Statistical Methods for High-dimensional Biology

Exploratory Data Analysis, Data visualization, Experimental Design

Yongjin Park, UBC Path&Lab, STAT, BC Cancer

Today's lecture: EDA & Exp Design

- **Exploratory Data Analysis**
  - First steps to data analysis
  - Traditional (low-dimensional) approaches
  - High-dimensional methods
  - Tips on how to organize your data/project
- **Experimental Design**
  - Observational vs. Experimental studies
  - Identification of unwanted variation (SVA)
  - Causal inference: matching, stratification, inverse propensity

What is EDA?

Graphing data *is a powerful approach to detecting* [data-specific] *problems. We refer to this as exploratory data analysis (EDA). Many important methodological contributions to existing techniques in data analysis were initiated by discoveries made via EDA.*

Rafael Irizzary

A few questions to ask after you obtained data

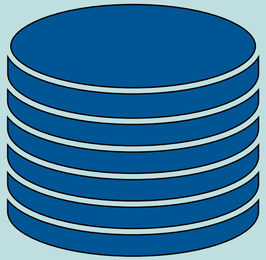
What are the data types?

{0, 1, 2, ...} ?

{-4.2, -3, -2, 0, 1, 2.1, ...}

{0.1, 0.8, 0.2, ...}

How is it stored?

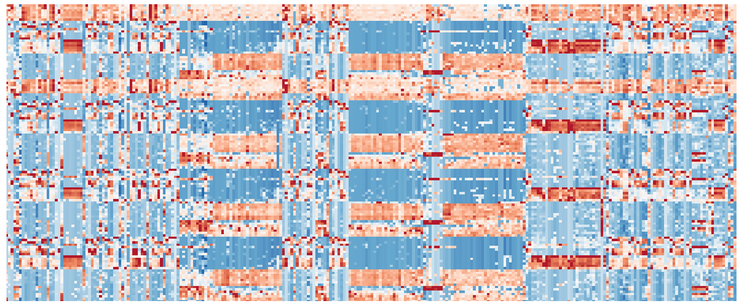


full batch

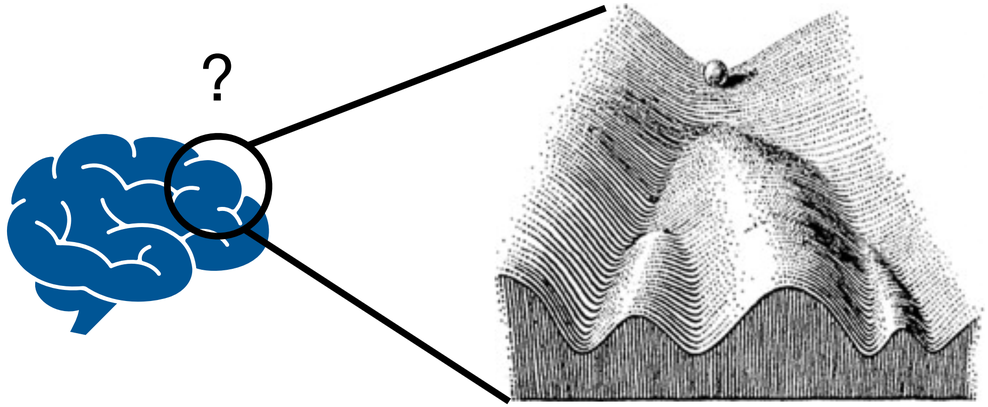
vs.

data steam (online)

How many features? samples?



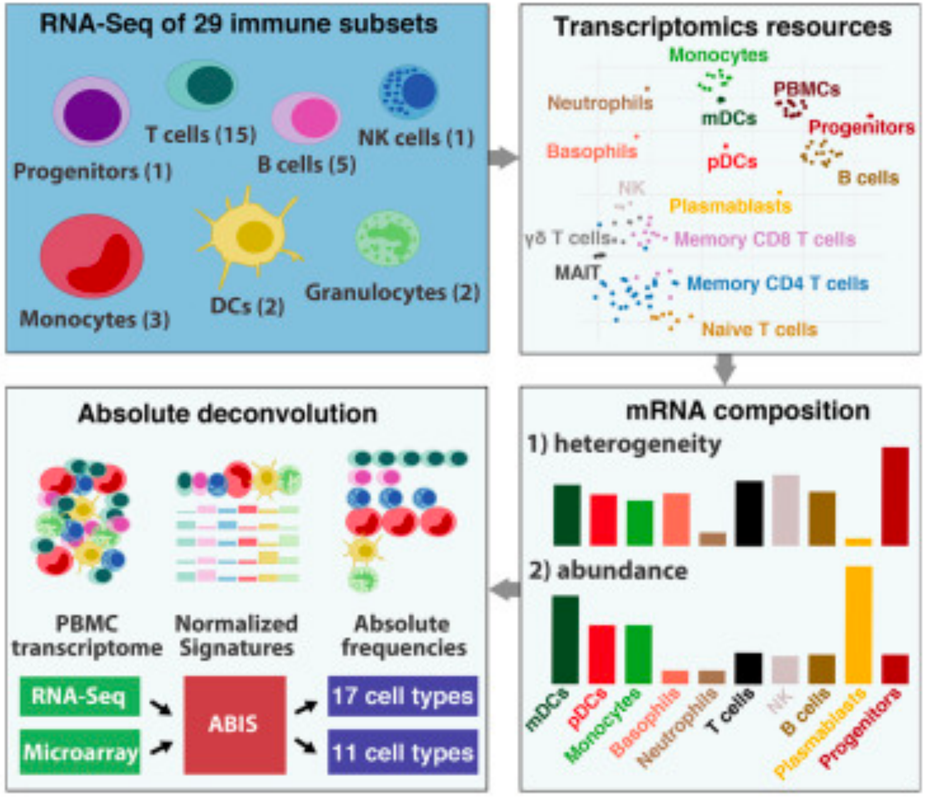
How can I picture it?



Why EDA *before* Data Analysis?

- Reading all the numbers vs. picturing all the numbers
- Modelling assumption vs. data distributions/shapes
- Start your hypothesis emprical, observed properties
- We may not know what to do at first
- Triage to figure out most suitable data analysis methods

A data set to demonstrate the principels of EDA




```r
if (!require("BiocManager", quietly = TRUE)){
    install.packages("BiocManager")
}
if(!require("GEOquery", quietly = TRUE)){
    BiocManager::install("GEOquery")
}
raw.data.file <- "GSE107011/GSE107011_Processed_data_TPM.txt.g
if(!file.exists(raw.data.file)){
    getGEOSuppFiles("GSE107011")
}
raw.data <- fread(raw.data.file)
X <- as.matrix(raw.data[, -1])
rownames(X) <- unlist(raw.data[,1])
```

Monaco + Lee *et al.* Cell Reports (2019)

Code available in our GitHub repo

https://github.com/STAT540-UBC/lectures/blob/main/lect03-eda/eda.Rmd

What do they look like?

```
X[1:5, 1:5]
```

```
##                    DZQV_CD8_naive DZQV_CD8_CM DZQV_CD8_EM DZQV_CD8_TE DZQV_MAIT
## ENSG00000223972.5       0.214321     0.00000  0.00421414  0.00953342 0.0112167
## ENSG00000227232.5       4.759630     3.49280  2.64877000  3.72232000 2.4792700
## ENSG00000278267.1       0.000000     2.61518  2.45976000  0.00000000 4.5861000
## ENSG00000243485.5       0.000000     0.00000  0.00000000  0.00000000 0.0000000
## ENSG00000284332.1       0.000000     0.00000  0.00000000  0.00000000 0.0000000
```

- What do you see?

What are the range of values?

