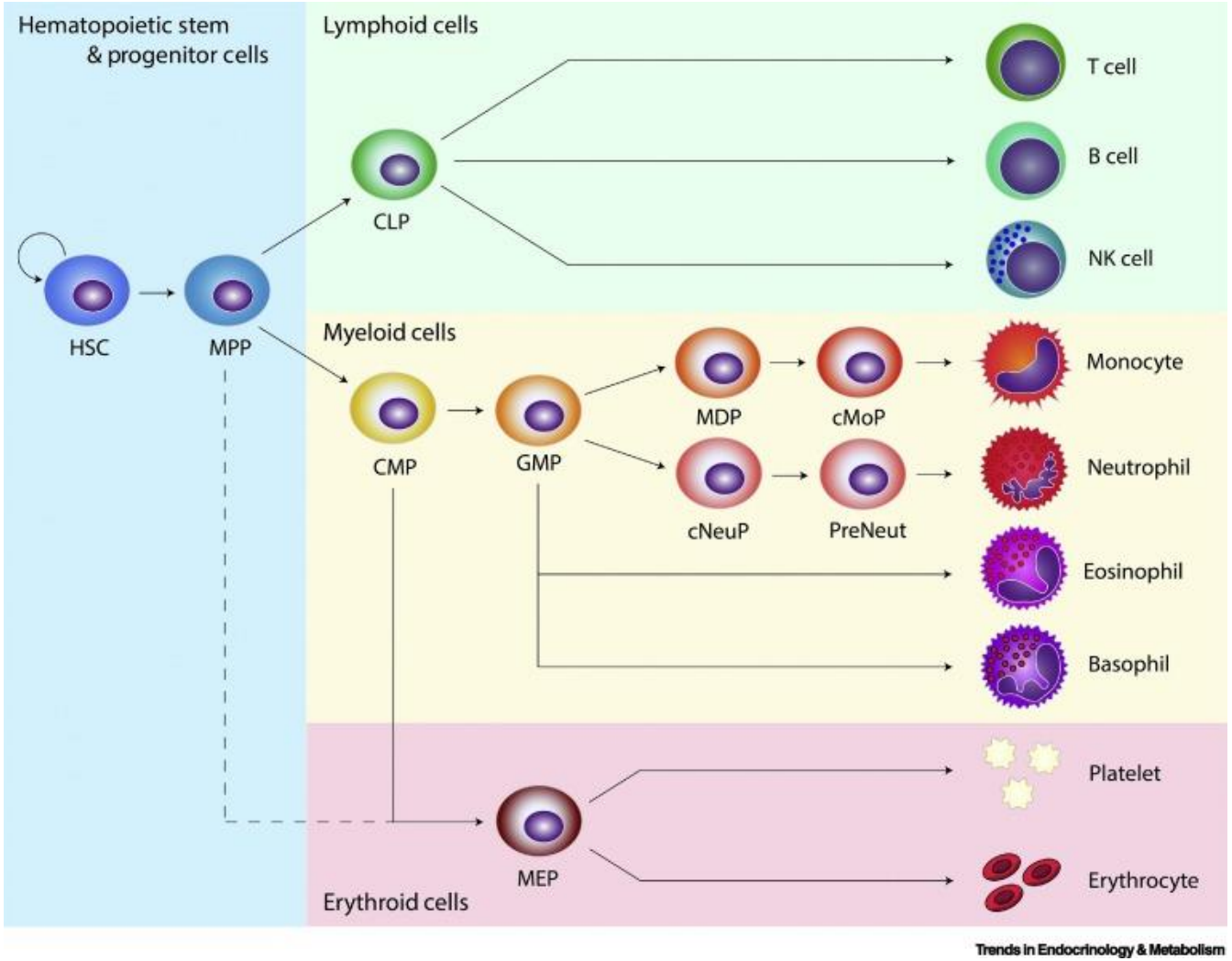
# Cellular Heterogeneity

Most human tissues and biospecimens are composed of many different types of cells.

DNA methylation plays a critical role in cell development and differentiation.

This includes peripheral blood cells, which is where most human population related DNA methylation comes from.

<u>Important Aside</u>: How is your tissue relevant to your research question?

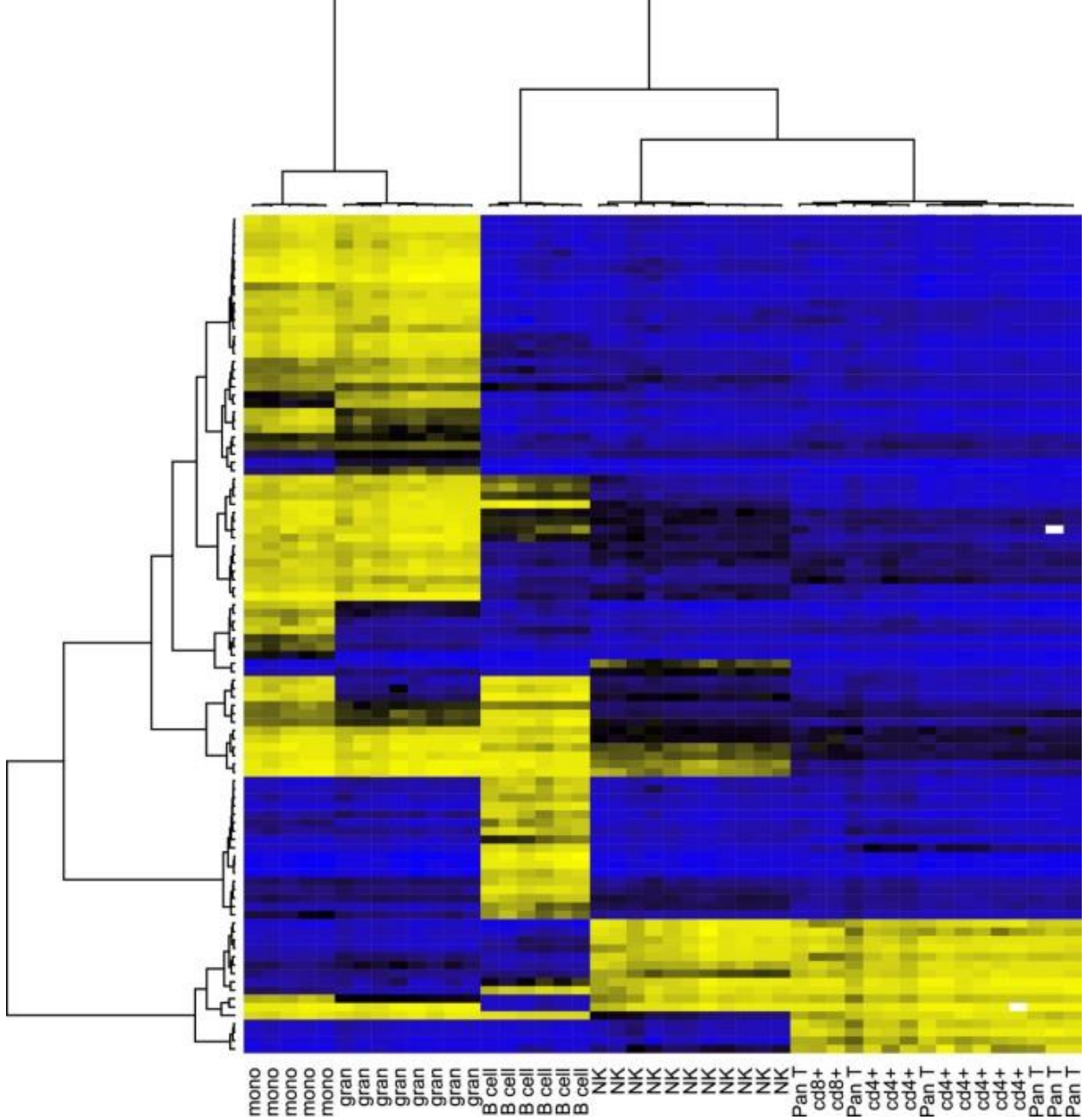



Lee et al., 2020

# Cellular Heterogeneity

Each different cell type has a different pattern of DNA methylation.

These differences are often greater than small impacts from an exposure or disease and can drastically influence results.



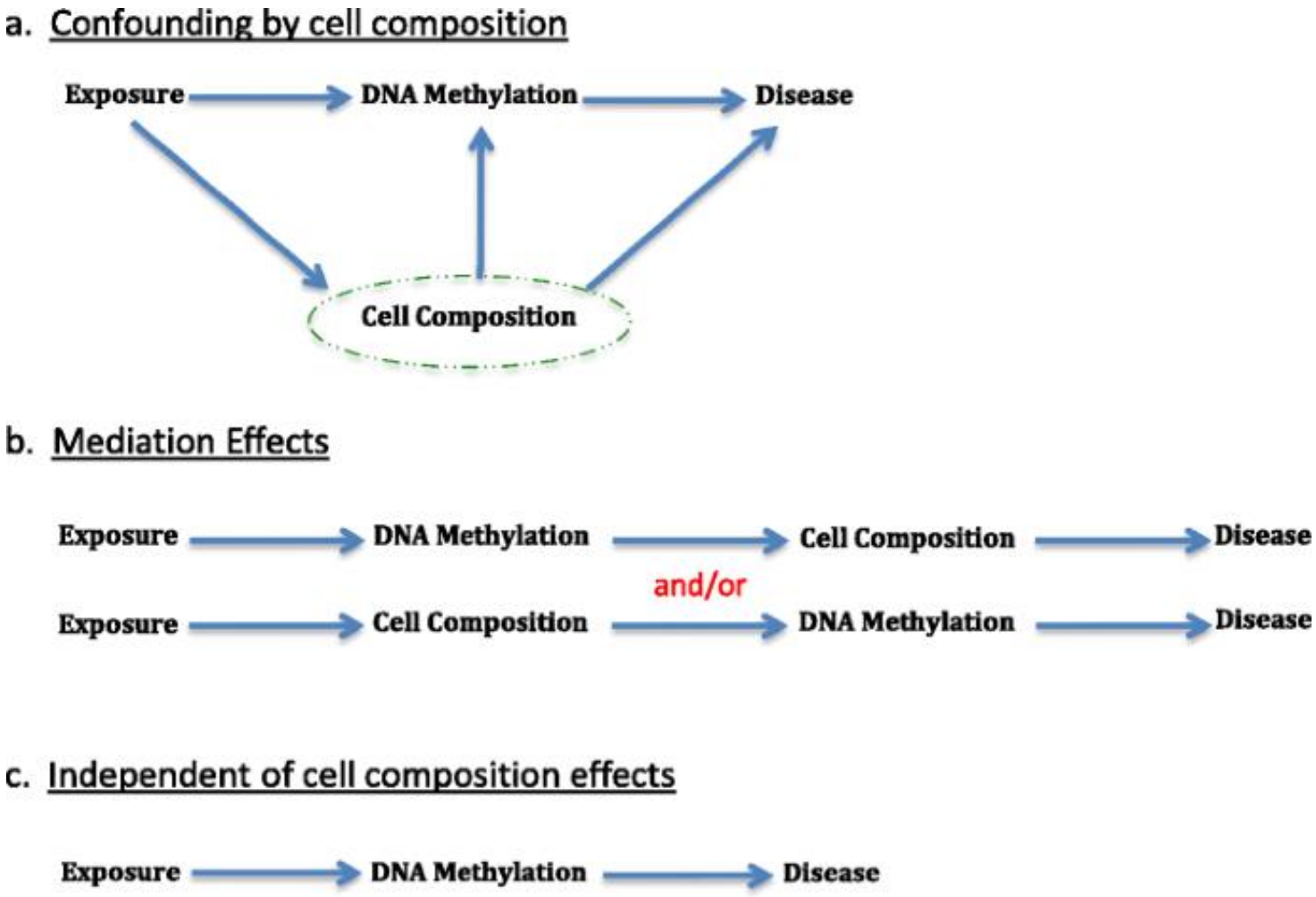

Houseman et al., 2012

# Should we adjust for cell type composition in our analyses?

This depends on your research question and hypotheses.

Cell composition can be a confounder, mediator or nothing at all.

But we often adjust for it since it can have a major impact on results.



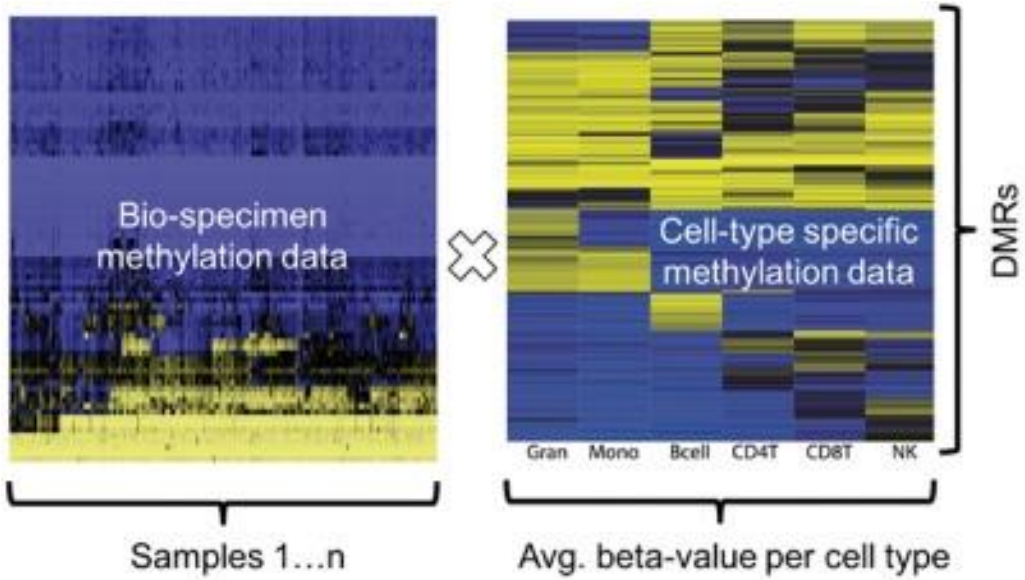

Houseman et al., 2015

# Cell Type Deconvolution

We can use reference panels with known information on cell types to determine the proportion of each major cell type in our samples.

There are reference databases for different sample types:

- Adult whole blood
- Cord blood (includes nucleated RBCs)
- Placenta
- Buccal Cells
- Saliva
- Nasal Cells




**Result:** a matrix of samples with estimated immune cell type proportions

| | Gran | Mono | B-cell | CD4T | CD8T | NK |
|---|---|---|---|---|---|---|
| Samples 1…n | $Gran_1$ | $Mono_1$ | $B\text{-}cell_1$ | $CD4T_1$ | $CD8T_1$ | $NK_1$ |
| | $Gran_2$ | $Mono_2$ | $B\text{-}cell_2$ | $CD4T_2$ | $CD8T_2$ | $NK_2$ |
| | … | … | … | … | … | … |
| | $Gran_n$ | $Mono_n$ | $B\text{-}cell_n$ | $CD4T_n$ | $CD8T_n$ | $NK_n$ |

Immune proportion estimates for samples 1…n

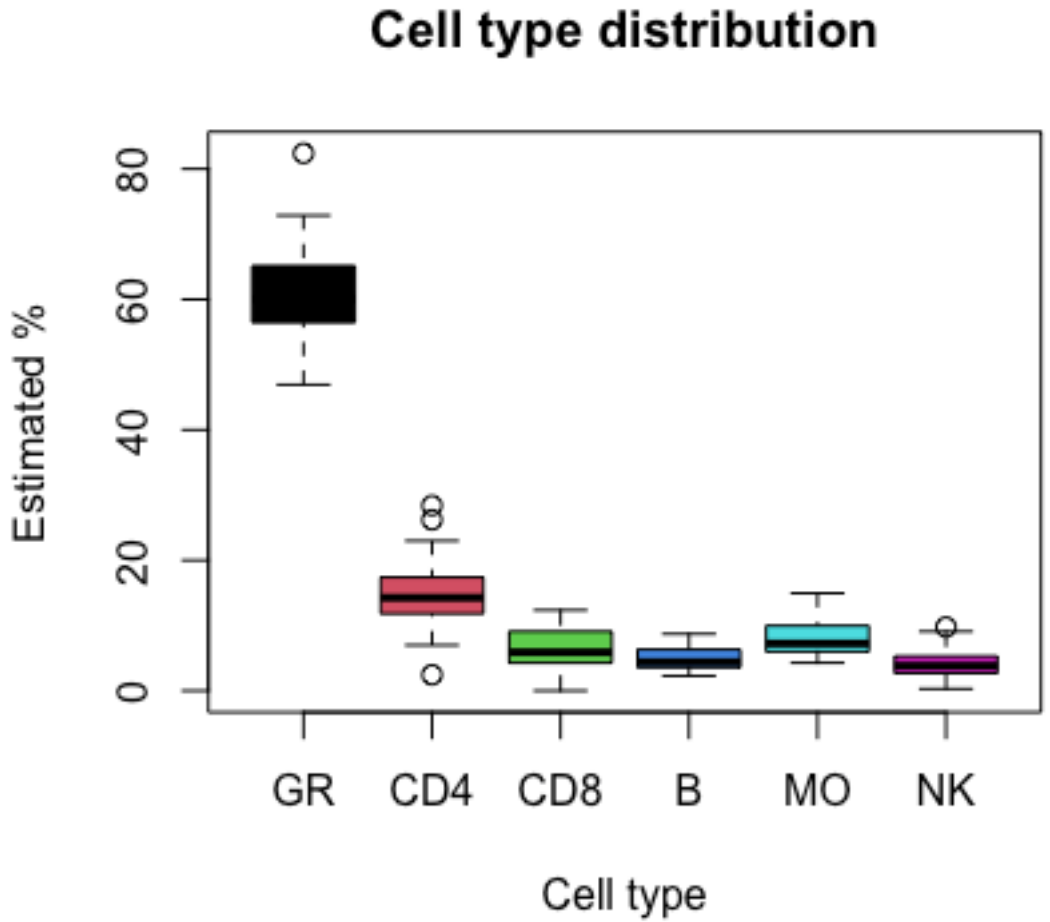
Titus et al., 2017

# Cell type estimation in practice

Requires ewastools. Can also do with minfi but is much slower and requires more memory

```
#' we are using the Reinius reference dataset
cellprop = estimateLC(betas.clean,ref="Reinius")
#' Here are the estimates
knitr::kable(cellprop, digits = 2)

#' note that they are close to summing to 1
summary(rowSums(cellprop))
```