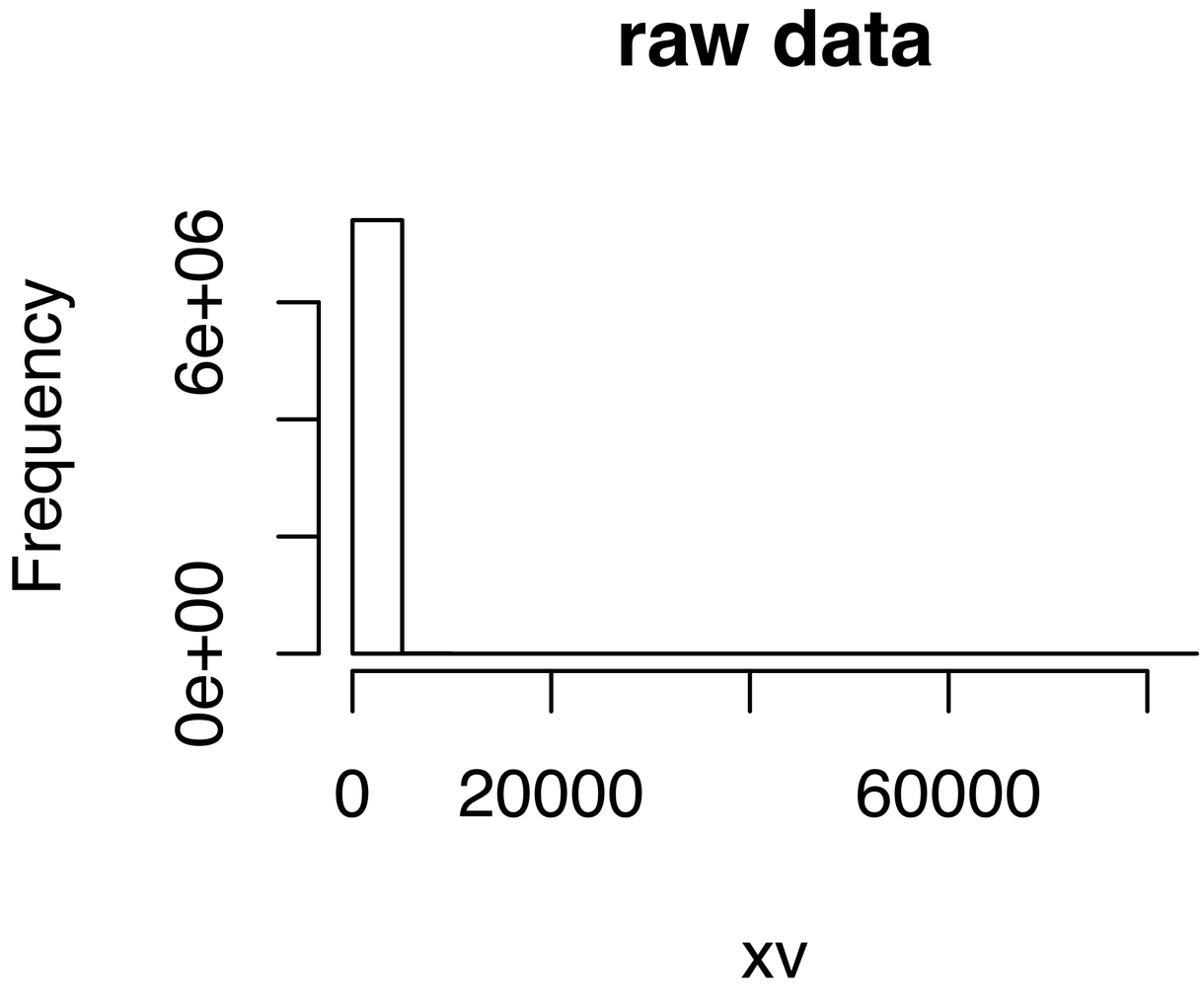
```
xv <- as.vector(X)
range(xv) %>%
    round(digits=2)
```

```
## [1]     0.0 80639.8
```

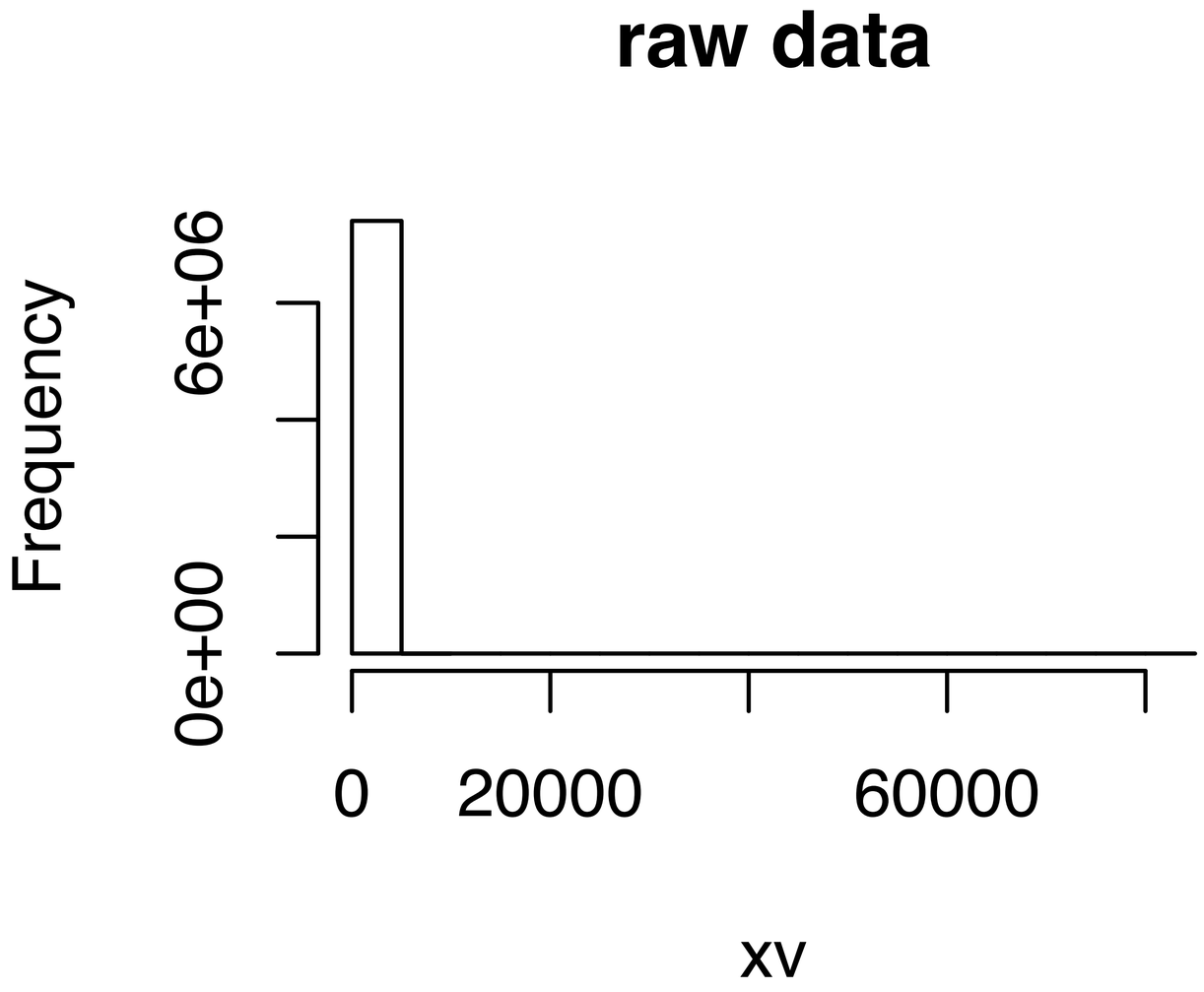
```
breaks <- c(0.1, 0.25, 0.5, 0.75, 0.9)
quantile(xv, probs=breaks) %>%
    round(digits=2)
```