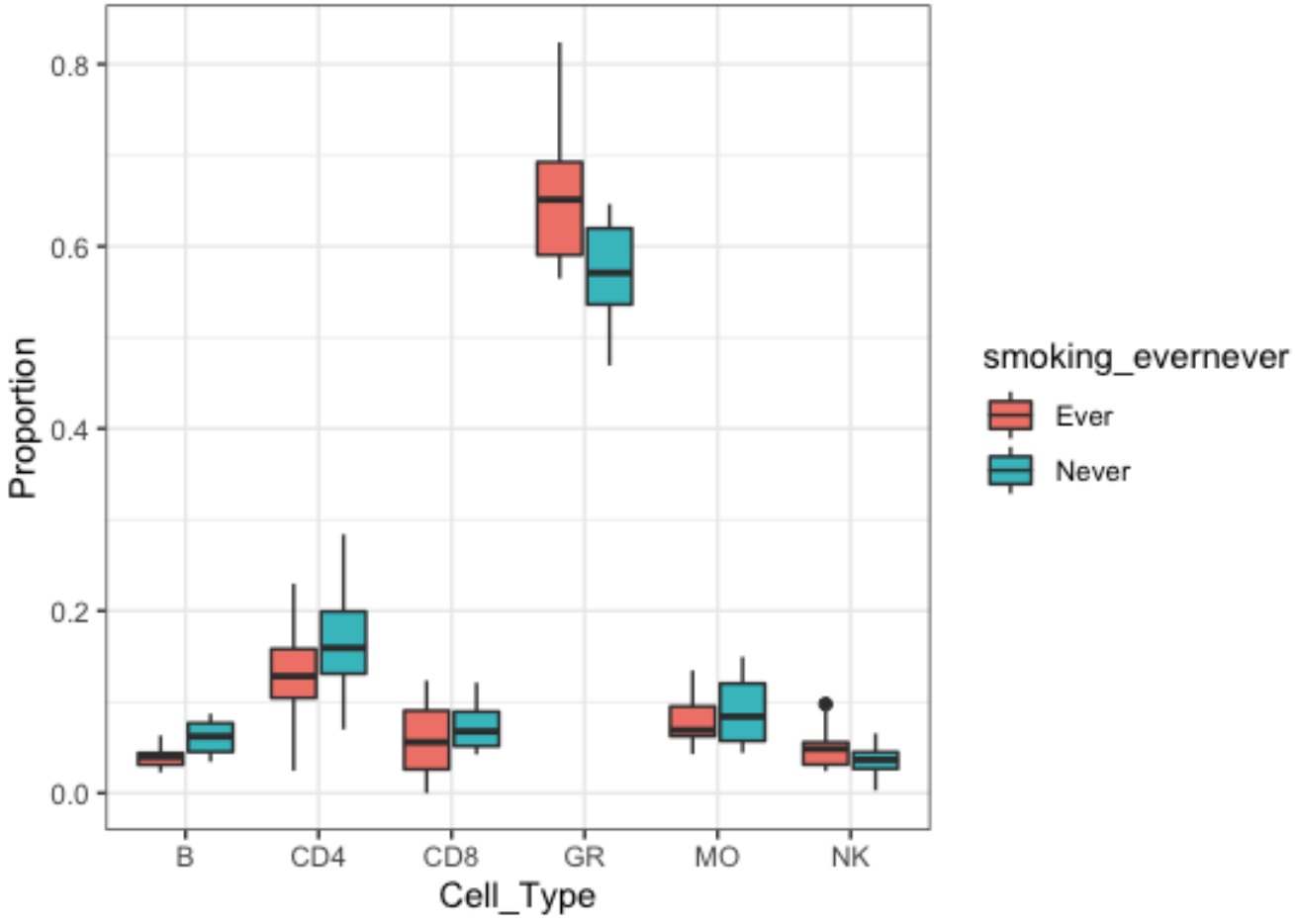
```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.9859  1.0043  1.0075  1.0074  1.0124  1.0226
```



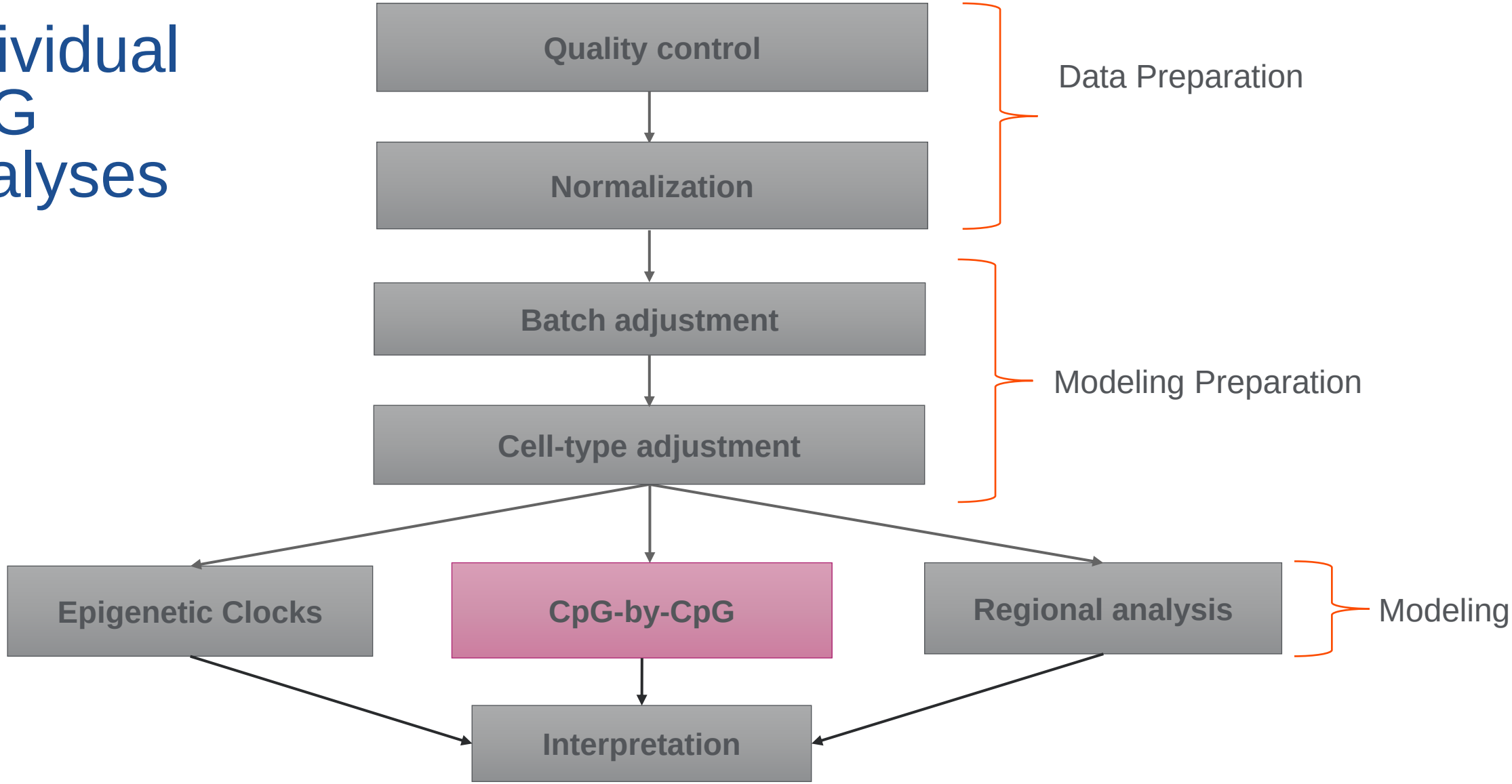

```
#'Distribution of estimated cell types
boxplot(cellprop*100, col=1:ncol(cellprop),
        xlab="Cell type",ylab="Estimated %",main="Cell type distribution")
```

```r
#'Distribution of estimated cell types by smoking status
all.equal(pheno$gsm, row.names(cellprop))
pheno = cbind(pheno, cellprop)
long = gather(pheno, key = "Cell_Type", value = "Proportion", 17:22)

ggplot(long, aes(x = Cell_Type, y = Proportion, fill = smoking_evernever)) +
  geom_boxplot() + theme_bw()
```

# Do cell types associate with our exposure?



# Individual CpG Analyses



# CpG-by-CpG Analyses

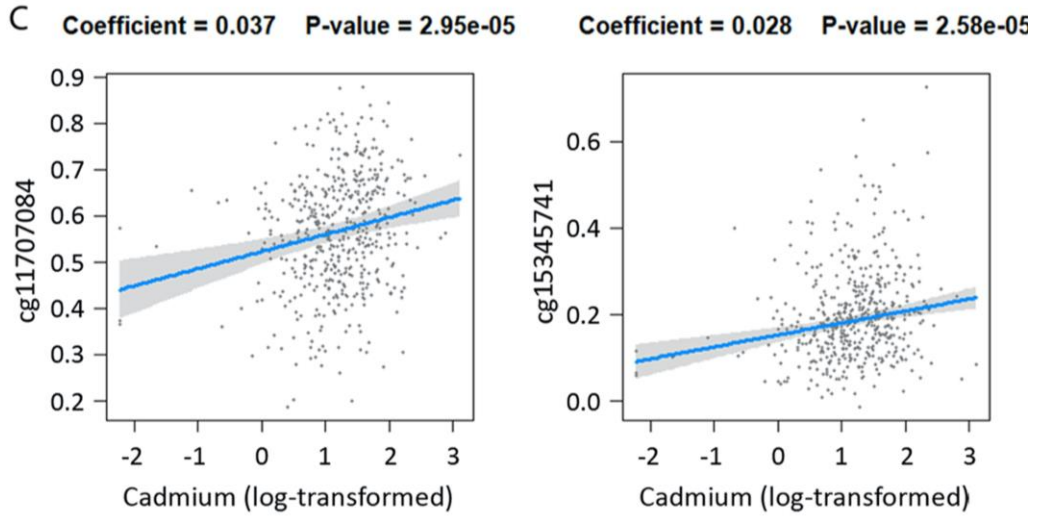
Most common and basic EWAS analysis.

Fit separate adjusted linear models for 450-850 CpGs

Estimate coefficients and p-values for each CpG site.

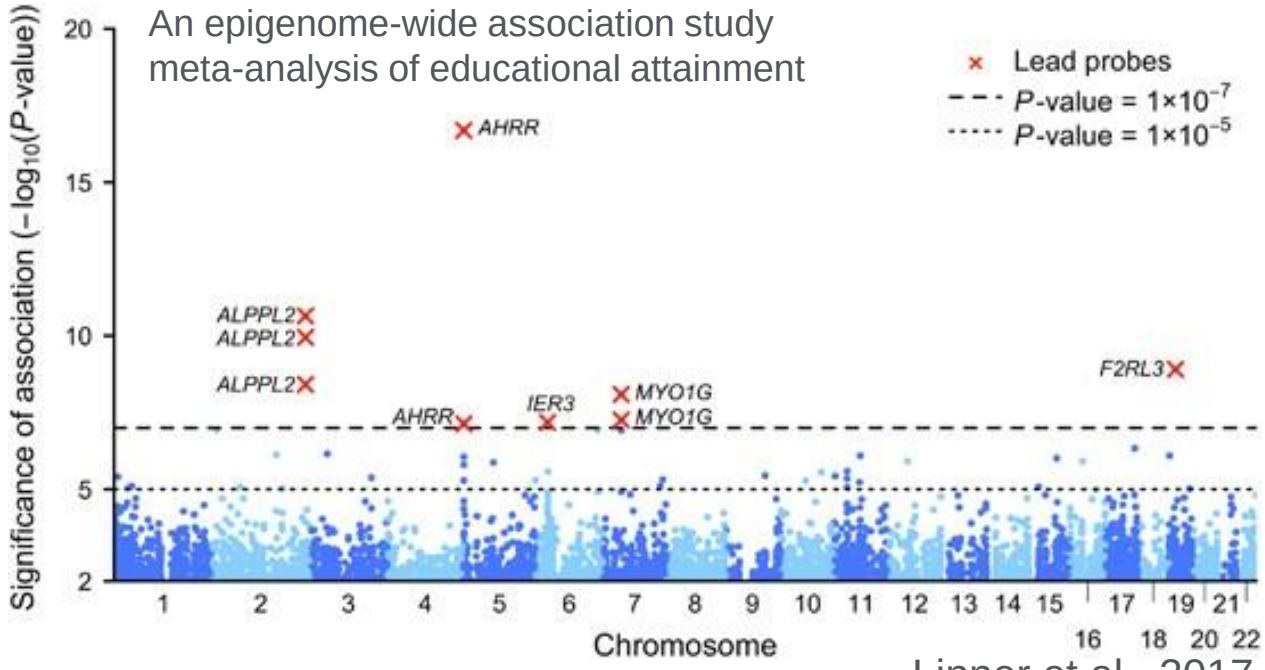
Modeling considerations still apply:

- Must fit model assumptions
- Consider potential relationships between all variables including confounders and mediators




Everson et al., 2018

An epigenome-wide association study meta-analysis of educational attainment




Linner et al., 2017

# Beta-values vs M-values

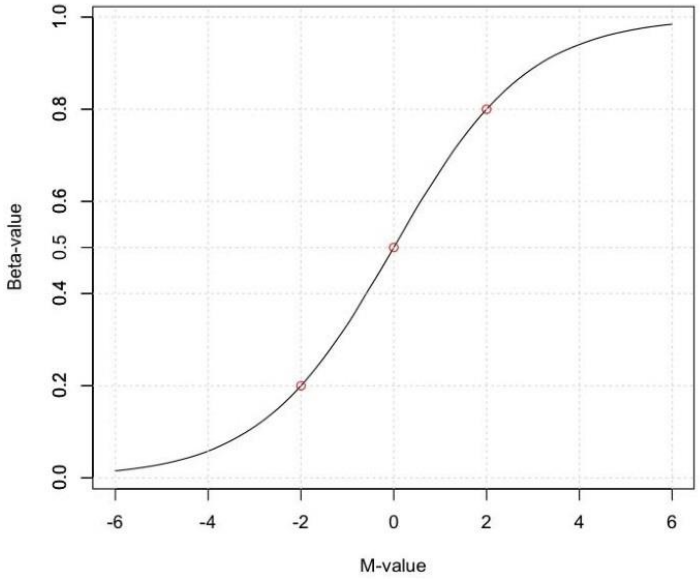
Beta-values: Proportion of methylated probes.

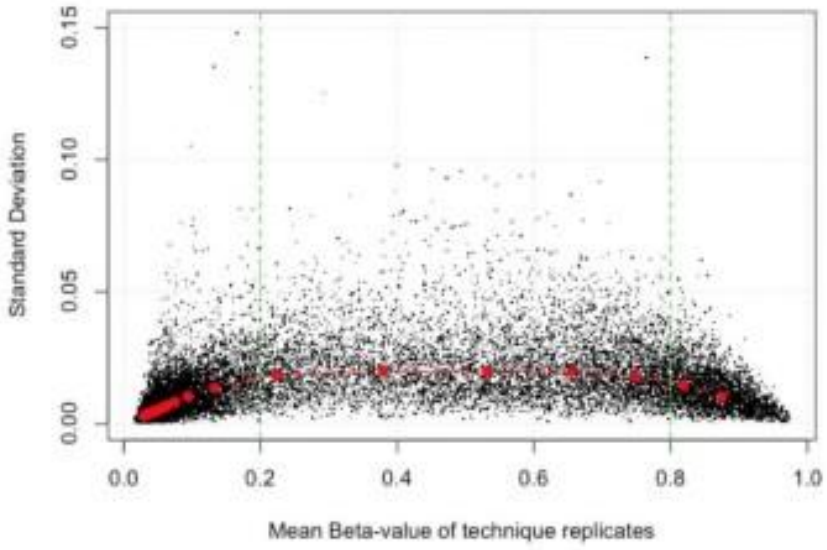
Bound between 0 and 1. Doesn't satisfy modeling assumptions.

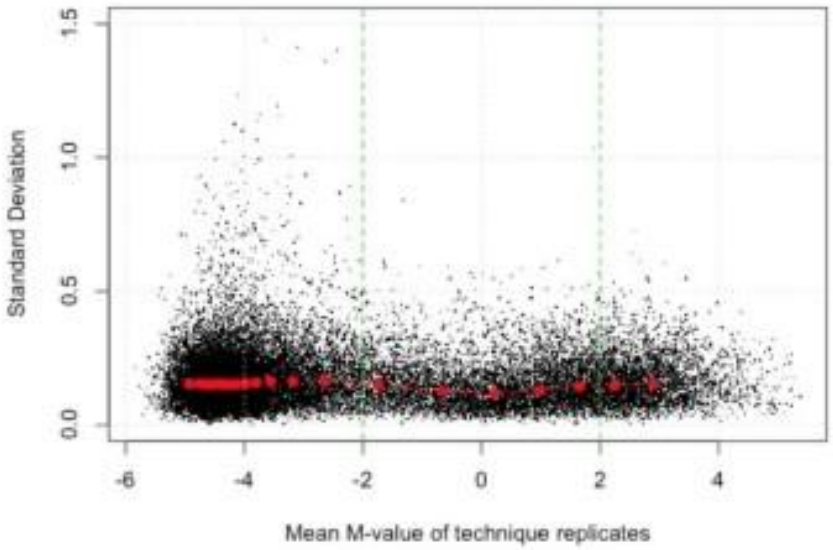
$$\beta_i = \frac{Mi}{Mi + U + 100}$$

M-values: Continuous and semi-homoscedastic value. But uninterpretable.

$$\text{M-value} = \log_2\left(\frac{\beta_i}{1-\beta_i}\right)$$




Du et al., 2010. https://doi.org/10.1186/1471-2105-11-587

# Multiple Testing and Type I Errors

Conducting ~800,000 hypothesis tests.

Leads to concerns about inflation of Type I error.

If we set our alpha at 0.05 then we could expect 42,500 significant sites by chance for the EPIC array.

| | | The truth | |
|---|---|---|---|
| | | $H_0$ True | $H_0$ False |
| Our decision | Don't reject | Correct Decision | **Incorrect : Type II Error** |
| | Reject | **Incorrect : Type I Error** | Correct Decision |

Bonferroni Correction:

Divide alpha by the number of tests conducted.

Often too conservative – restricts power to detect true effects.

False Discovery Rate (FDR) Correction:

Controls the expected proportion of false positives.

FDR is the proportion of significant sites that are false positives.

Most frequently use Benjamini-Hochberg for this: provides a q-value

# Genomic Inflation (*λ*)

Another concept borrowed from genetics.

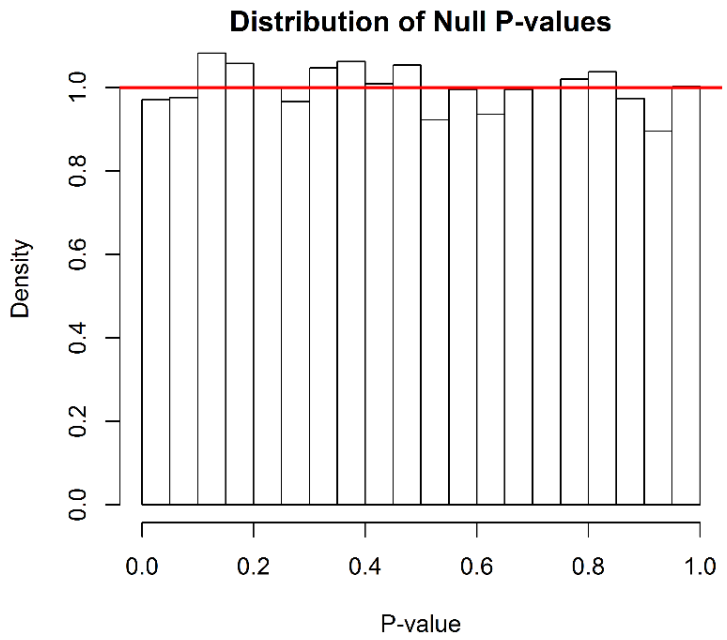
*Lamba (λ): T*he ratio of the median of the empirically observed chi-square test statistics to the expected median under the null.

*λ = 1:* Null

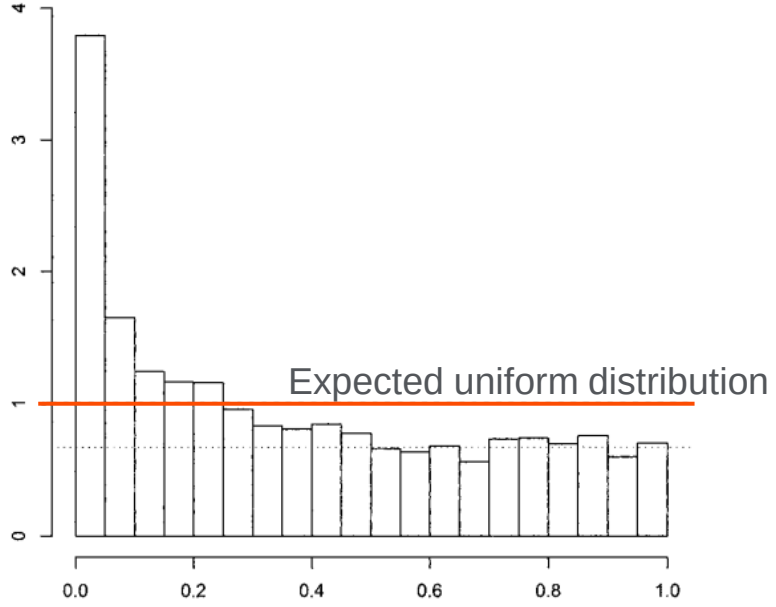
*λ > 1:* Overinflated

*λ < 1:* Underinflated

NULL




OVERINFLATED

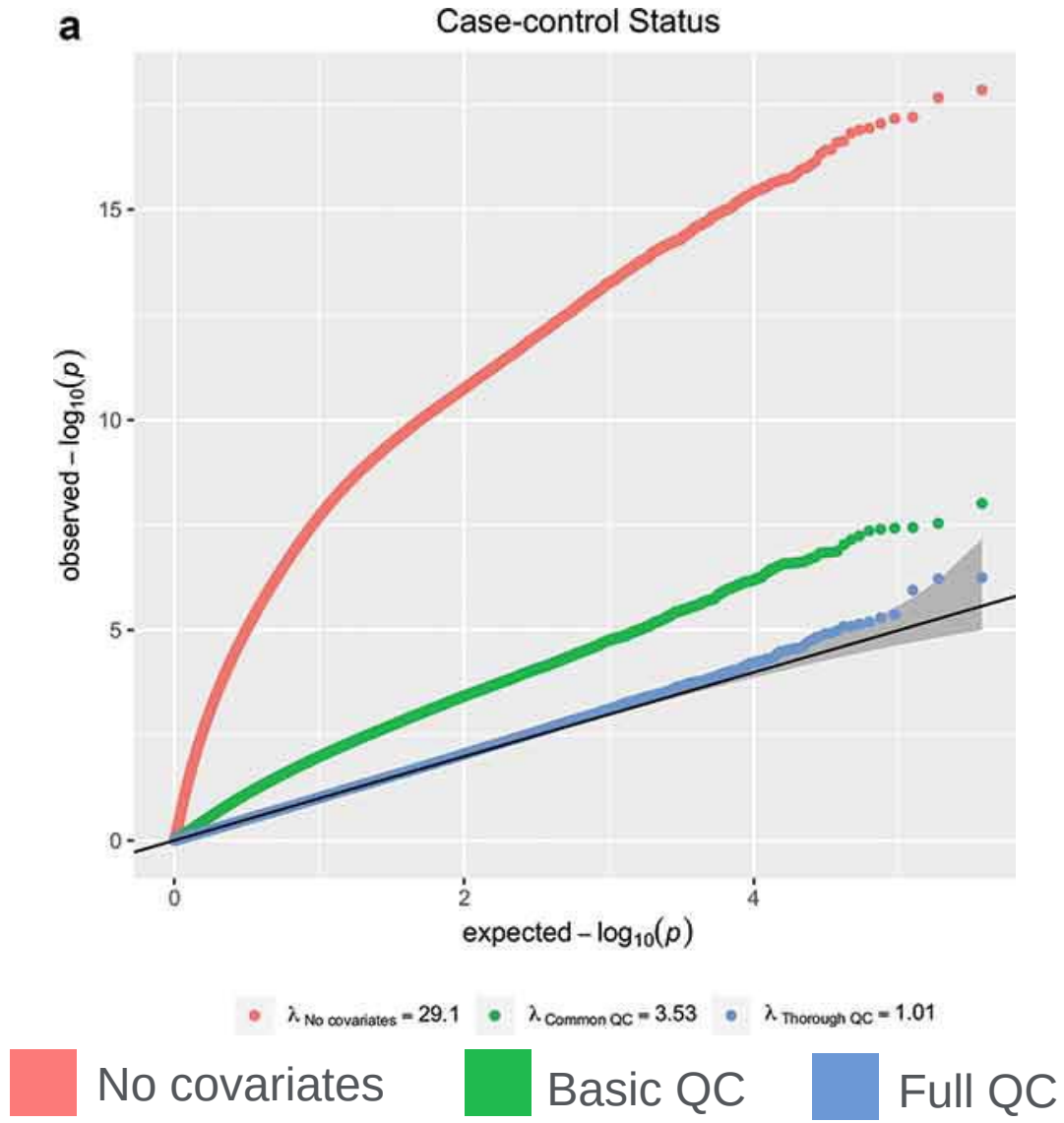


# Q-Q Plots

Graphical examination of genomic inflation.

A high genomic inflation may indicate unaccounted for confounding.

QC, batch and cell type adjustments can reduce genomic inflation.

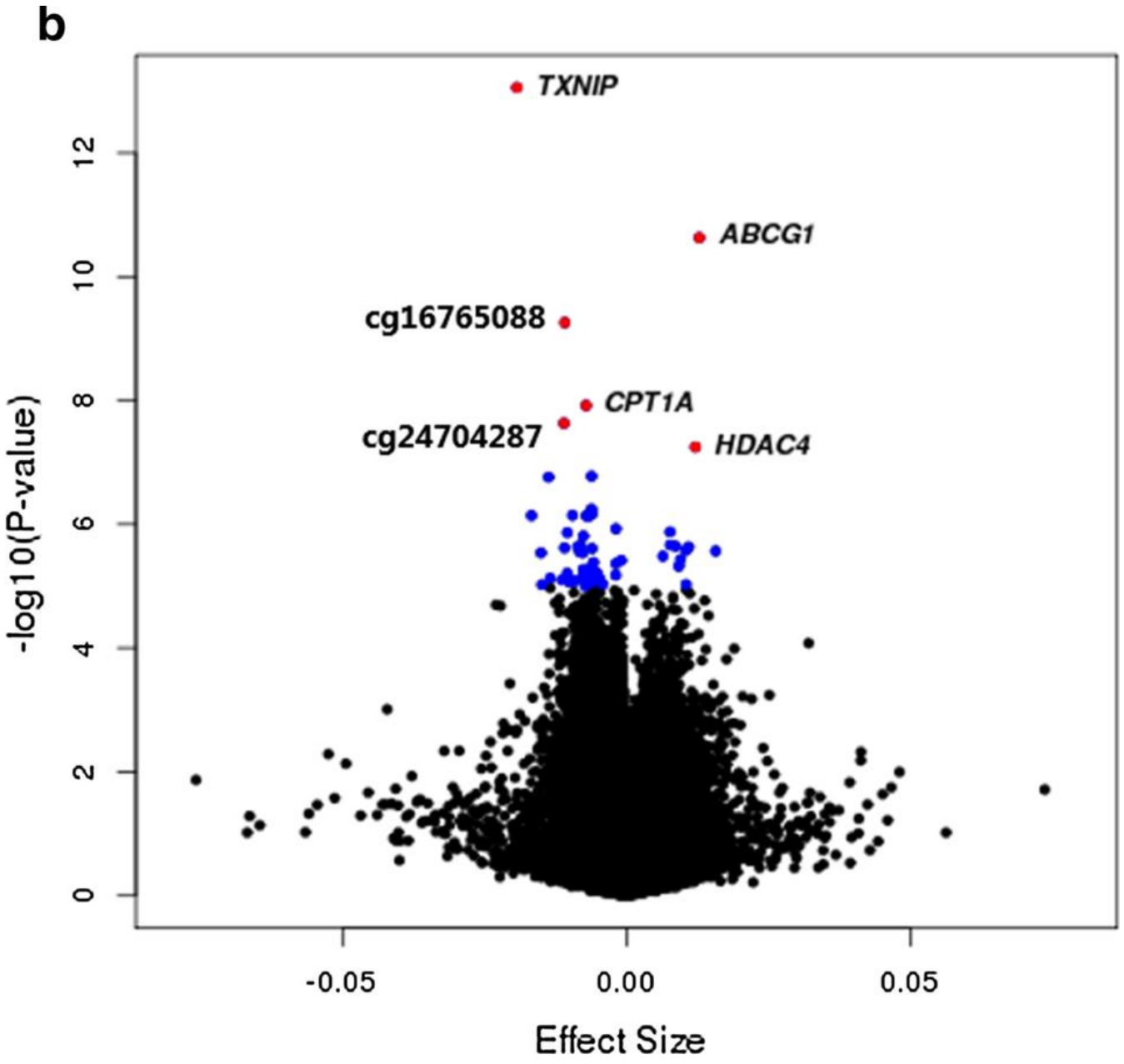



Guintivano et al., 2020

# Visualizing your results: Volcano Plots

Visualize effect estimates and p-values.

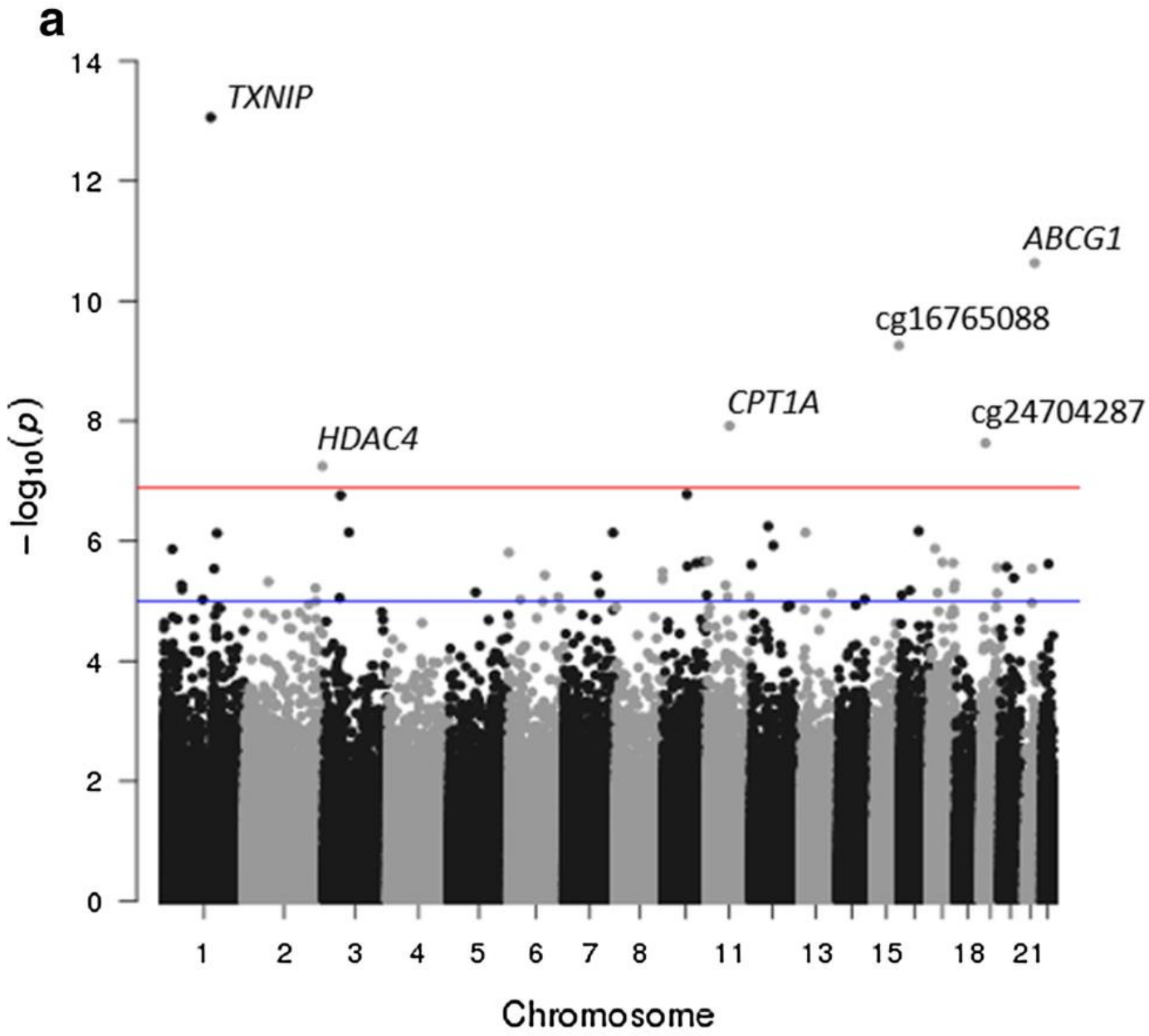
Can see if results are skewed or how many CpGs are significant.



# Visualizing your results: Manhattan Plots

Allows us to see associations that may be spatially related.

Also to make sure that we don't have skewed findings by chromosome or region.