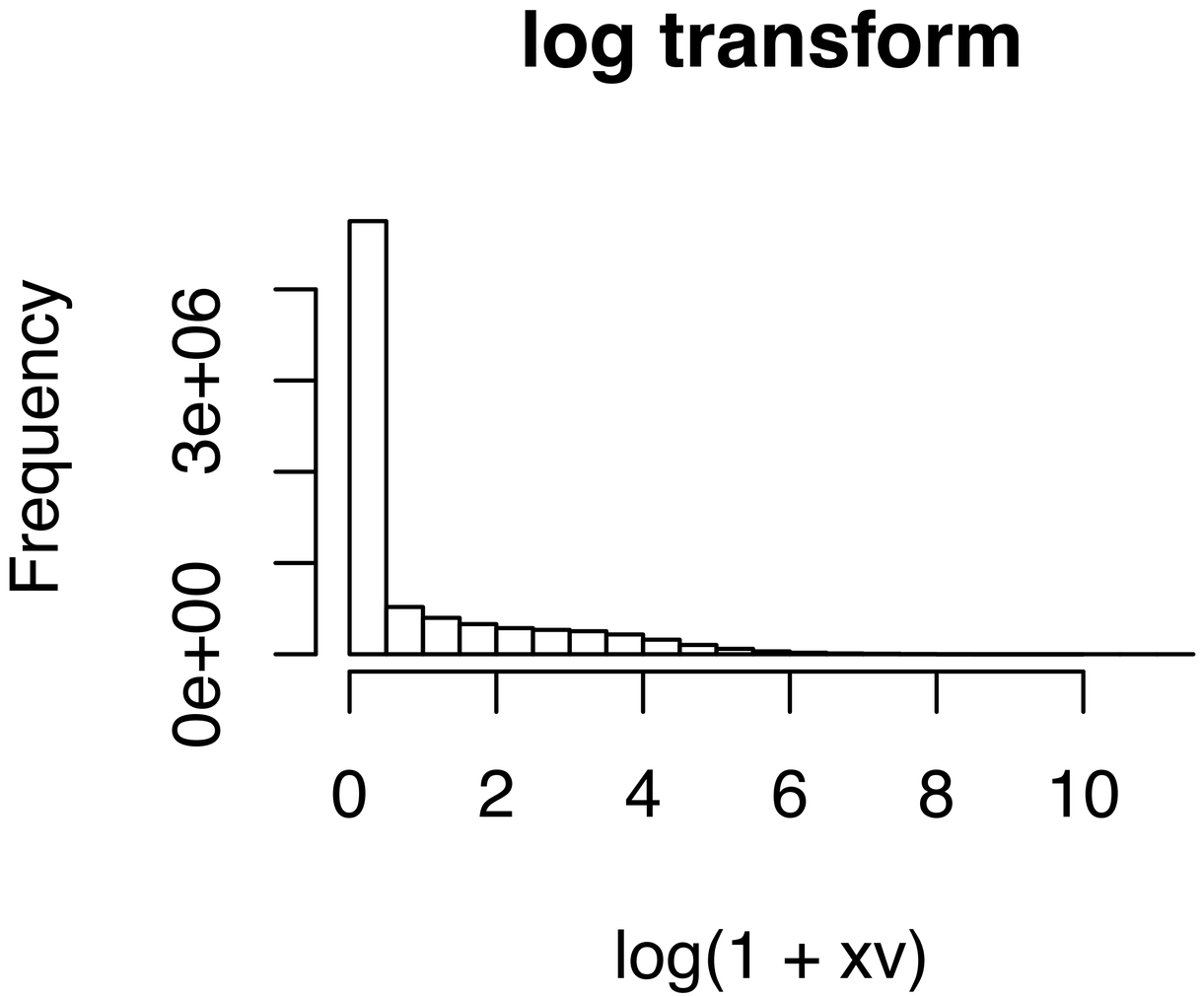
```
##    10%    25%    50%    75%    90%
##   0.00   0.00   0.00   2.86  24.16
```

Histogram `hist`: Roughly... how are they distributed?

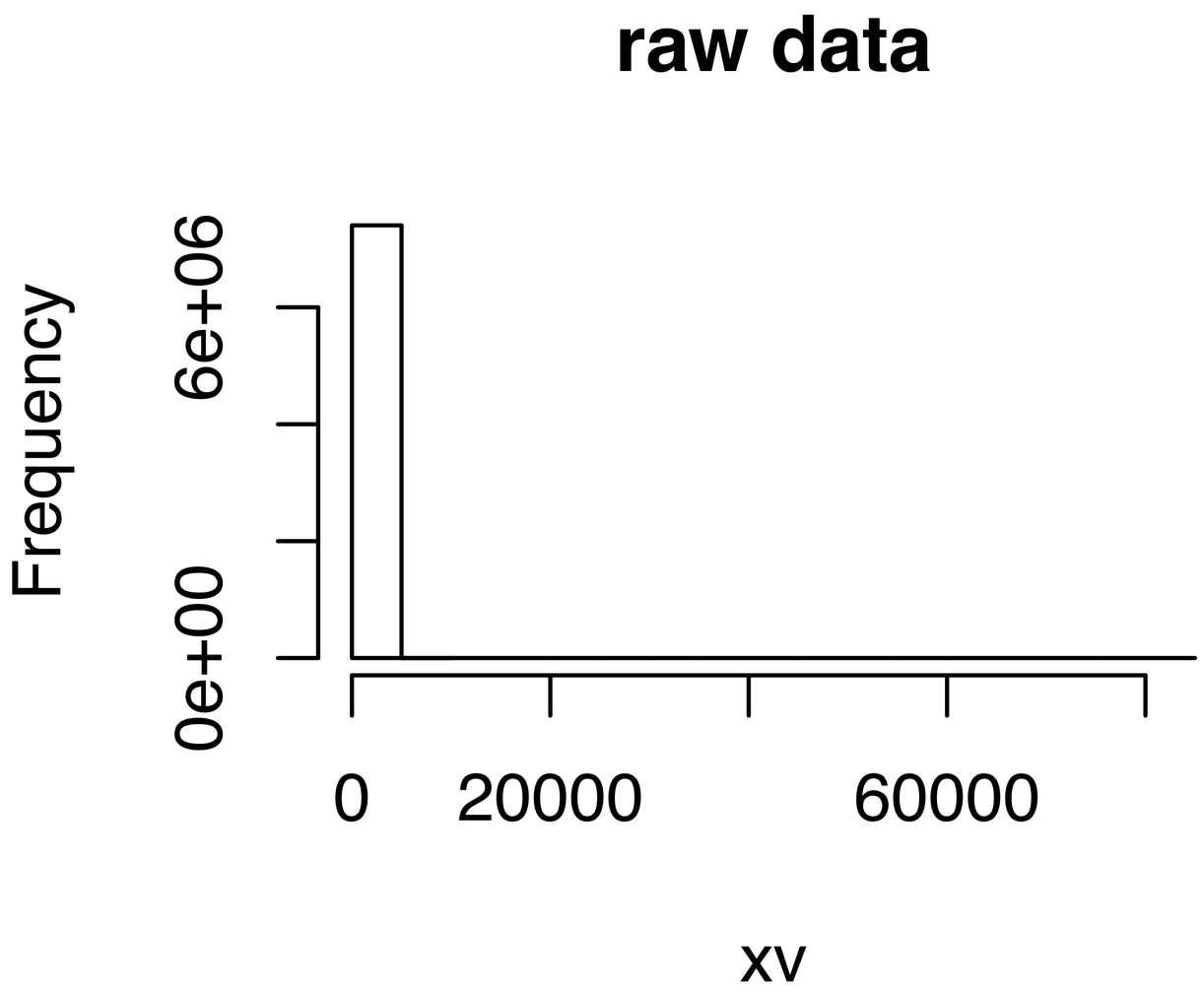


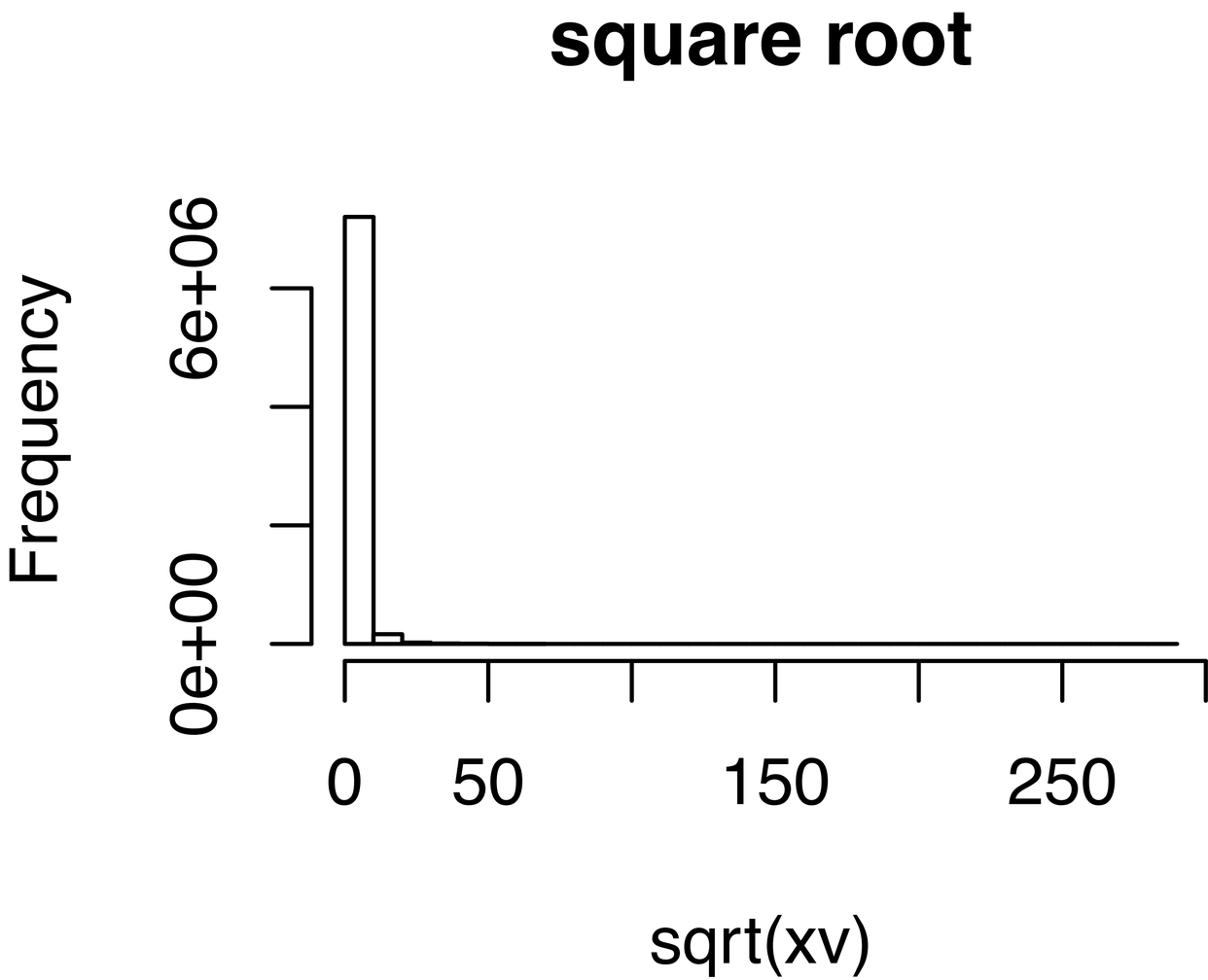
Histogram `hist`: Roughly... how are they distributed?





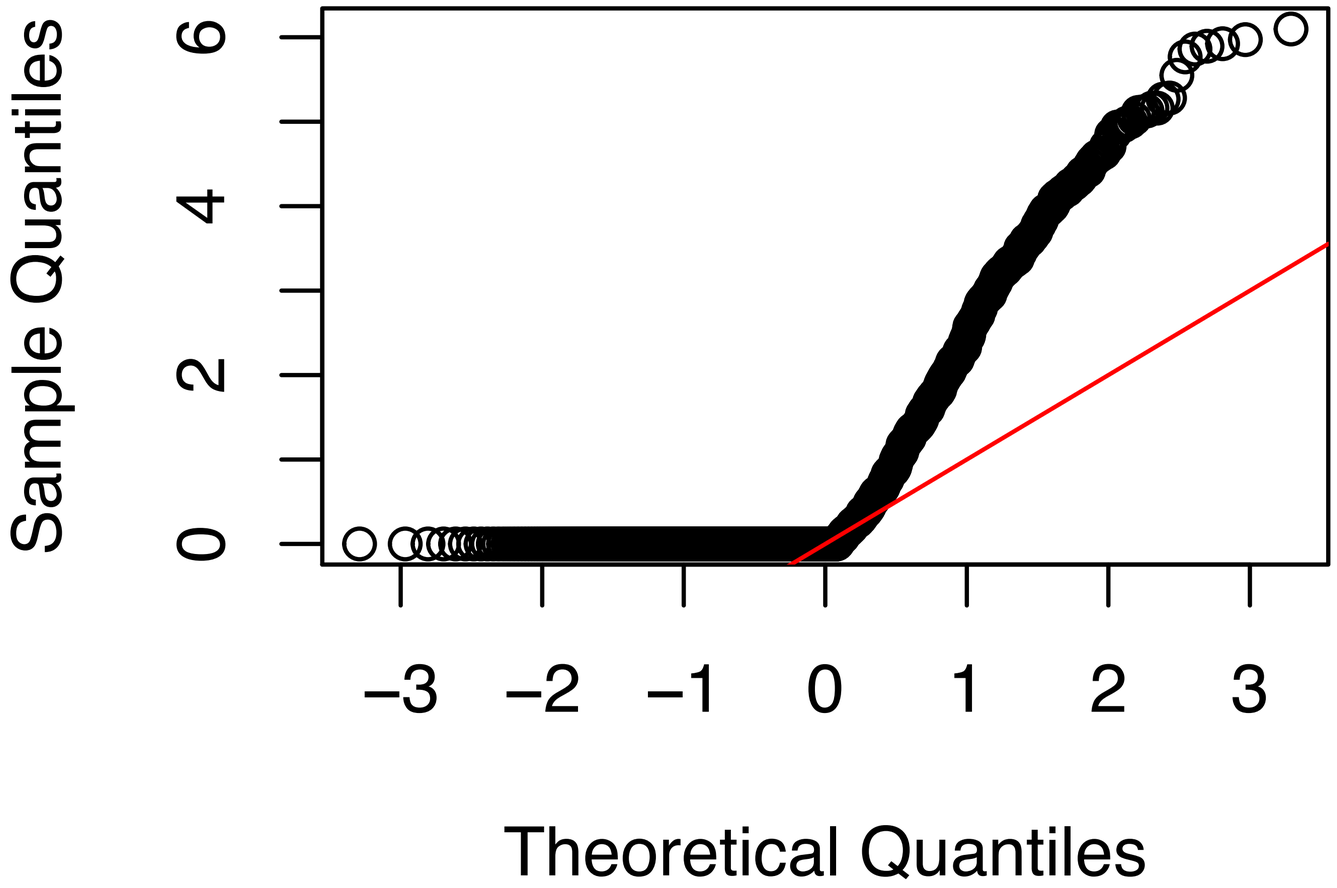
Histogram `hist`: Roughly... how are they distributed?





Quantile-quantile plot against standard Normal

```r
xv.rand <- sample(xv, 1000)       # random 1000 ponits
qqnorm(log(1+xv.rand), main="") # compare with the Normal
abline(a=0,b=1,col=2)             # diagnoal line
```



Side note: Why do many statisticians care about Normality?

- Full understanding of its properties
- Maximum entropy (most random/chaotic) distribution of real numbers
- Defined by first and second moments (mean and $\approx$ var.)
- Central Limit Theorem (will discuss in the next lecture)
- Many statistical methods were built on the Normality assumption.

Let's investigate the data more with ggplot

ggplot and dplyr love fully linearized format.

```
x.melt <- reshape2::melt(X); head(x.melt, 3)
```

```
##                 Var1            Var2    value
## 1 ENSG00000223972.5 DZQV_CD8_naive 0.214321
## 2 ENSG00000227232.5 DZQV_CD8_naive 4.759630
## 3 ENSG00000278267.1 DZQV_CD8_naive 0.000000
```

What are Var1 and Var2?

```
head(rownames(X), 3)
```

```
##                 V11                 V12                 V13
## "ENSG00000223972.5" "ENSG00000227232.5" "ENSG00000278267.1"
```

```
head(colnames(X), 3)
```

```
## [1] "DZQV_CD8_naive" "DZQV_CD8_CM"    "DZQV_CD8_EM"
```