# Important consideration: Outliers

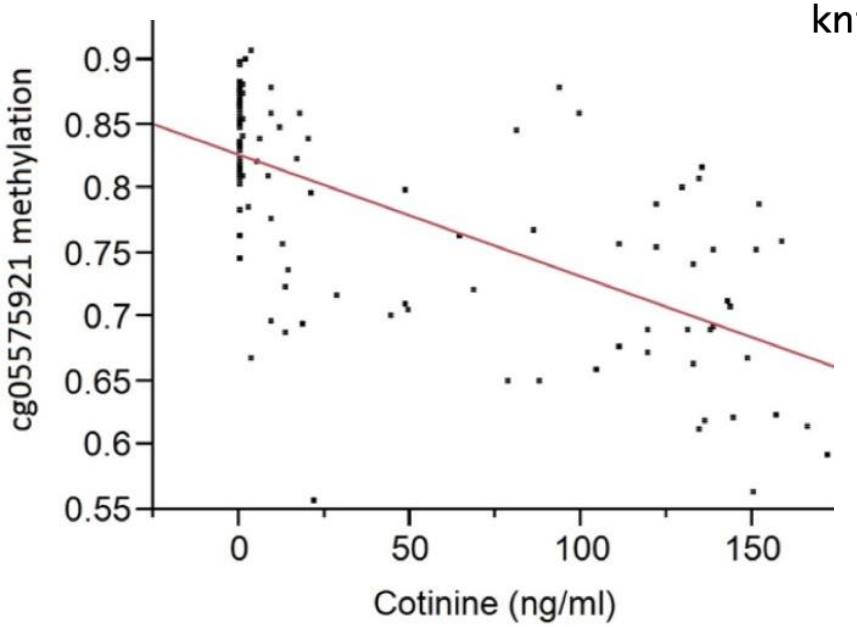
| Modeling Strategy | Pros | Cons | Functions |
|---|---|---|---|
| Ordinary least squares | Fast | Sensitive to outliers | Cpg.assoc |
| Robust regression | Insensitive to outliers | Slow | Rlm (robust package) |
| Limma (robust M-estimation) | Allows a small # of outliers | slower than OLS, faster than rlm | limma |
| Removal of values < or > 3 IQR or SD | Removes outliers | Leaves missing values | |

# Start with a single CpG

```
CpG.name = "cg05575921"
CpG.level <- betas.clean[CpG.name,]
```

```
#' make a smoking dummy variable
pheno$smoke2 <- ifelse(pheno$smoking_evernever == "Ever", 1, 0)
```

CpG within the aryl hydrocarbon receptor repressor.




```
#' difference in methylation between smokers and non-smokers for this CpG
#' some descriptive statistics
knitr::kable(cbind(Min    = round( tapply(CpG.level,pheno$smoke2,min    ),3),
                   Mean   = round( tapply(CpG.level,pheno$smoke2,mean   ),3),
                   Median= round( tapply(CpG.level,pheno$smoke2,median),3),
                   Max    = round( tapply(CpG.level,pheno$smoke2,max    ),3),
                   SD     = round( tapply(CpG.level,pheno$smoke2,sd     ),3),
                   N      = table( pheno$smoke2 )))
```

Clear difference between smokers and nonsmokers

| | Min | Mean | Median | Max | SD | N |
|---|---|---|---|---|---|---|
| 0 | 0.872 | 0.887 | 0.888 | 0.909 | 0.013 | 10 |
| 1 | 0.491 | 0.759 | 0.841 | 0.875 | 0.143 | 11 |

Philibert et al., Clin Epigenetics. 2013

# Run the regressions

Beta values

```
#' linear regression on betas
summary(lm(CpG.level~pheno$smoke2))$
  coefficients[2,c("Estimate", "Pr(>|t|)","Std. Error")]
```

```
   Estimate    Pr(>|t|)  Std. Error
-0.12801478  0.01087274  0.04535242
```

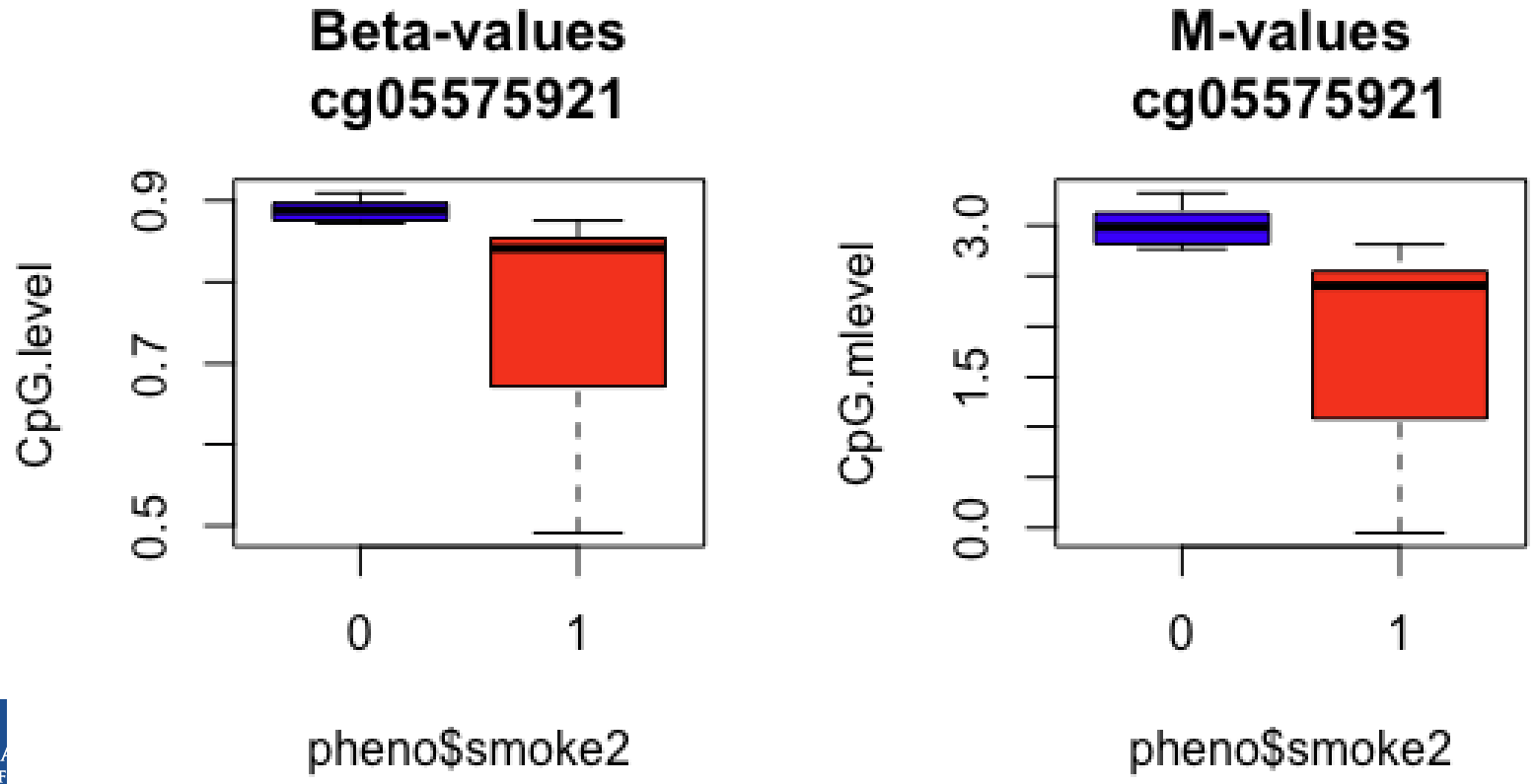
M values

```
#' comparison with m-values
CpG.mlevel <- log2(CpG.level/(1-CpG.level))

#' linear regression on m-values
summary(lm(CpG.mlevel~pheno$smoke2))$
  coefficients[2,c("Estimate", "Pr(>|t|)","Std. Error")]
```

```
    Estimate     Pr(>|t|)   Std. Error
-1.157195095  0.002999132  0.340272865
```

# Visualize the findings

```
par(mfrow=c(1,2))
boxplot(CpG.level  ~ pheno$smoke2, main=paste0("Beta-values\n",CpG.name), col=c("blue","red"))
boxplot(CpG.mlevel ~ pheno$smoke2, main=paste0("M-values\n"   ,CpG.name), col=c("blue","red"))
```



# EWAS and results using CpGassoc

Barfield et al. Bioinformatics 2012 http://www.ncbi.nlm.nih.gov/pubmed/22451269

```
system.time(results1 <- cpg.assoc(betas.clean, pheno$smoke2))
head(cbind(results1$coefficients[,4:5], P.value=results1$results[,3]))
#' and the top hits
head(cbind(results1$coefficients[,4:5], P.value=results1$results[,3])[order(results1$results[,3]),])
#' check with previous result on our selected CpG (running lm without CpGassoc)
cbind(results1$coefficients[,4:5],results1$results[,c(1,3)])[CpG.name,]
summary(lm(CpG.level~pheno$smoke2))$coefficients[2,c("Estimate", "Pr(>|t|)","Std. Error")]
```

What the results look like:

| | effect.size | std.error | P.value |
|---|---|---|---|
| rs10796216 | 0.13709814 | 0.1511038 | 0.3756065 |
| rs715359 | -0.06905560 | 0.1562395 | 0.6634895 |
| rs1040870 | -0.07694116 | 0.1517100 | 0.6178787 |
| rs10936224 | -0.03998900 | 0.1258925 | 0.7542194 |
| rs213028 | 0.20073963 | 0.1334768 | 0.1490411 |
| rs2385226 | -0.07040598 | 0.1363635 | 0.6115941 |

Top hits

| | effect.size | std.error | P.value |
|---|---|---|---|
| cg19089328 | 0.075446334 | 0.0112270936 | 2.013721e-06 |
| cg01222380 | 0.032745190 | 0.0052952767 | 6.090862e-06 |
| cg17108971 | -0.009406299 | 0.0015338409 | 6.785121e-06 |
| cg20849025 | 0.038218478 | 0.0063114730 | 7.984584e-06 |
| cg09906747 | 0.002468012 | 0.0004198609 | 1.164015e-05 |
| cg26540559 | -0.058774394 | 0.0104376064 | 1.981634e-05 |

# Run adjusted models

We can see that there are no FDR significant hits.

```
results2 <- cpg.assoc(
  betas.clean
  ,pheno$smoke2
  ,covariates=pheno[,c("age_sampling", "CD8","CD4","NK","B","MO","GR", "Sentrix_ID")]
)

  The top ten CpG sites were:
       CPG.Labels T.statistic      P.value Holm.sig       FDR   gc.p.value
384931 cg24755163  -12.175625 2.548006e-07    FALSE 0.1103192 3.091175e-07
115137 cg25561762    9.851164 1.823440e-06    FALSE 0.2464881 2.199352e-06
391947 cg13997140    9.675570 2.148791e-06    FALSE 0.2464881 2.590239e-06
50616  cg20364839    9.544839 2.432067e-06    FALSE 0.2464881 2.930363e-06
120425 cg05201784    9.380813 2.846526e-06    FALSE 0.2464881 3.427676e-06
329543 cg17409276    8.609876 6.152515e-06    FALSE 0.4439685 7.384745e-06
195776 cg22935501    8.377838 7.842192e-06    FALSE 0.4850542 9.402286e-06
111887 cg09234599    8.064877 1.096993e-05    FALSE 0.5390736 1.313068e-05
351054 cg26210602   -8.045352 1.120572e-05    FALSE 0.5390736 1.341148e-05
82199  cg00022558    7.775378 1.509769e-05    FALSE 0.6271988 1.804145e-05

  To access results for all  432963  CpG sites use object$results
  or sort(object)$results to obtain results sorted by p-value.

  General info:
    Min.P.Observed Num.Cov fdr.cutoff FDR.method Phenotype chipinfo num.Holm num.fdr
  1   2.548006e-07       8       0.05         BH    smoke2     NULL        0       0

  0 sites were found significant by the Holm method
  0 sites were found significant by BH method
```

# Compare to models with M-values

Set logit.transform = TRUE

Top CpG is the same – But the others are not in the same order.

For instance cg00022558 is 3rd here but was 10th with beta values

```
#'using mvalues
results3 <- cpg.assoc(
  betas.clean
  ,pheno$smoke2
  ,covariates=pheno[,c("age_sampling", "CD8","CD4","NK","B","MO","GR", "Sentrix_ID")]
  ,logit.transform=TRUE
)
```

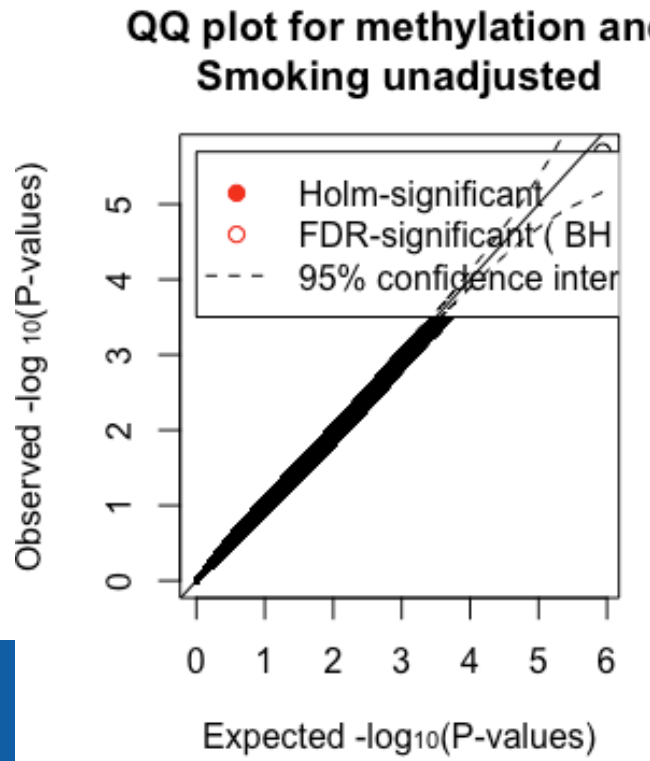
```
The top ten CpG sites were:
       CPG.Labels T.statistic      P.value Holm.sig       FDR   gc.p.value
384931 cg24755163  -11.037629 6.386197e-07    FALSE 0.1777160 6.714800e-07
391947 cg13997140   10.743140 8.209292e-07    FALSE 0.1777160 8.630036e-07
82199  cg00022558   10.222123 1.299173e-06    FALSE 0.1874979 1.365241e-06
120425 cg05201784    9.369009 2.879193e-06    FALSE 0.3116460 3.023363e-06
50616  cg20364839    8.695980 5.630039e-06    FALSE 0.3704131 5.907678e-06
115137 cg25561762    8.665893 5.806901e-06    FALSE 0.3704131 6.093044e-06
195776 cg22935501    8.635986 5.988714e-06    FALSE 0.3704131 6.283591e-06
351054 cg26210602   -8.251880 8.965912e-06    FALSE 0.4852385 9.402782e-06
9925   cg03892551    8.055157 1.108663e-05    FALSE 0.5333443 1.162365e-05
51292  cg17108971   -7.854350 1.382595e-05    FALSE 0.5986123 1.449136e-05

To access results for all  432963  CpG sites use object$results
or sort(object)$results to obtain results sorted by p-value.

General info:
  Min.P.Observed Num.Cov fdr.cutoff FDR.method Phenotype chipinfo num.Holm num.fdr
1   6.386197e-07       8       0.05         BH    smoke2     NULL        0       0
```

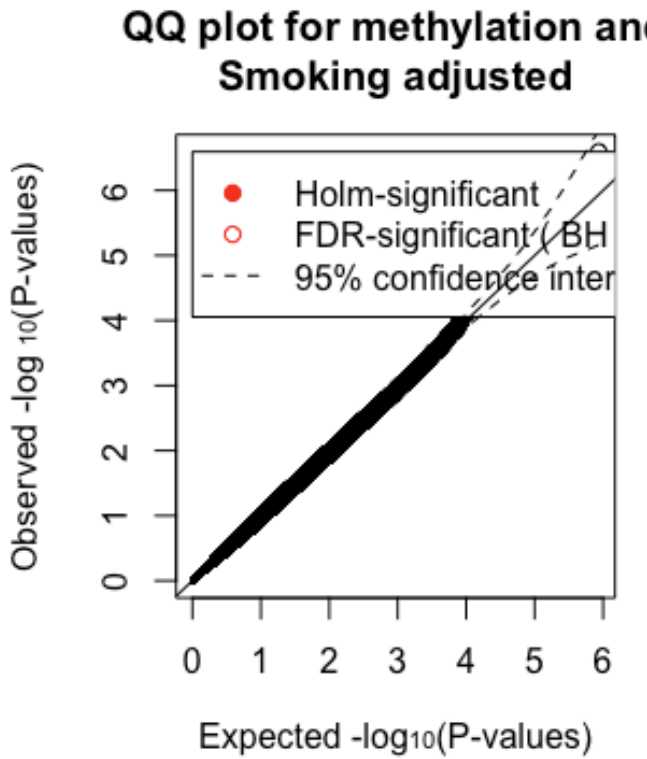
# Examine the genomic inflation

```
par(mfrow=c(1,2))
plot(results1, main="QQ plot for association between methylation and Smoking\nadjusted for cell proportions")
plot(results2, main="QQ plot for association between (mvals) methylation and Smoking\nadjusted for cell proportions")

lambda <- function(p) median(qchisq(p, df=1, lower.tail=FALSE),
                             na.rm=TRUE) / qchisq(0.5, df=1)

#' Lambda before cell type adjustment
lambda(results1$results[,3])
#' Lambda after cell type adjustment
lambda(results2$results[,3])
```

```
[1] 1.036742
[1] 0.9685715
```

No evidence for genomic inflation – underinflation indicates lack of power



# Map to genomic annotations

```r
#' Extract the CpGs with p < 0.001 for later GO analysis
sigCpGs <- results2$results$CPG.Labels[results2$results$P.value < 0.001]
allCpGs <- results2$results$CPG.Labels
```

```r
#' Read in the illumina annotations
IlluminaAnnot <- readRDS("IlluminaAnnot.rds")

#' Restrict to good quality probes and order data frames
IlluminaAnnot <- IlluminaAnnot[IlluminaAnnot$Name %in% allCpGs,]
IlluminaAnnot <- IlluminaAnnot[match(allCpGs, IlluminaAnnot$Name),]

datamanhat <- data.frame(CpG=results2$results[,1],Chr=as.character(IlluminaAnnot$chr),
                         Mapinfo=IlluminaAnnot$pos, UCSC_RefGene_Name=IlluminaAnnot$UCSC_RefGene_Name,
                         Pval=results2$results[,3], Eff.Size = results2$coefficients[,4],
                         Std.Error = results2$coefficients[,5])
```

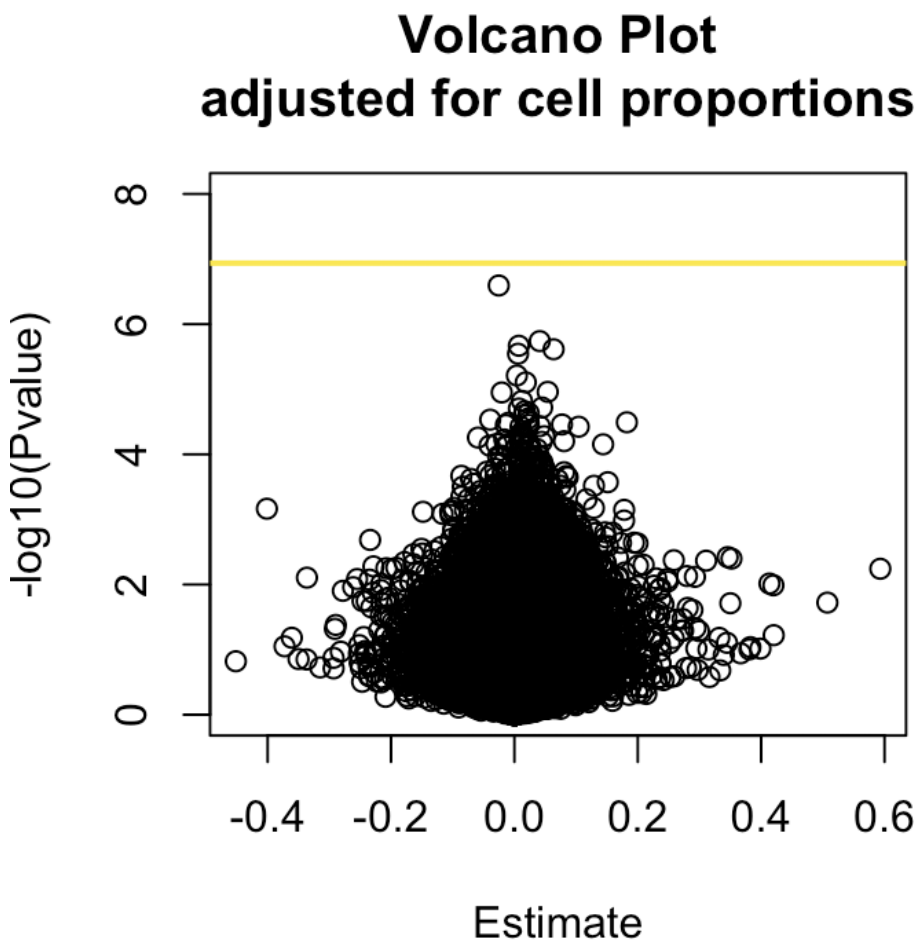
Info on chromosome, genomic location, and nearest gene

| | CpG | Chr | Mapinfo | UCSC_RefGene_Name | Pval | Eff.Size | Std.Error |
|---|---|---|---|---|---|---|---|
| 384931 | cg24755163 | chr7 | 26416987 | | 2.548006e-07 | -0.025380055 | 0.0020844971 |
| 115137 | cg25561762 | chr13 | 20876028 | | 1.823440e-06 | 0.041071803 | 0.0041692334 |
| 391947 | cg13997140 | chr7 | 86849809 | C7orf23 | 2.148791e-06 | 0.007223002 | 0.0007465195 |
| 50616 | cg20364839 | chr10 | 54075528 | DKK1 | 2.432067e-06 | 0.063543767 | 0.0066573951 |
| 120425 | cg05201784 | chr13 | 74709101 | KLF12 | 2.846526e-06 | 0.005903483 | 0.0006293147 |
| 329543 | cg17409276 | chr5 | 60241424 | NDUFAF2;ERCC8 | 6.152515e-06 | 0.003988757 | 0.0004632770 |
| 195776 | cg22935501 | chr17 | 78234799 | RNF213;RNF213 | 7.842192e-06 | 0.018412046 | 0.0021977084 |

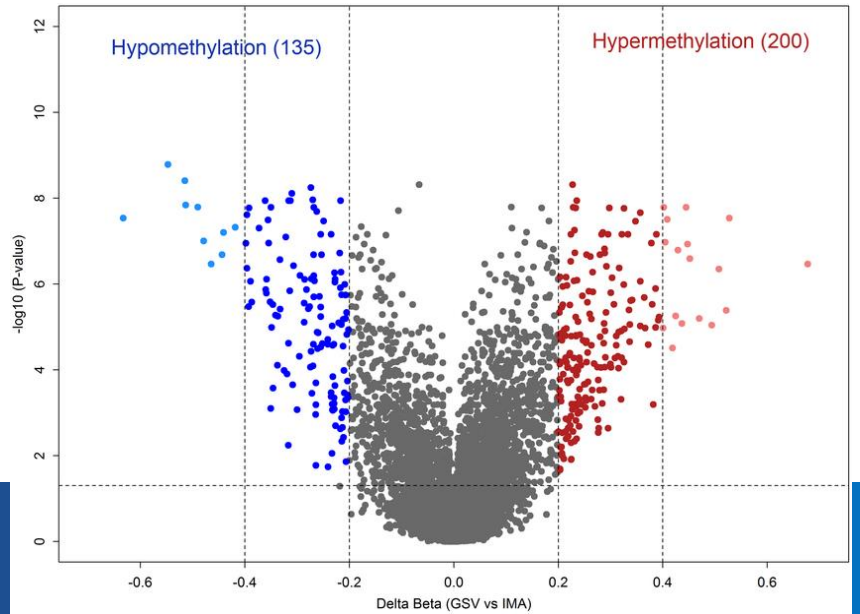
```
par(mfrow=c(1,1))
plot(results2$coefficients[,4],-log10(results2$results[,3]),
     xlab="Estimate", ylab="-log10(Pvalue)", main="Volcano Plot\nadjusted for cell proportions",ylim=c(0,8))
#Bonferroni threshold & FDR threshold
abline(h = -log10(0.05/(nCpG)), lty=1, col="#FDE725FF", lwd=2)
```

# Volcano Plots

Can clearly see that there is little significance



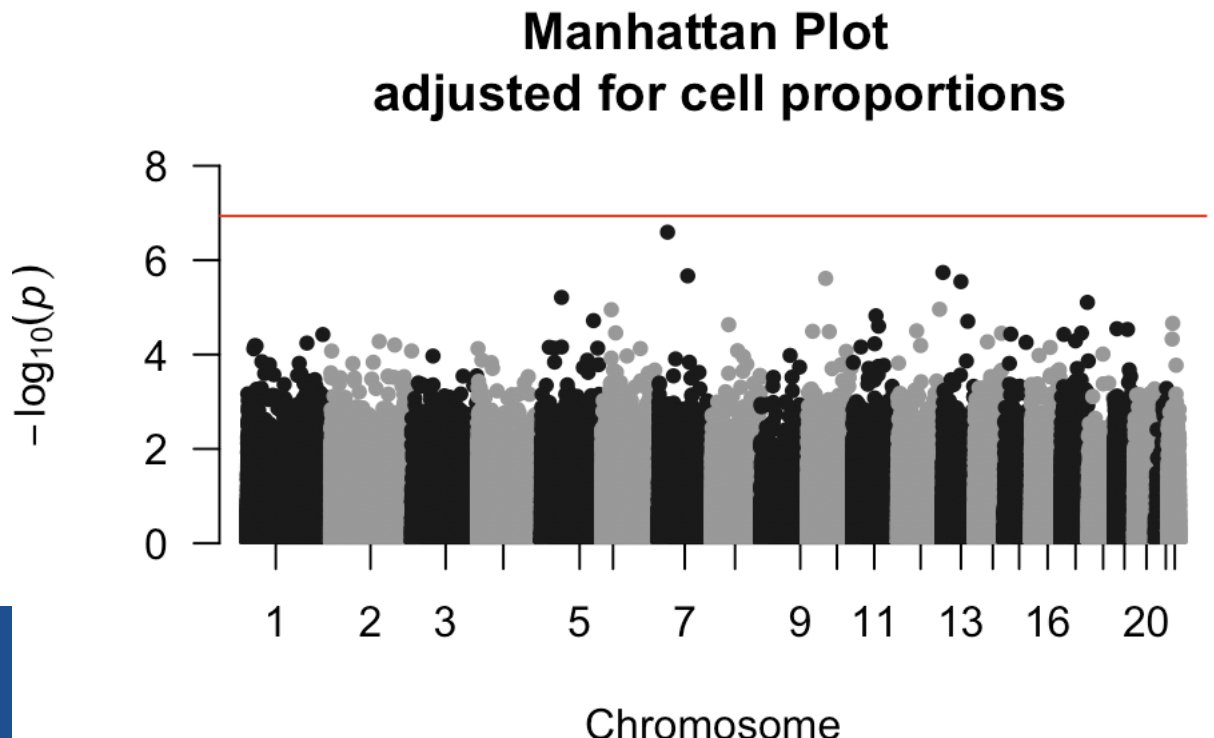

Plot would ideally look like this



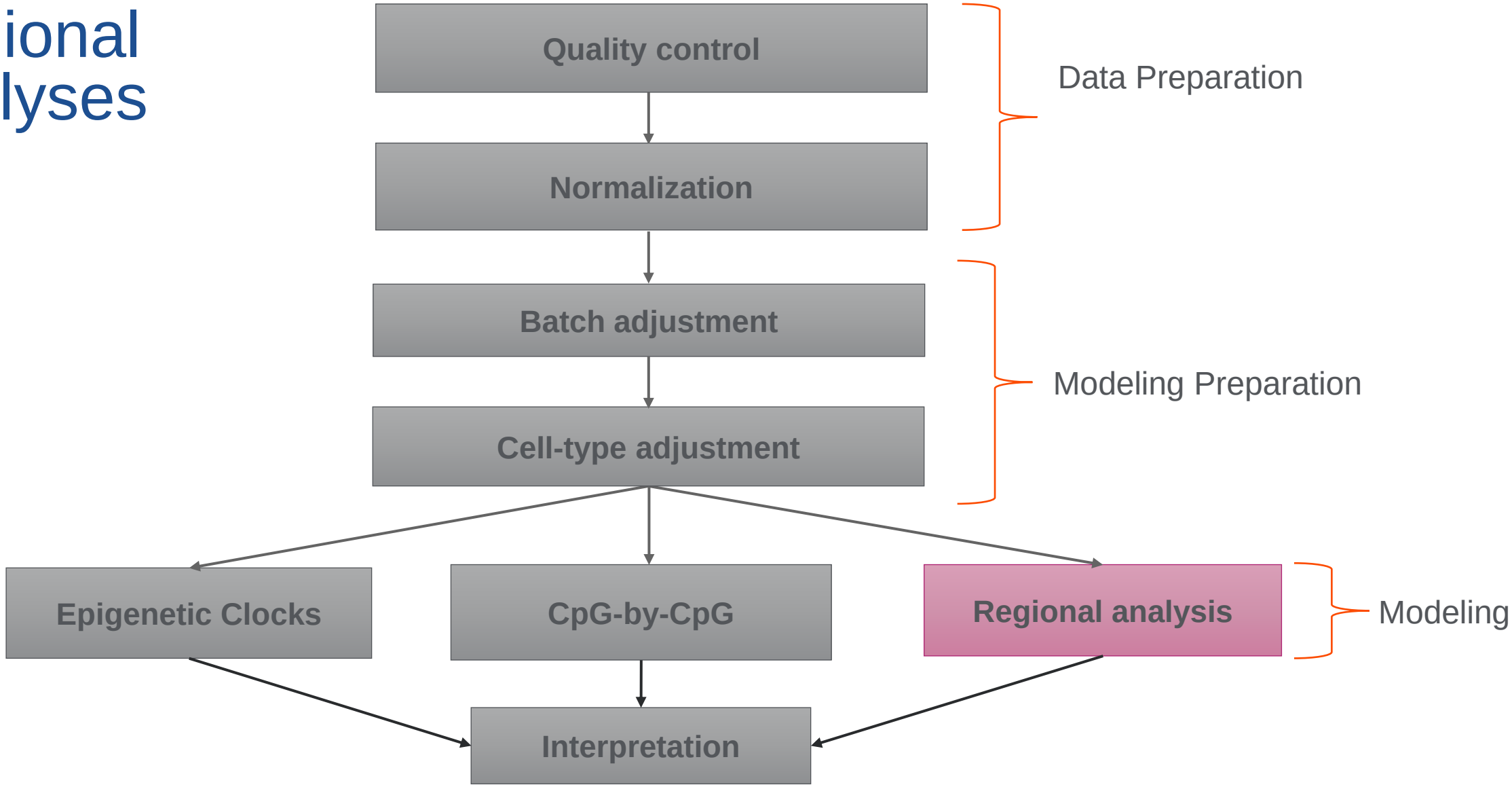

Nazarenko et al., 2015

```r
#'## Manhattan plot for cell-type adjusted EWAS
#' Reformat the variable Chr (so we can simplify and use a numeric x-axis)
datamanhat <- subset(datamanhat, !is.na(Chr))
datamanhat$Chr <- as.numeric(sub("chr","",datamanhat$Chr))

#' the function manhattan needs data.frame including CpG, Chr, MapInfo and Pvalues
manhattan(datamanhat,"Chr","Mapinfo", "Pval", "CpG",
          genomewideline = -log10(0.05/(nCpG)), suggestiveline = FALSE,
          main = "Manhattan Plot \n adjusted for cell proportions",ylim=c(0,8))
```

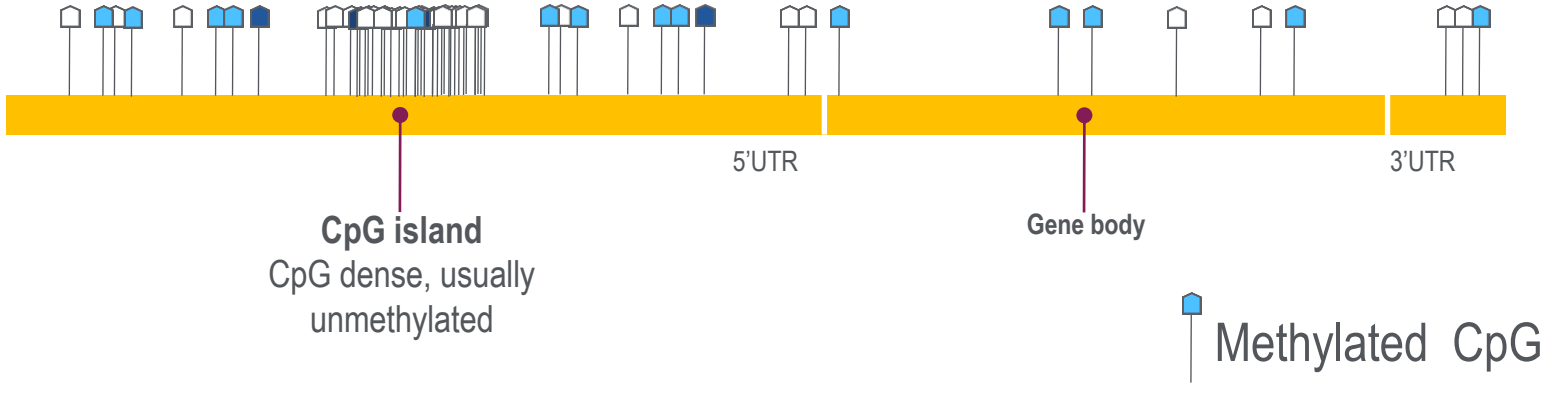
# Manhattan Plots



# Regional Analyses



# Regional Analyses




What is the significance of a change in methylation of a single CpG?

Many CpGs are located near each other in a CpG island or gene body.

Individual CpG analyses assume independent tests – but many CpGs are correlated

Methods are available to model groups of CpGs

- Bump-hunting (Jaffe, et al. *Int J Epidemiol*, 2012): uses smoothed methylation values to detect DMRs
- Comb-P (Pedersen, et al. *Bioinformatics*, 2012): Finds regions of enrichment from spatially assigned *P* values
- **DMRcate** (Peters, et al. *Epigenetics Chromatin*, 2015)

# DMRcate: Steps

DMRcate uses a default value of $\lambda$=1,000 bp, as do *Bumphunter* and *Pro be Lasso*.

1. Apply standard linear modelling to the data using exposures, outcomes, and covariates.
2. Apply Gaussian smoothing to the resulting per-CpG-site test statistics using a given bandwidth, $\lambda$.
3. Model the smoothed test statistics.
4. Compute *P* values based on this model, adjust for multiple comparisons and select threshold.
5. Agglomerate nearby significant CpG sites, again using $\lambda$.

# DMRcate: Pros and Cons

Advantages:

Minimizes multiple testing

Can scale with technology

Very fast

Complementary to linear adjusted models (limma or linear regression)

DMRs are based on effect size, not direction of effect

Disadvantages:

Difficult to make clusters when CpG coverage is sparse

Assumes our definition of clusters is:

1. Meaningful
2. Correct
3. CpGs within behaves similarly

Variation between datasets

Susceptible to overinflation

# DMRcate

Define the model

```
model <- model.matrix( ~smoke2+age_sampling+CD4+CD8+NK+B+MO+GR+Sentrix_ID,data=pheno)

myannotation <- cpg.annotate("array", na.omit(betas.clean),
                             analysis.type="differential",arraytype="450K",
                             what="Beta",design=model, coef=2)
```

Annotate for selected CpGs

```
#'Regions are now agglomerated from groups of significant probes
#'where the distance to the next consecutive probe is less than lambda nucleotides away
dmrcoutput.smoking <- dmrcate(myannotation, lambda=1000, C=2, pcutoff = 0.0001)
```

This is the average distance

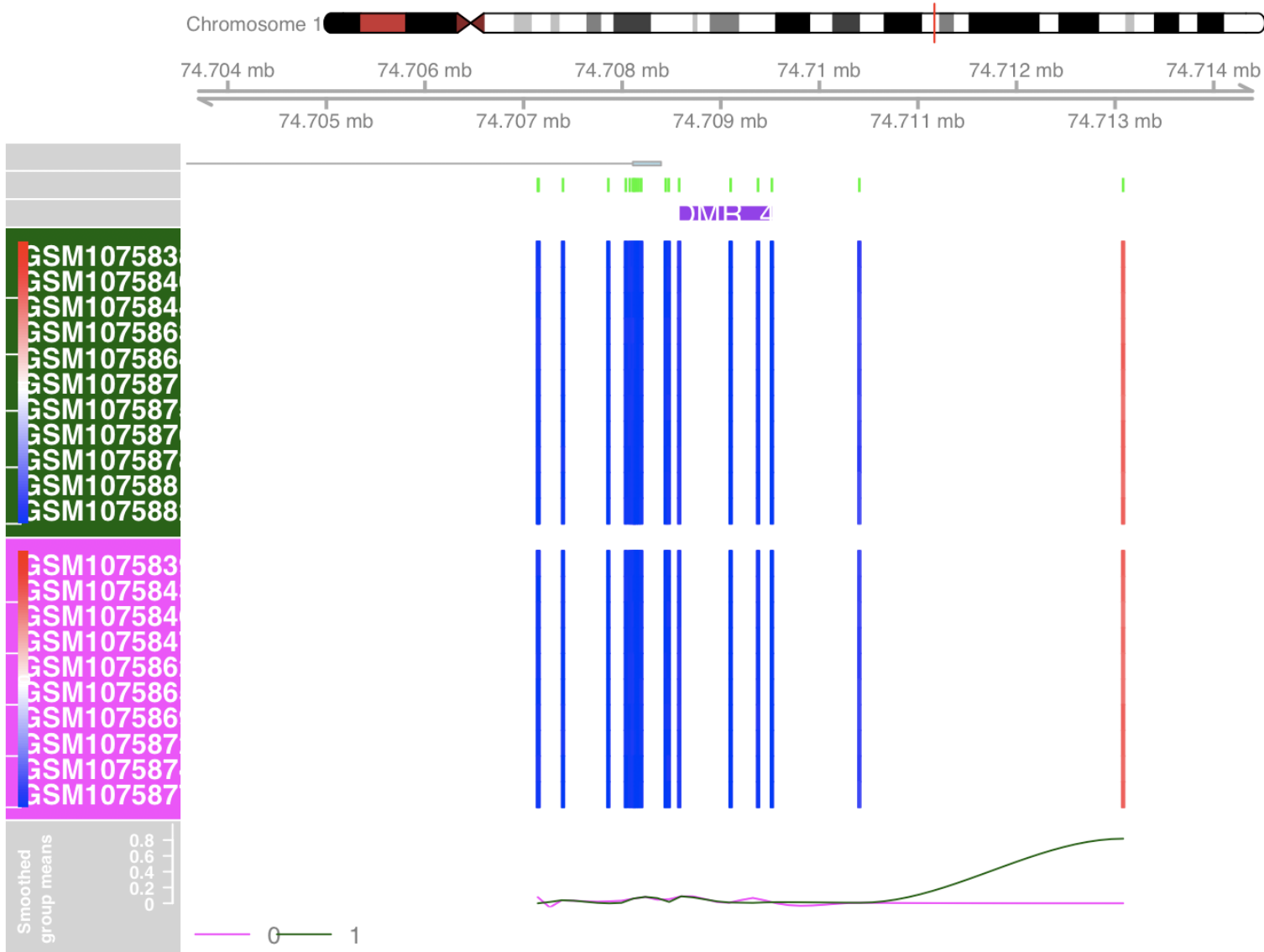
Here we set a more liberal p-value cutoff to have hits for example purposes