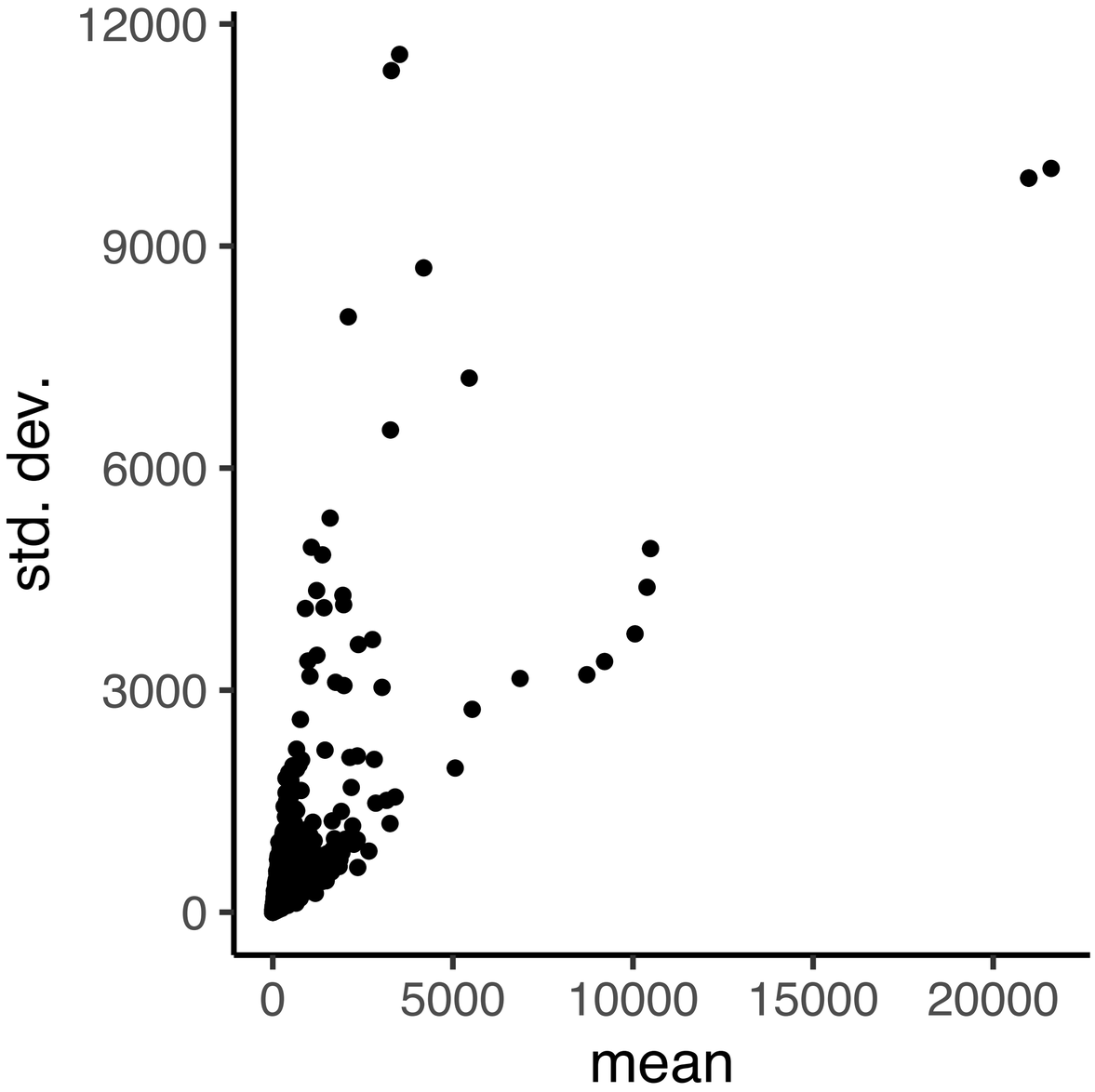
Excess of zeros... Can we zoom-in non-zero values?

```
row.stat <- x.melt %>%
    group_by(`Var1`) %>%
    summarize(m=mean(`value`),
              s=sd(`value`),
              nz=sum(`value` > 0),
              cv=`s`/`m`) %>%
    ungroup()
```

- `m`: the mean of each row (`Var1`)
- `s`: standard deviation
- `nz`: number of non-zeros
- `cv`: coefficient of variation (`s/m`)

```
zero.rows <- row.stat %>% filter(m == 0) %>%
    nrow()
tot.rows <- nrow(row.stat)
```

- 10,326 rows/features (out of 58,311) are just empty



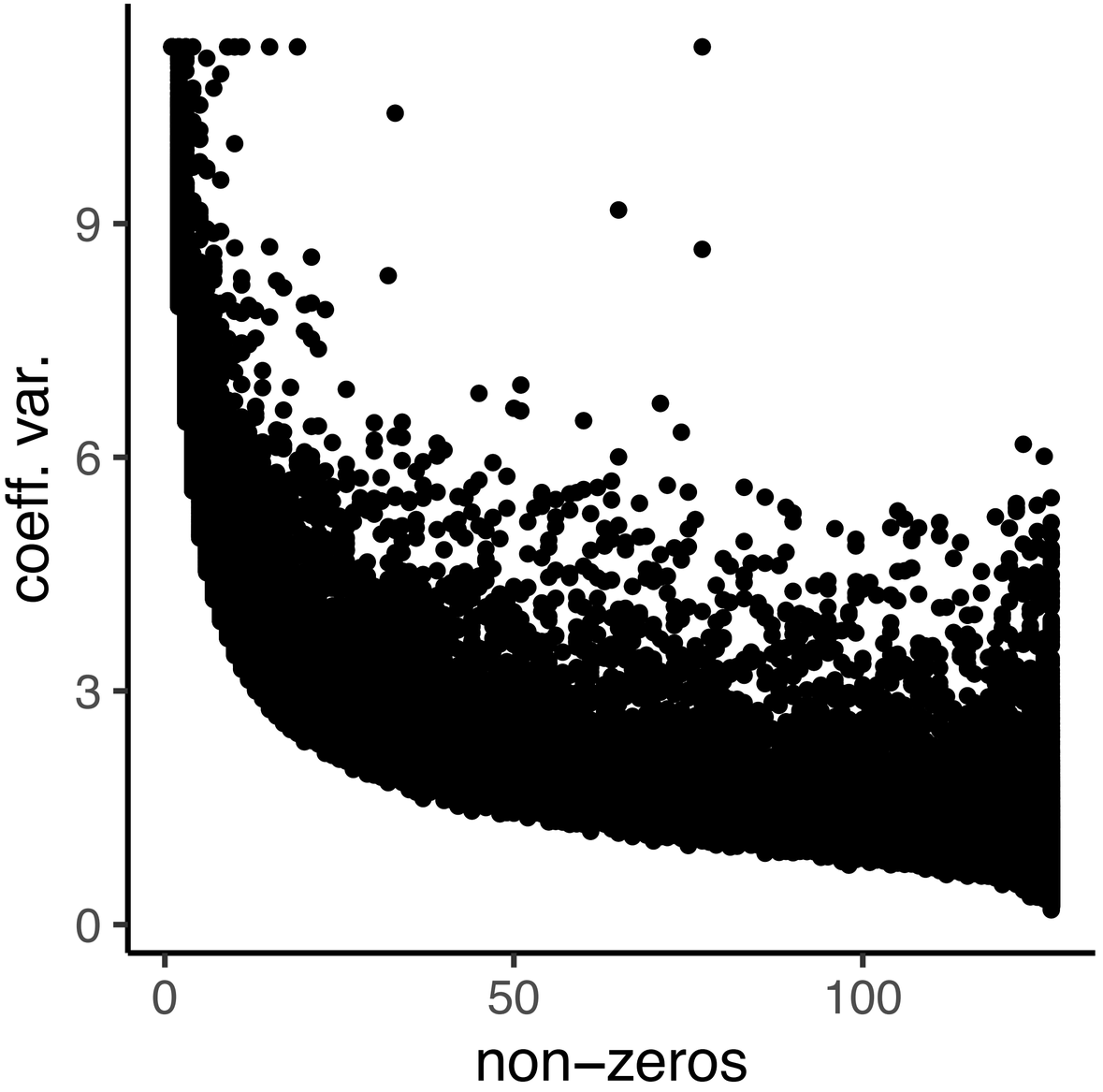
Excess of zeros... Can we zoom-in non-zero values?

```
row.stat <- x.melt %>%
    group_by(`Var1`) %>%
    summarize(m=mean(`value`),
              s=sd(`value`),
              nz=sum(`value` > 0),
              cv=`s`/`m`) %>%
    ungroup()
```

- `m`: the mean of each row (`Var1`)
- `s`: standard deviation
- `nz`: number of non-zeros
- `cv`: coefficient of variation (`s/m`)

```
zero.rows <- row.stat %>% filter(m == 0) %>%
    nrow()
tot.rows <- nrow(row.stat)
```