# Look at the results

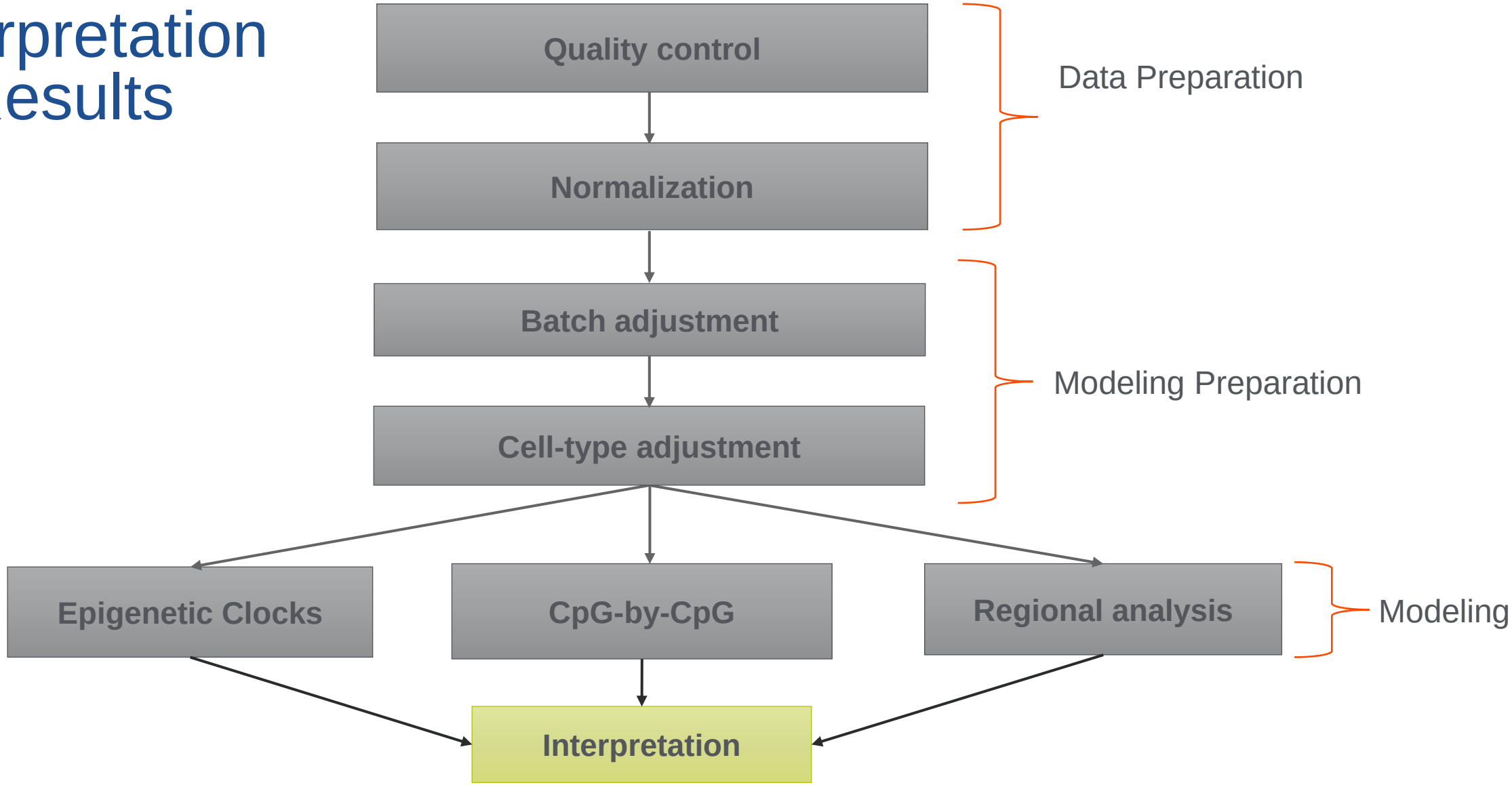
```
GRanges object with 66 ranges and 8 metadata columns:
       seqnames              ranges strand |   no.cpgs min_smoothed_fdr  Stouffer     HMFDR
          <Rle>           <IRanges>  <Rle> | <integer>        <numeric> <numeric> <numeric>
   [1]    chr11     1283875-1283946      * |         2      1.53972e-17  0.993473  0.430288
   [2]     chr7   26416735-26416987      * |         3      9.66354e-11  0.999995  0.531155
   [3]    chr10       616959-617105      * |         3      2.27770e-08  0.999995  0.531155
   [4]    chr13   74708579-74709519      * |         4      2.37825e-13  1.000000  0.601677
   [5]    chr12 133307618-133307702      * |         3      3.10976e-07  1.000000  0.851237

                 Fisher    maxdiff    meandiff overlapping.genes
              <numeric>  <numeric>   <numeric>       <character>
               0.628877  0.1327492  0.11031085              <NA>
               0.858440 -0.0253801 -0.00844255              <NA>
               0.858440 -0.0705536 -0.03935917             DIP2C
               0.957477  0.0154242  0.00597809              <NA>
               0.990869 -0.0069583 -0.00260668            ANKLE2
```

```r
# set up the grouping variables and colours
pheno$smoker<-as.factor(pheno$smoke2)
cols <- c("magenta","darkgreen")[pheno$smoker]
names(cols) <- levels(pheno$smoker)[pheno$smoker]
```

# Make plots



# Interpretation of Results



# Interpretation of results

It’s not enough to say – these CpG sites were associated with our exposure.

How can our results be applied?

- Predictive biomarkers or disease mechanisms?

What biological process do they indicate?

- Are they enriched in specific pathways?

How do they compare to previous studies?

# Where to find information about your CpGs

Lots of information is in the Illumina manifest:

- Chromosome and locations
- Nearest gene
- GpG context

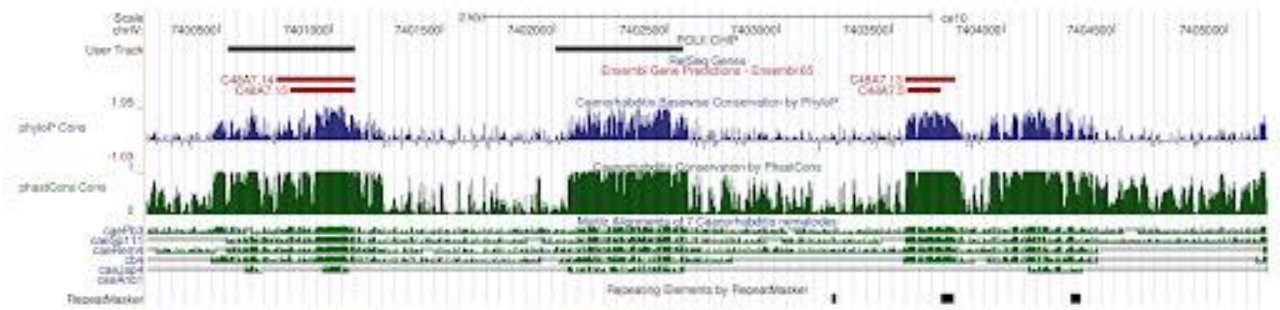
Explore sites on UCSC genome browser

Literature review on top sites

Compare results to previous studies

Gene ontology analyses (gometh in missMethyl package)

Pathway analyses (gometh/DAVID)

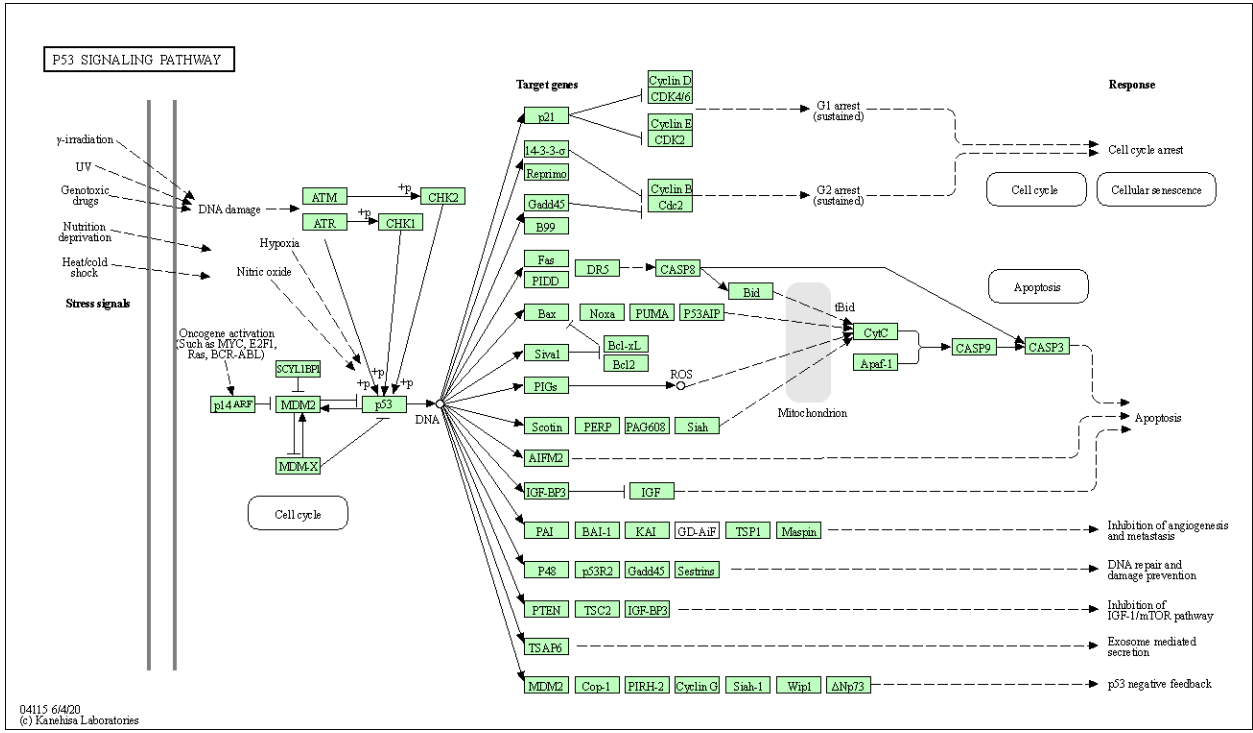


# gometh: Gene ontology testing for 450K methylation data

An ontology comprises a set of well-defined terms with well-defined relationships.
No need for user-dependent, non-systematic, manual annotations when there are numerous affected genes

Two step process

1. Identify target genes of the epigenetic change
2. Use ontological analysis to guess the functional impact of epigenetic changes




https://genomebiology.biomedcentral.com/articles/10.1186/s13059-021-02388-x

# Running gometh

```
check <- getMappedEntrezIDs(sig.cpg = sigCpGs)
length(check$sig.eg)
```

316 genes

```
#' Run gometh using GO
#' Need to include all CpGs tested so that there is a background
gst <- gometh(sig.cpg=sigCpGs, all.cpg=allCpGs, collection="GO",
              plot.bias=TRUE)
topGSA(gst, n=10)
```

Can see that none of these are significant

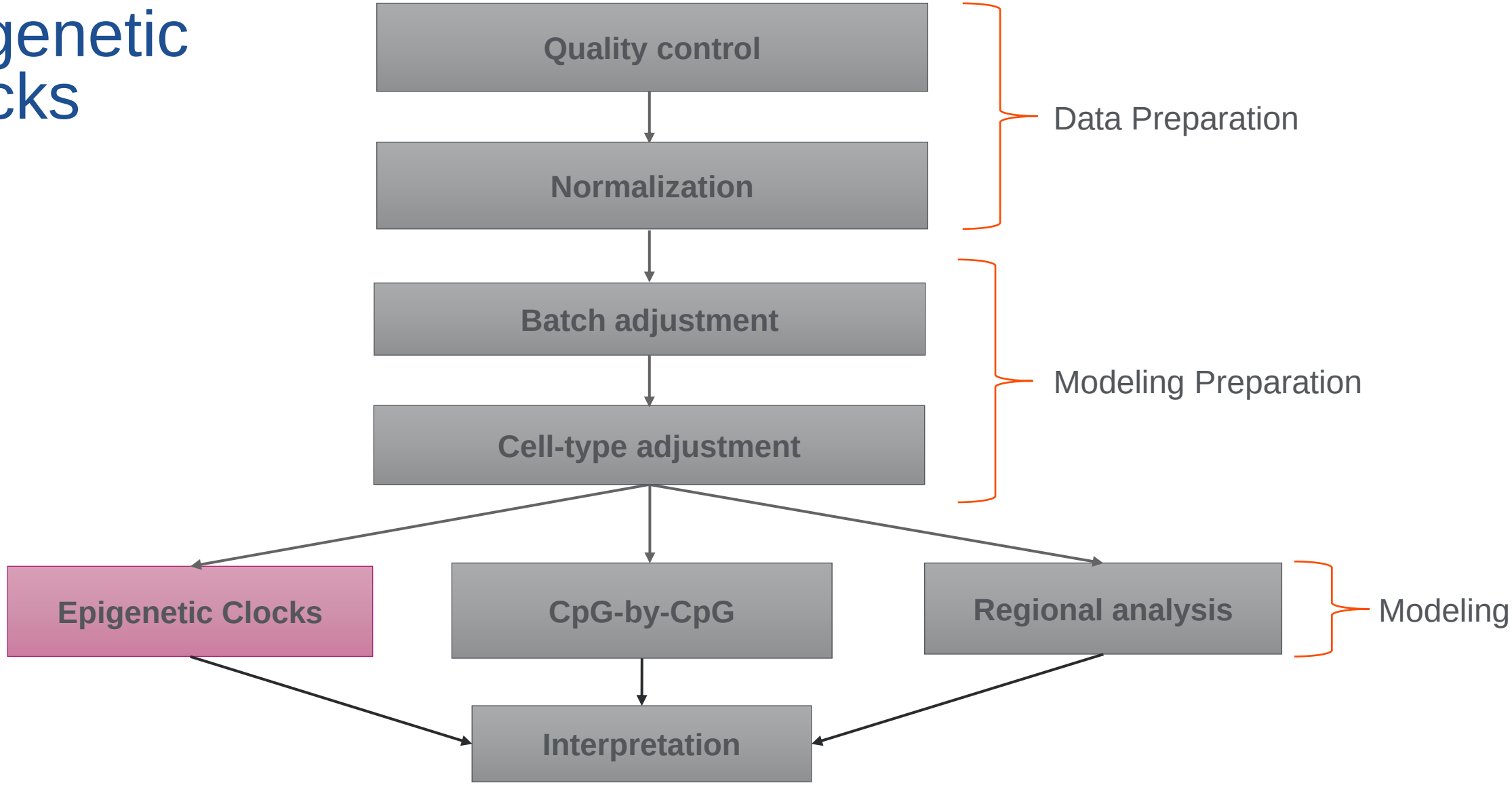
| | ONTOLOGY | TERM | N | DE | P.DE | FDR |
|---|---|---|---|---|---|---|
| GO:0007131 | BP | reciprocal meiotic recombination | 52 | 7 | 1.859867e-05 | 0.2393422 |
| GO:0035825 | BP | homologous recombination | 53 | 7 | 2.117042e-05 | 0.2393422 |
| GO:0003149 | BP | membranous septum morphogenesis | 10 | 4 | 8.113152e-05 | 0.6114883 |
| GO:0060412 | BP | ventricular septum morphogenesis | 45 | 7 | 1.258568e-04 | 0.7039517 |
| GO:0097305 | BP | response to alcohol | 226 | 14 | 1.556658e-04 | 0.7039517 |
| GO:0007127 | BP | meiosis I | 112 | 8 | 4.506708e-04 | 1.0000000 |
| GO:0022612 | BP | gland morphogenesis | 121 | 10 | 5.774465e-04 | 1.0000000 |
| GO:0061982 | BP | meiosis I cell cycle process | 117 | 8 | 6.303332e-04 | 1.0000000 |
| GO:0003281 | BP | ventricular septum development | 76 | 8 | 6.363774e-04 | 1.0000000 |
| GO:0060411 | BP | cardiac septum morphogenesis | 77 | 8 | 8.352745e-04 | 1.0000000 |