- 10,326 rows/features (out of 58,311) are just empty



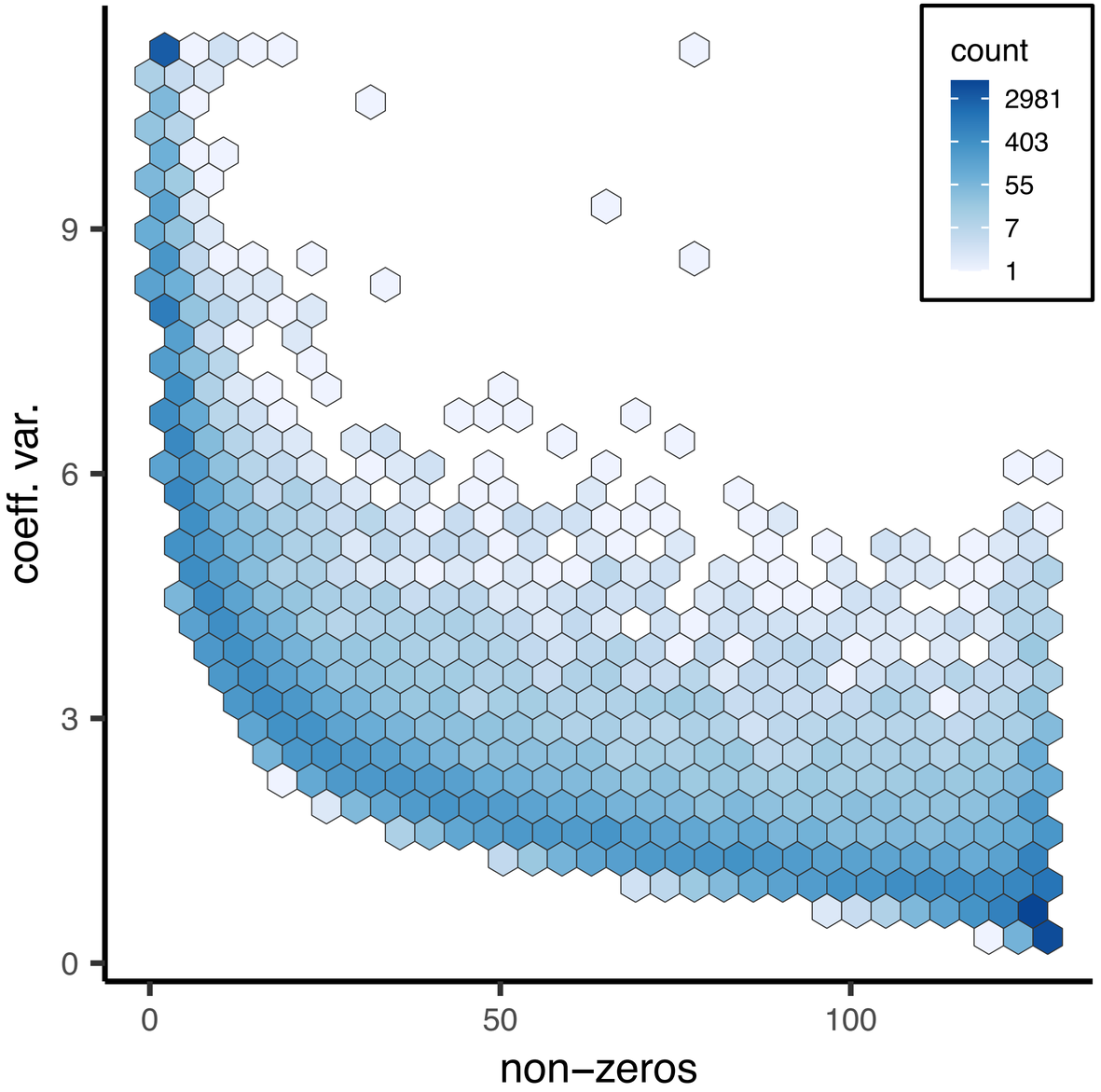
Excess of zeros... Can we zoom-in non-zero values?

```
row.stat <- x.melt %>%
    group_by(`Var1`) %>%
    summarize(m=mean(`value`),
              s=sd(`value`),
              nz=sum(`value` > 0),
              cv=`s`/`m`) %>%
    ungroup()
```

- `m`: the mean of each row (`Var1`)
- `s`: standard deviation
- `nz`: number of non-zeros
- `cv`: coefficient of variation (`s/m`)

```
zero.rows <- row.stat %>% filter(m == 0) %>%
    nrow()
tot.rows <- nrow(row.stat)
```

- 10,326 rows/features (out of 58,311) are just empty



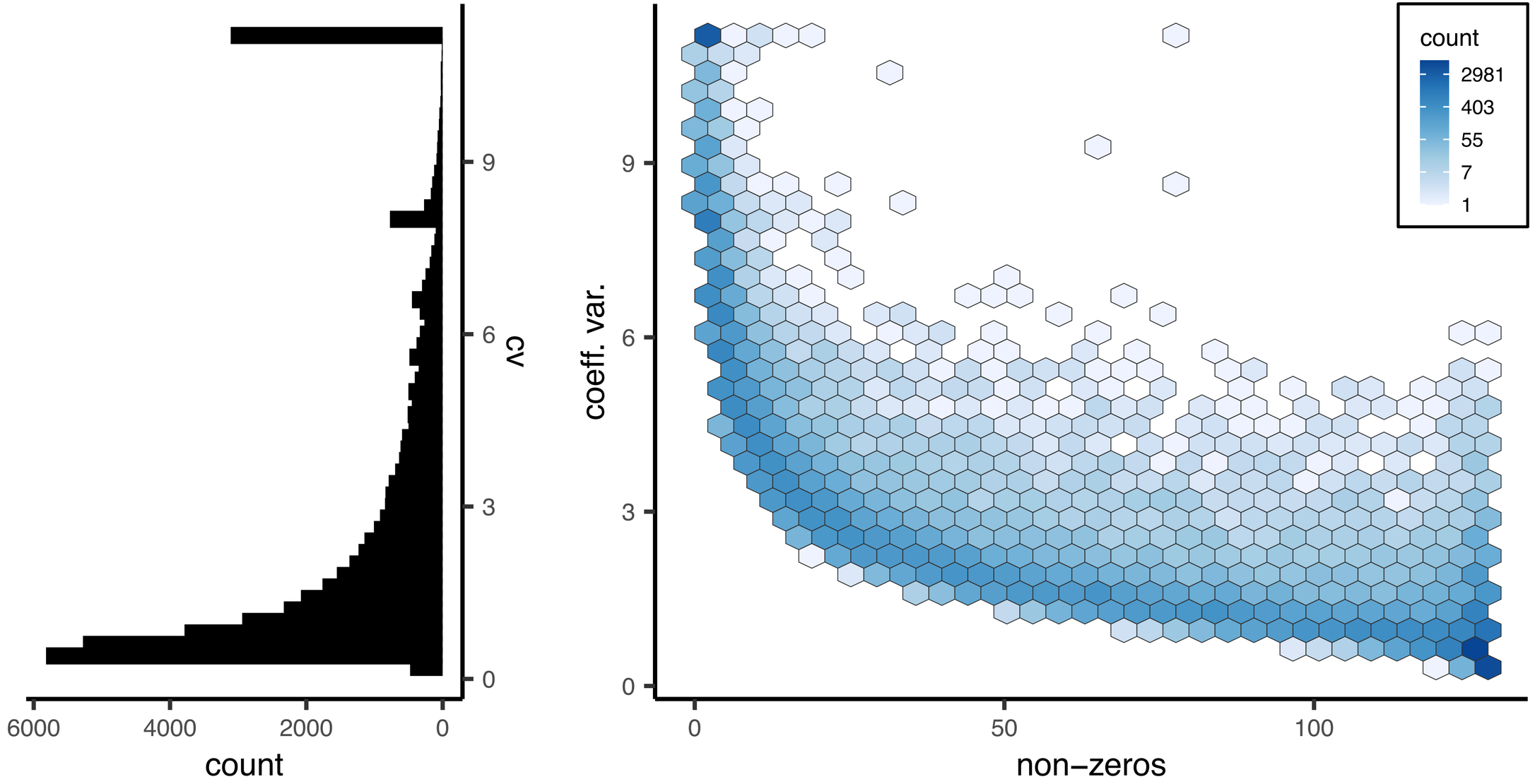
Filter out features (based on some statistics)

```
.df <- row.stat %>%
    filter(nz > 0)

p1 <-
    .gg.plot(.df, aes(nz, cv)) +
    theme(legend.position = c(1,1), legend.justifi
    geom_hex(color="gray20", size=.1) +
    scale_fill_distiller(trans="log", labels=round
    xlab("non-zeros") +
    ylab("coeff. var.")
```

```
.df.cv.hist <- .df %>%
    group_by(cv = round(cv*5)/5) %>%
    summarize(count = n()) %>%
    ungroup()

p0 <-
    .gg.plot(.df.cv.hist, aes(x=0, xend=`count`, y=
    geom_segment(size=2) +
    scale_x_reverse("count") +
    scale_y_continuous(position="right")
```