# KEGG pathway analysis

```
gst.kegg <- gometh(sig.cpg=sigCpGs, all.cpg=allCpGs, collection="KEGG")
topGSA(gst.kegg, n=10)
```

| | Description | N | DE | P.DE | FDR |
|---|---|---|---|---|---|
| path:hsa00130 | Ubiquinone and other terpenoid-quinone biosynthesis | 11 | 2 | 0.008934156 | 1 |
| path:hsa00360 | Phenylalanine metabolism | 15 | 2 | 0.025027551 | 1 |
| path:hsa00730 | Thiamine metabolism | 15 | 2 | 0.034576087 | 1 |
| path:hsa04714 | Thermogenesis | 212 | 8 | 0.040611279 | 1 |
| path:hsa04919 | Thyroid hormone signaling pathway | 117 | 6 | 0.049434961 | 1 |
| path:hsa03040 | Spliceosome | 124 | 5 | 0.058843071 | 1 |
| path:hsa01240 | Biosynthesis of cofactors | 149 | 5 | 0.076989895 | 1 |
| path:hsa03013 | Nucleocytoplasmic transport | 101 | 4 | 0.082222929 | 1 |
| path:hsa05166 | Human T-cell leukemia virus 1 infection | 214 | 8 | 0.083788637 | 1 |
| path:hsa04110 | Cell cycle | 123 | 5 | 0.087252966 | 1 |

# Epigenetic Clocks



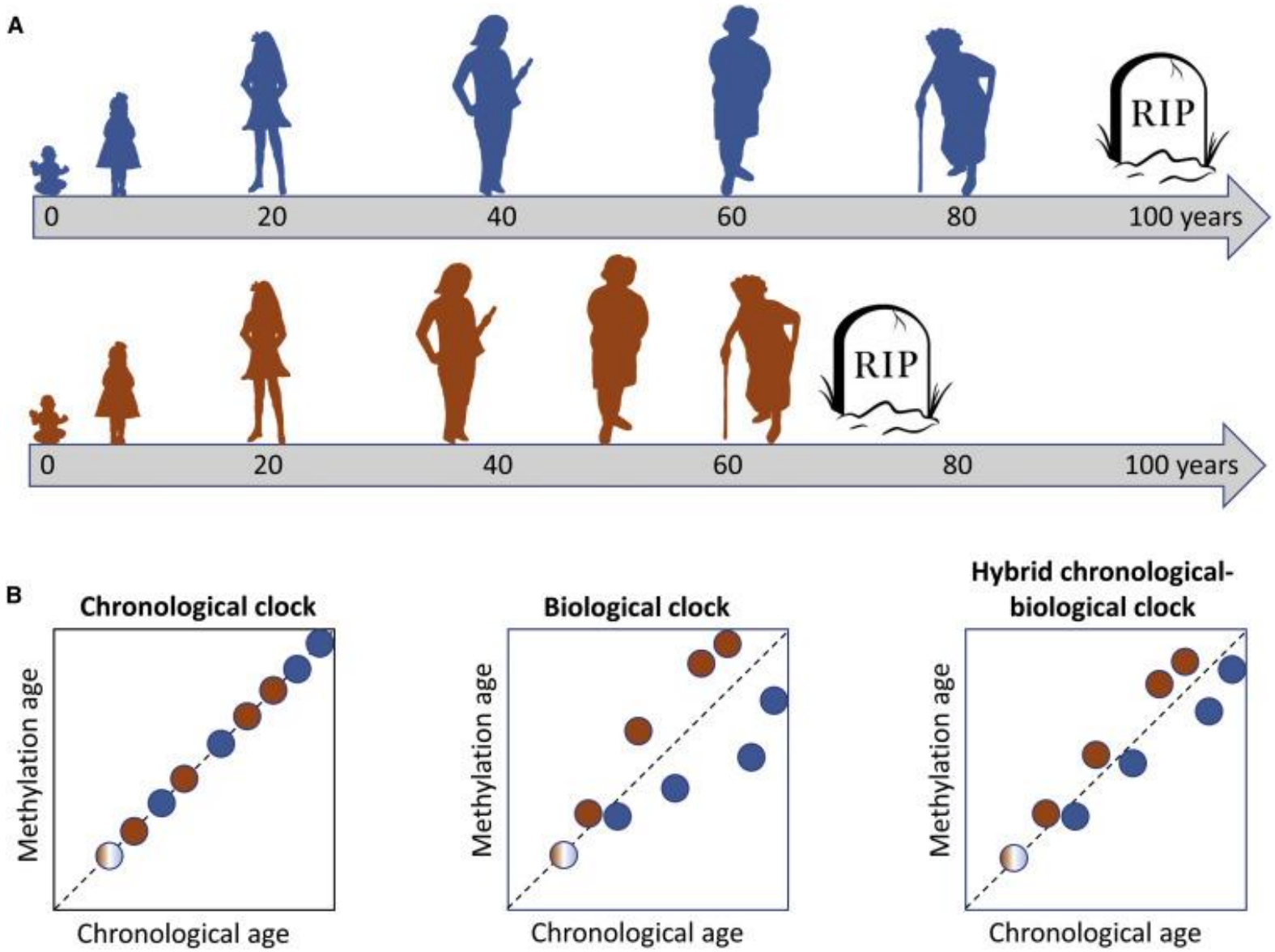
Field et al., 2018

# Epigenetic clocks

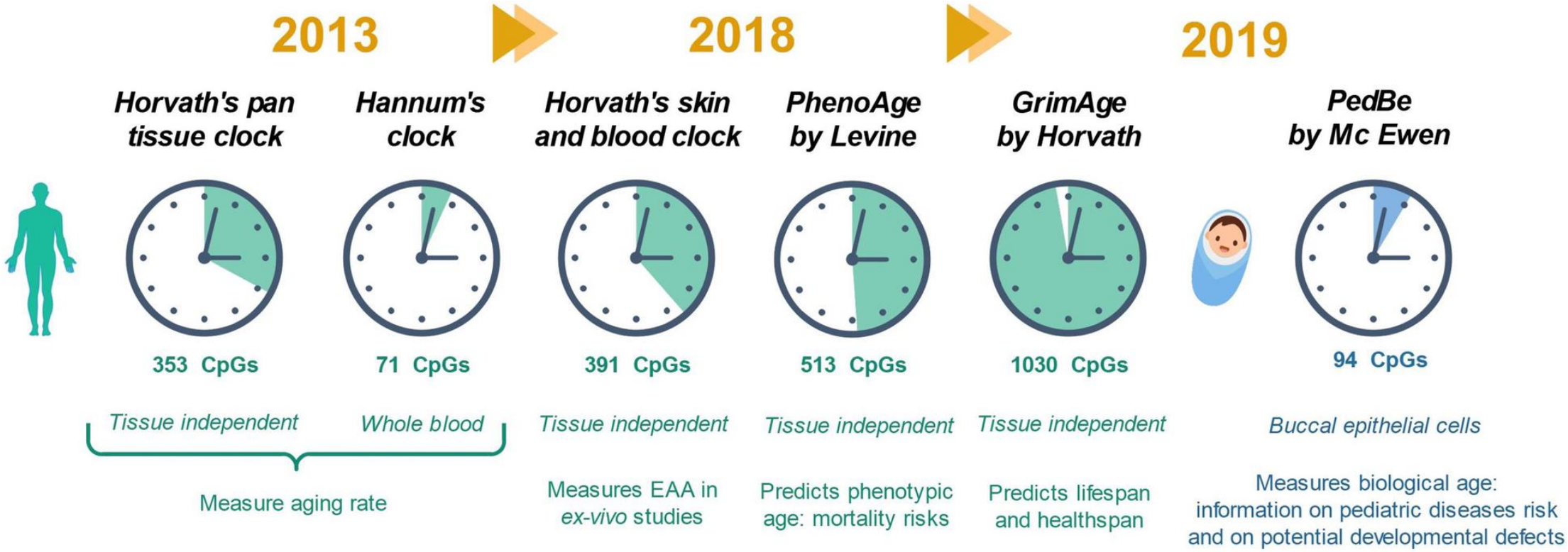
Healthspan and lifespan are not always equivalent.

Different individuals may age at different rates according to their genetics, lifestyle, and environment.

The epigenome has been found to be a sensitive indicator of biological aging processes.



# The many epigenetic clocks




Topart et al., 2020

Most clocks were developed using machine learning to predict chronological age – but more recently epigenetic clocks have focused on phenotypic aging and mortality.

# Horvath's DNA methylation age

Developed using 8,000 samples and 51 tissue types.

Consists of 353 CpG sites.

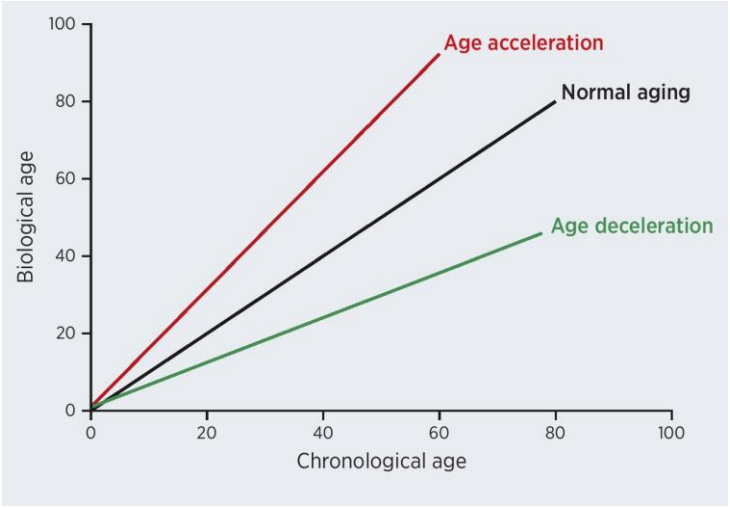
CpGs show enrichment for cell death/survival, cellular growth/ proliferation, organismal/tissue development, and cancer

Has since been adapted for 850K data but is only available via the web portal: http://dnamage.genetics.ucla.edu/

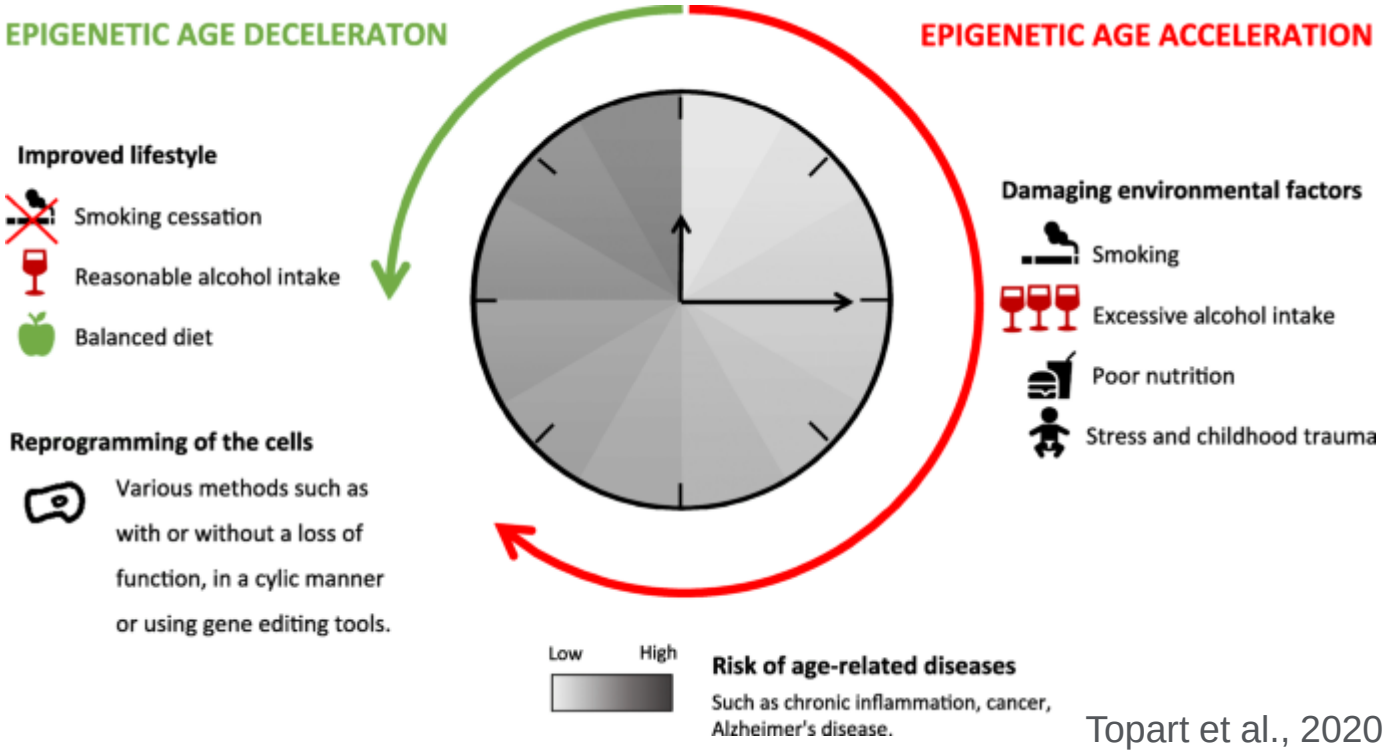
# Epigenetic Age Acceleration

We often use the difference between chronological age and DNA methylation age as a more sensitive indicator of biological aging.

Can also calculate using the residuals of a regression of DNA methylation age by chronological age.




Yu et la., 2020




Topart et al., 2020

# Questions to ask ourselves in DNAm age analyses

- What is the goal of my study?

  Each clock was developed based on a specific set of predictors – chronological age, aging phenotypes, or mortality.

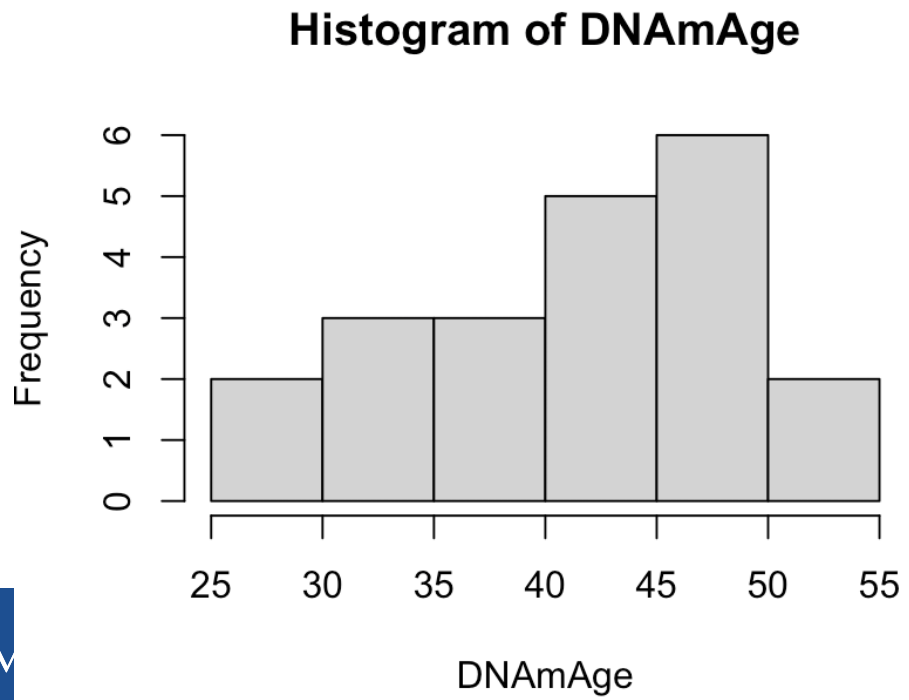
- Why is this an important research question?

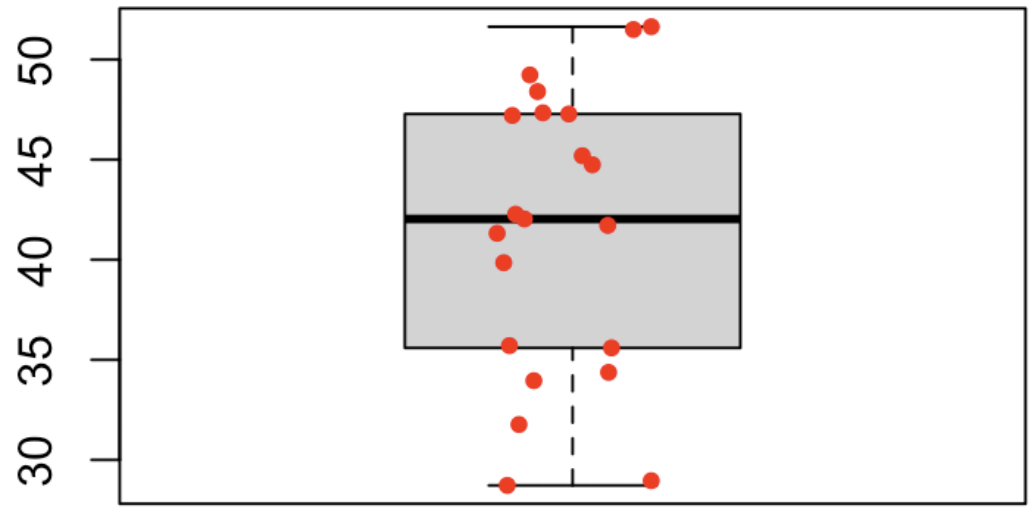
- How are we going to apply these results?

- It's important to keep in mind that these clocks were developed as predictors – they do not necessarily indicate a causal process.