Filter out features (based on some statistics)

```
wrap_plots(p0, p1, nrow=1, widths=c(2,4))
```

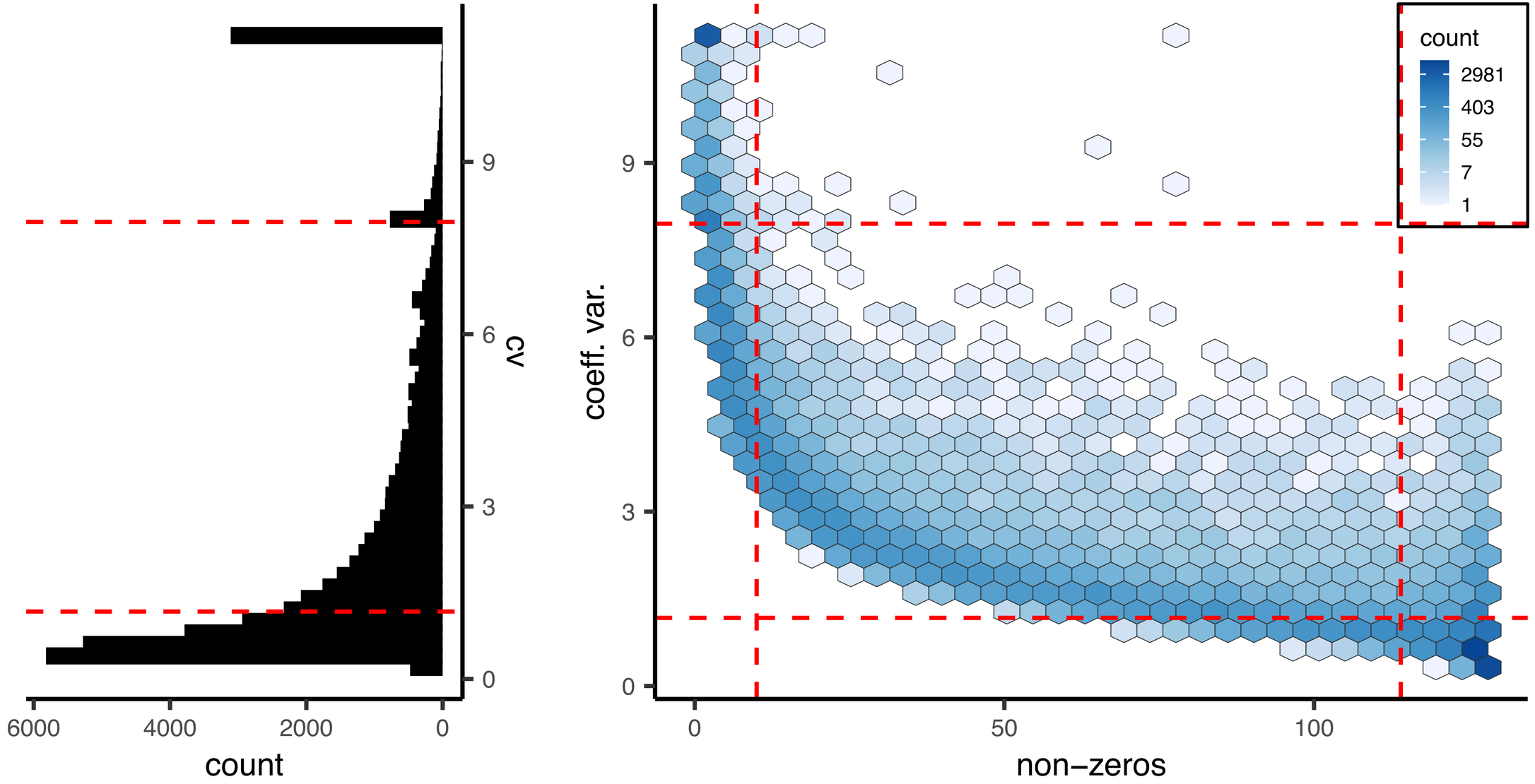


Filter out features (based on some statistics)

```
ub.cv <- quantile(.df$cv, .9) # throw away top 90% of CV
lb.cv <- quantile(.df$cv, .4) # bottom 30% of CV

lb.nz <- 10  # non-zero in at least 10 samples
ub.nz <- round(ncol(X) * .9) # avoid ubiquitously expressed

p1.1 <- p1 +
    geom_hline(yintercept = ub.cv, color="red", lty=2) +
    geom_hline(yintercept = lb.cv, color="red", lty=2) +
    geom_vline(xintercept = ub.nz, color="red", lty=2) +
    geom_vline(xintercept = lb.nz, color="red", lty=2)

p0.1 <- p0 +
    geom_hline(yintercept = ub.cv, color="red", lty=2) +
    geom_hline(yintercept = lb.cv, color="red", lty=2)
```

Filter out features (based on some statistics)

```
wrap_plots(p0.1, p1.1, nrow=1, widths=c(2,4))
```



How do we find the threshold levels?

- Scientific rationale (based on the sequencing technology)
- Communications with collaborators
- Gut feeling/experience in the field
- Be honest and transparent (never look at the desired outcome, e.g., p-hacking)
- Sensitivity analysis

Let's take a deeper look at the sub-matrix focusing on informative features

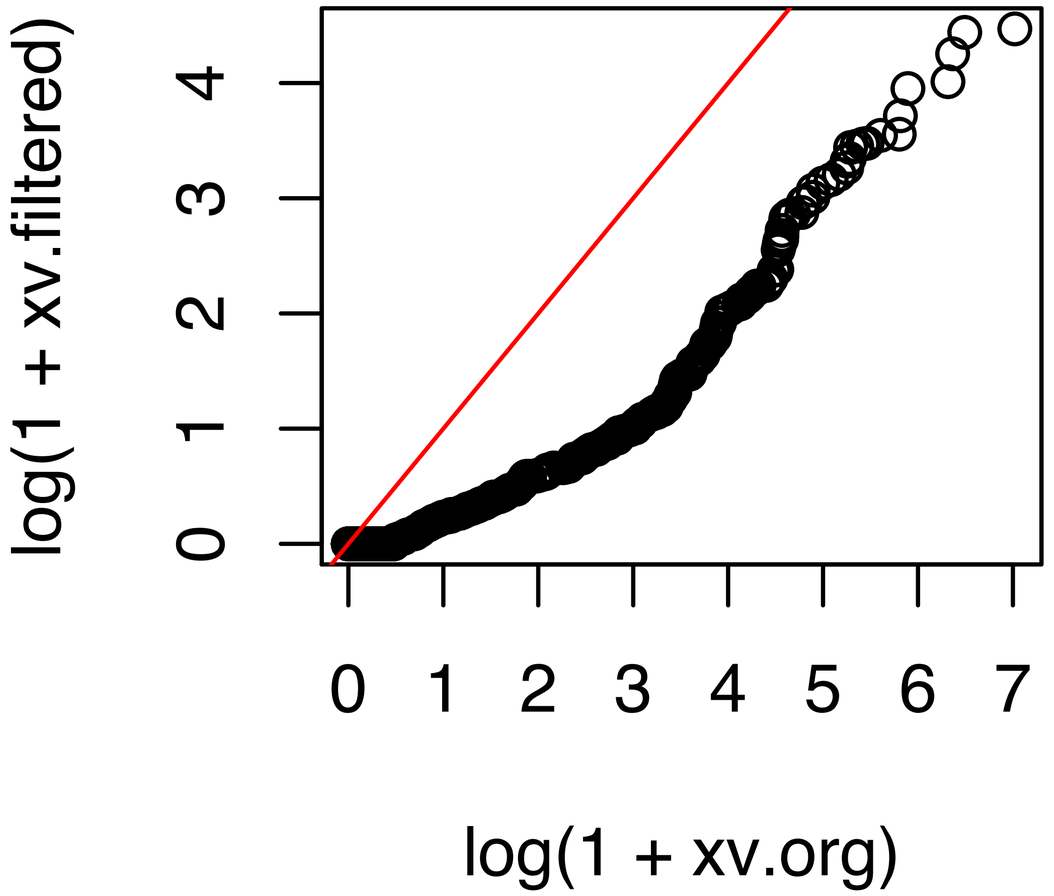
Intersection with the full data tuples

```
valid.rows <-
    row.stat %>%
    filter(`cv` < ub.cv, `cv` > lb.cv) %>%
    filter(`nz` < ub.nz, `nz` > lb.nz) %>%
    select(`Var1`)

x.melt.valid <-
    left_join(valid.rows, x.melt, by = "Var1")
```