# Estimating DNAm age with the WateRmelon package

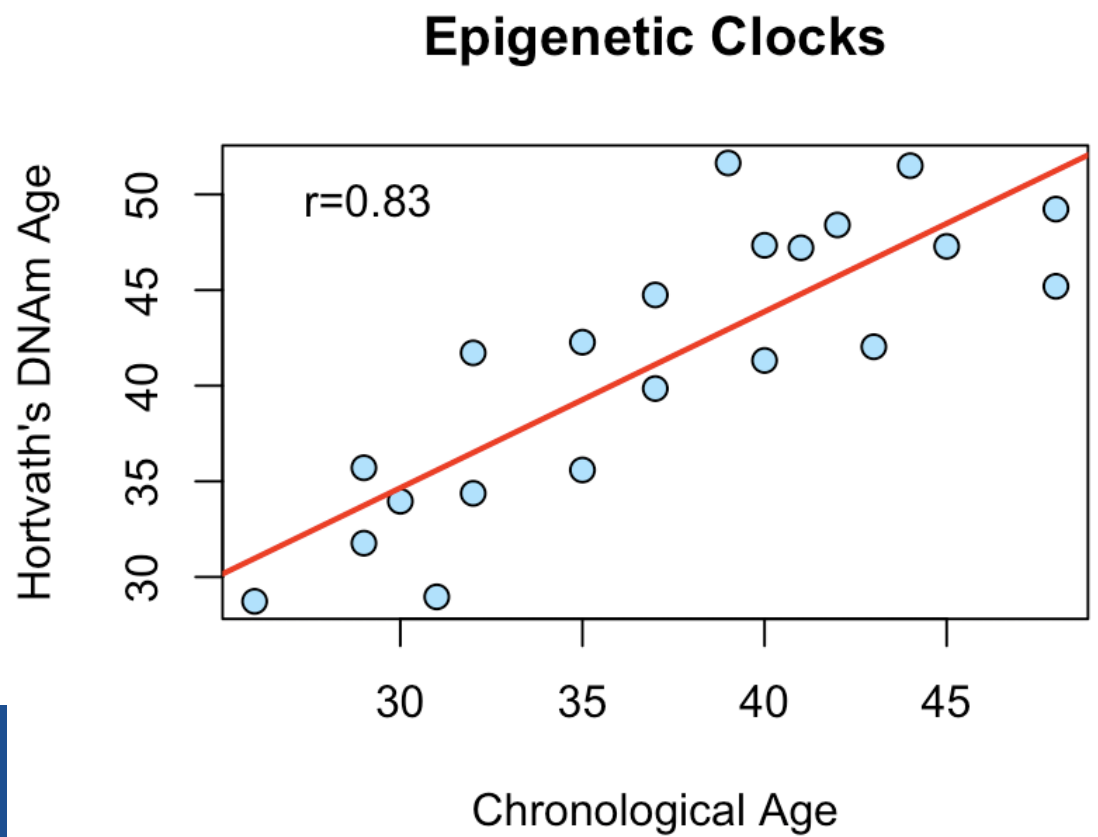
```
suppressMessages(library(wateRmelon))

DNAmAge <- as.vector(agep(betas.clean))
hist(DNAmAge)
boxplot(DNAmAge);
stripchart(DNAmAge, vertical = T,method = "jitter", add = T, pch = 20, col = 'red')
```



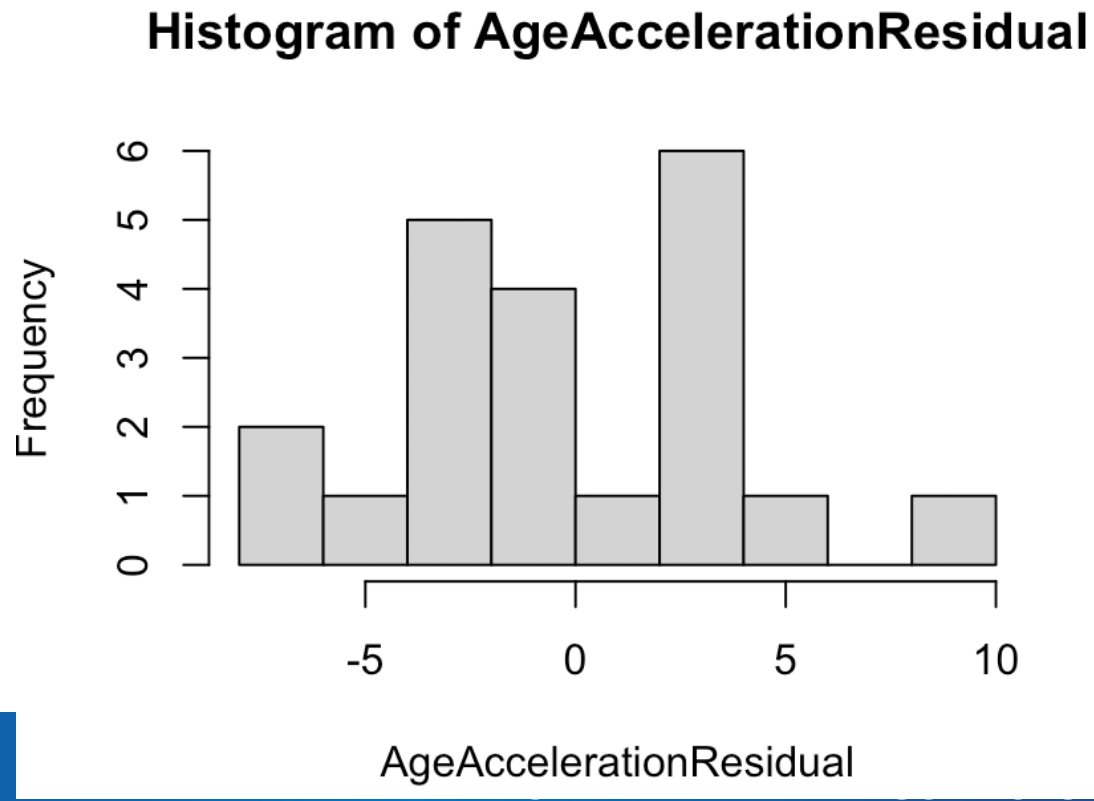
# Relate to chronological age

```
#' Correlation; agreement
plot(pheno$age_sampling,DNAmAge,pch=21,ylab="Hortvath's DNAm Age",
     xlab="Chronological Age",cex=1.2, bg=alpha("deepskyblue",0.45),main="Epigenetic Clocks")
legend("topleft",legend=paste0("r=",round(cor(pheno$age_sampling,DNAmAge),2)),bty="n")
abline(lm(DNAmAge~pheno$age_sampling),col="red",lw=2)
```



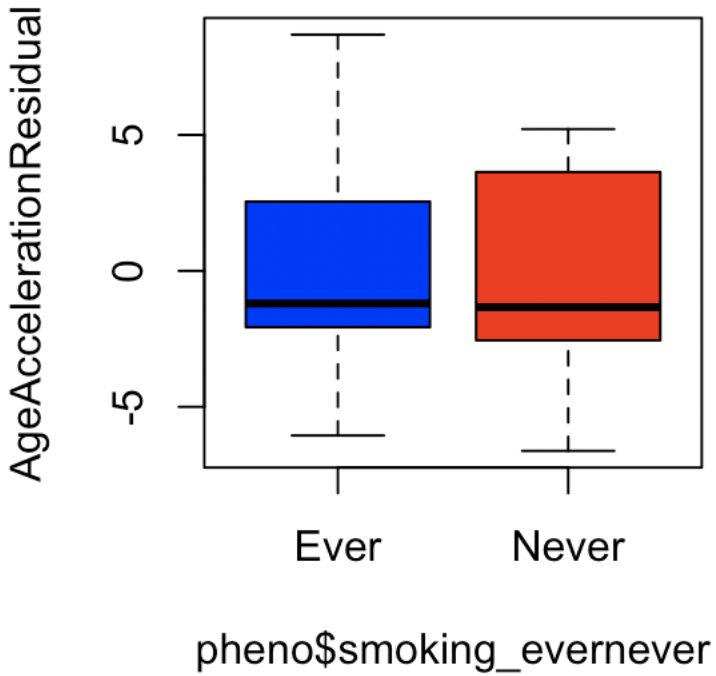
# Calculate age acceleration

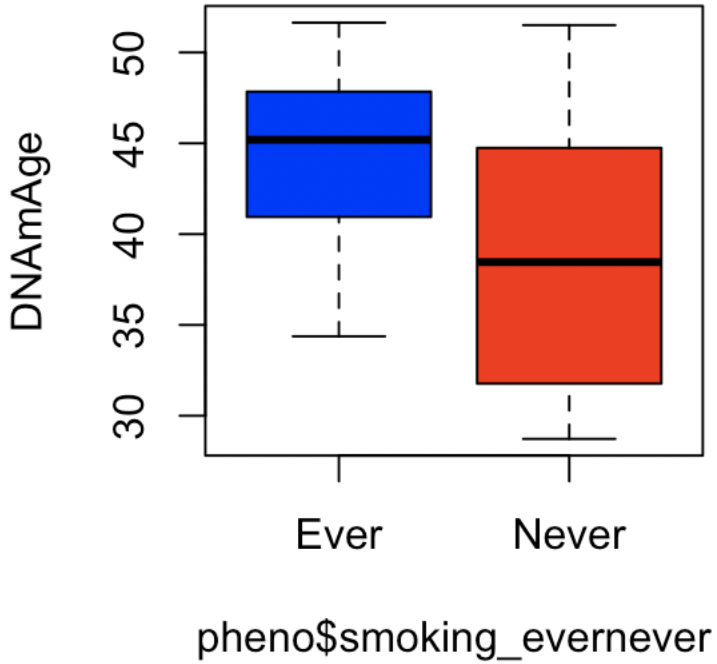
```
#' Age Acceleration Residuals
AgeAccelerationResidual <- residuals(lm(DNAmAge~pheno$age_sampling))
hist(AgeAccelerationResidual)
```



```
boxplot(AgeAccelerationResidual ~pheno$smoking_evernever, col=c("blue","red"))
wilcox.test(AgeAccelerationResidual ~ pheno$smoking_evernever)

boxplot(DNAmAge ~pheno$smoking_evernever, col=c("blue","red"))
wilcox.test(DNAmAge ~ pheno$smoking_evernever)
```

# Relation to smoking status




```
data:  AgeAccelerationResidual by pheno$smoking_evernever
W = 54, p-value = 0.9725
```

```
data:  DNAmAge by pheno$smoking_evernever
W = 80, p-value = 0.08452
```

# Extra: Validation and Replication

# Discovery vs. Replication

## Discovery only (single sample analysis)

- Prone to false positive findings (negative too)

## Internal Replication

- Sample two or more groups from the same population
- K-fold, leave one out, etc.
- Overall power lower than same-size discovery only

## External (Independent) Replication

- Two (or more) independent studies
- Ensure validation + generalizability

## Meta-analysis

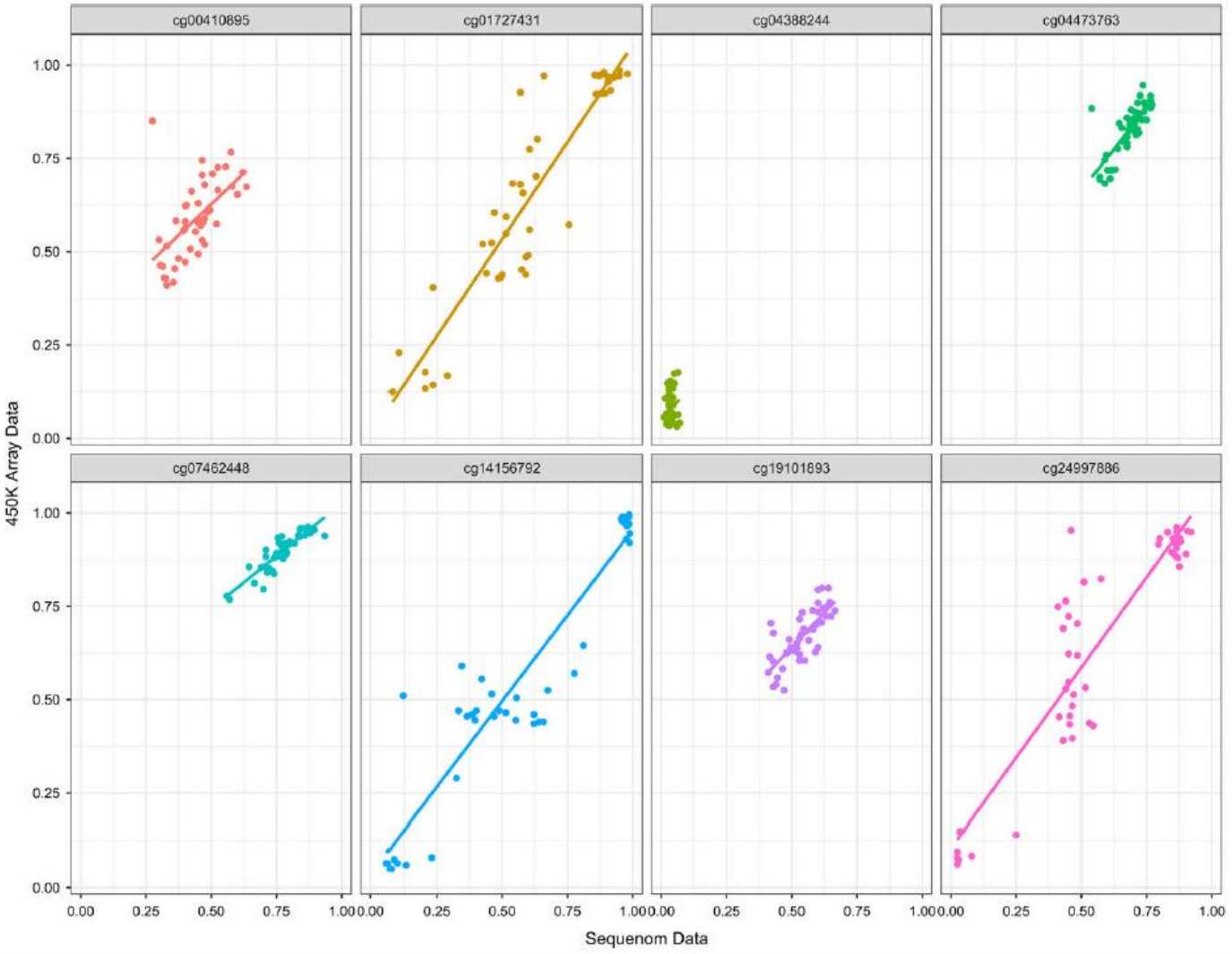
# Cross Platform Validation

Absolute estimates of methylation will differ based on approach

- Hope that rank order stays the same

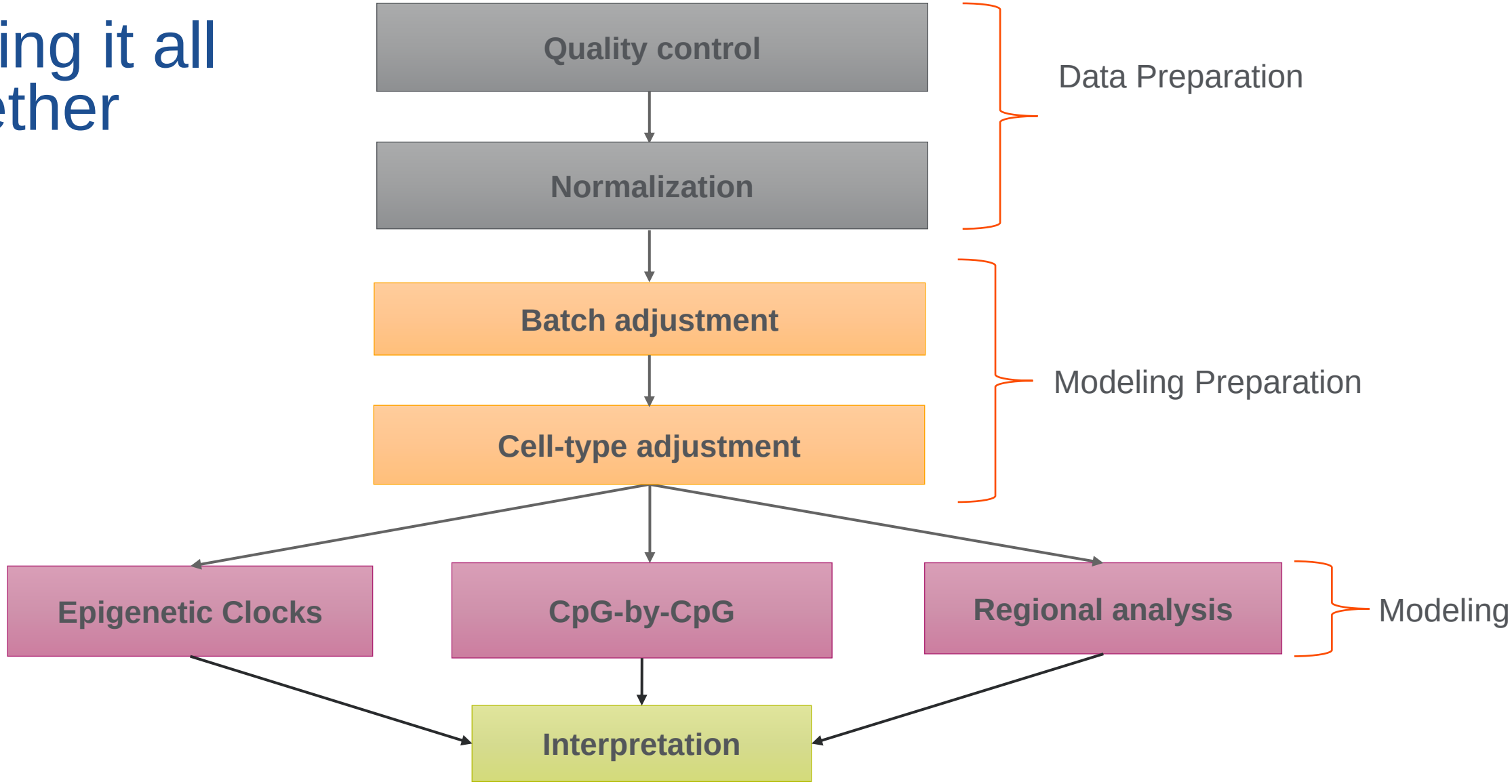
Sometimes, different platforms will not correlate

- Less frequent in significant CpGs (although it still happens)




Wu et al. 2017

# Putting it all together



# Conclusions

1. There are many powerful tools available for EWAS.
2. However, these are not a substitute for good study design, clear hypotheses and a good understanding of statistics.
3. Be aware of potential pitfalls for regression
4. Be careful in interpretation of findings
5. Always have external replication when possible

# Questions??

Email: ak4181@cumc.Columbia.edu