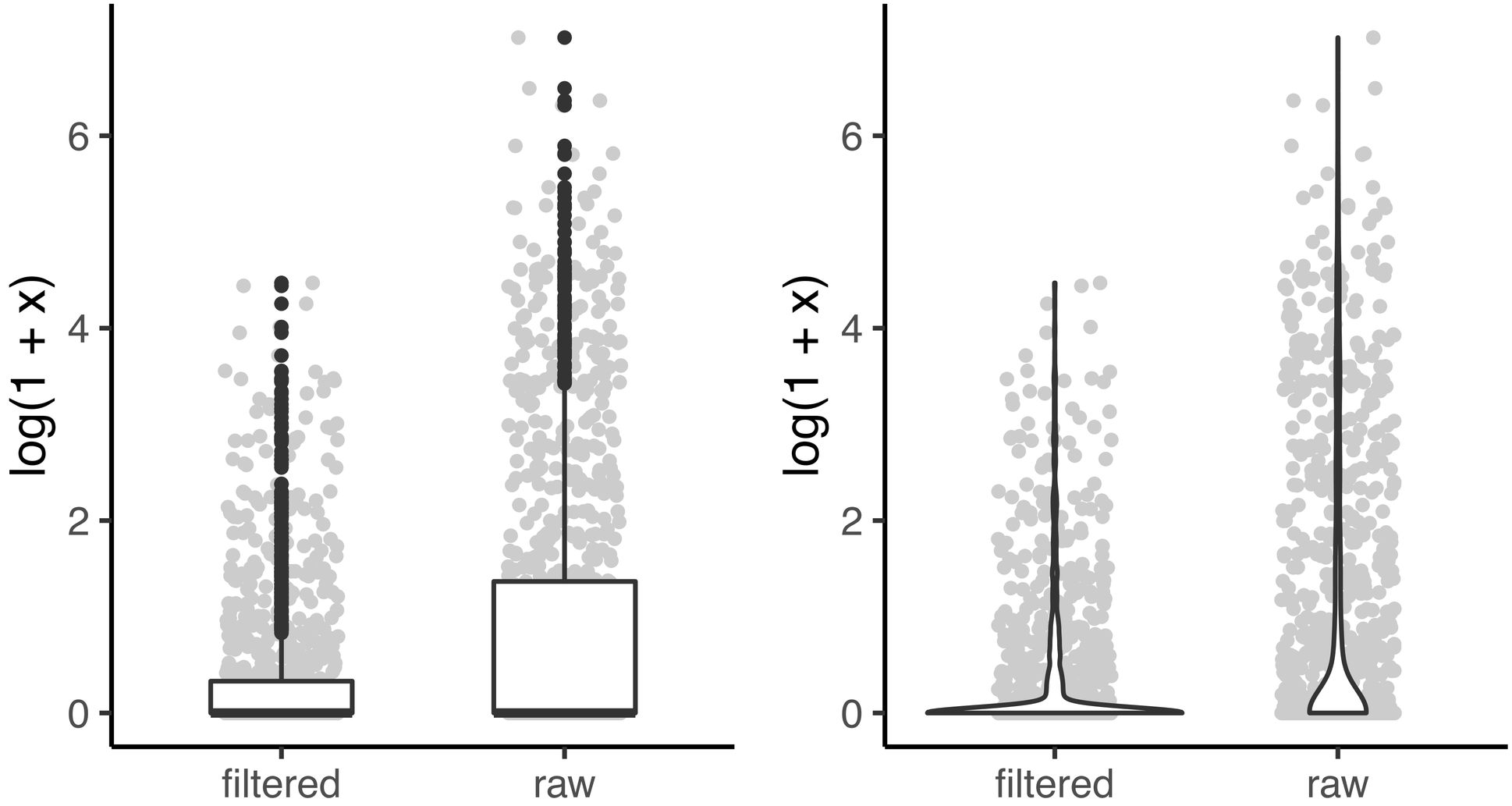
What could happen after the filtering?

```
## random 1000 ponits
xv.org <- sample(xv, 1000)
## random 1000 ponits
## among the retained
xv.filtered <-
    sample(x.melt.valid$value, 1000)
```

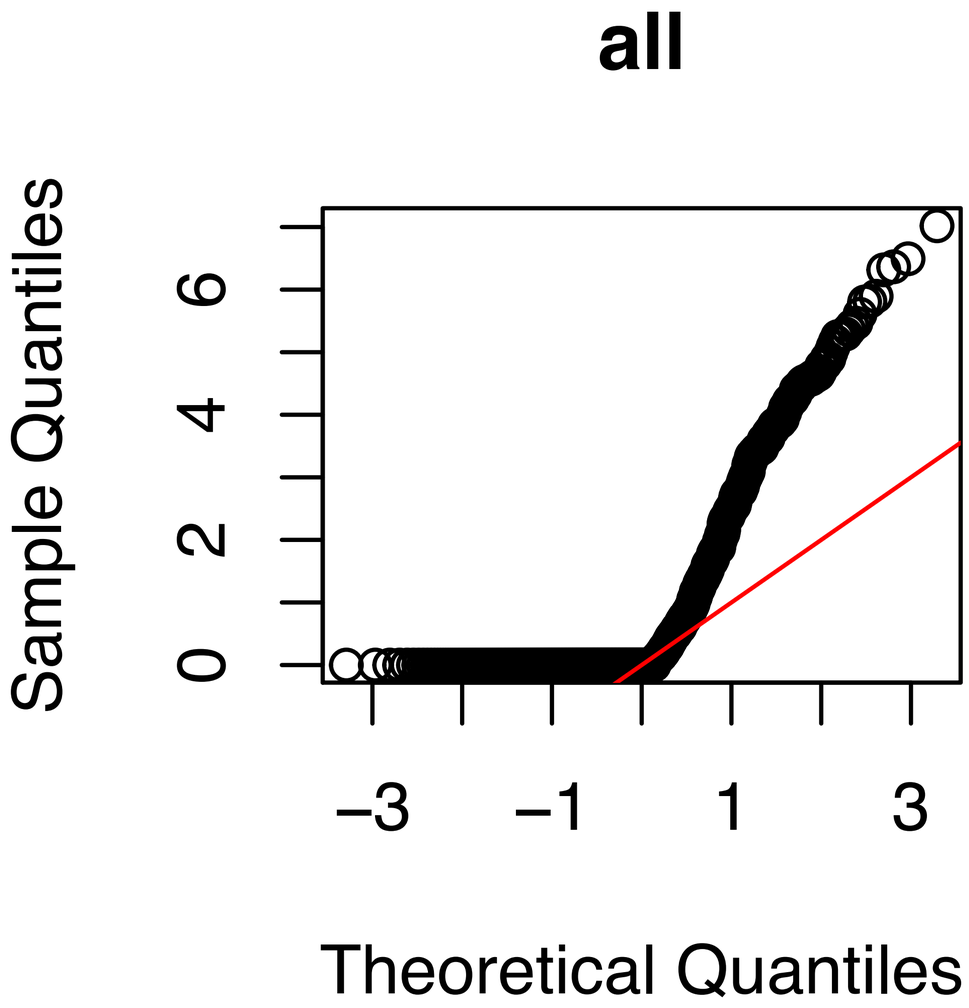
- What have we done?
- Which side is generally bigger?

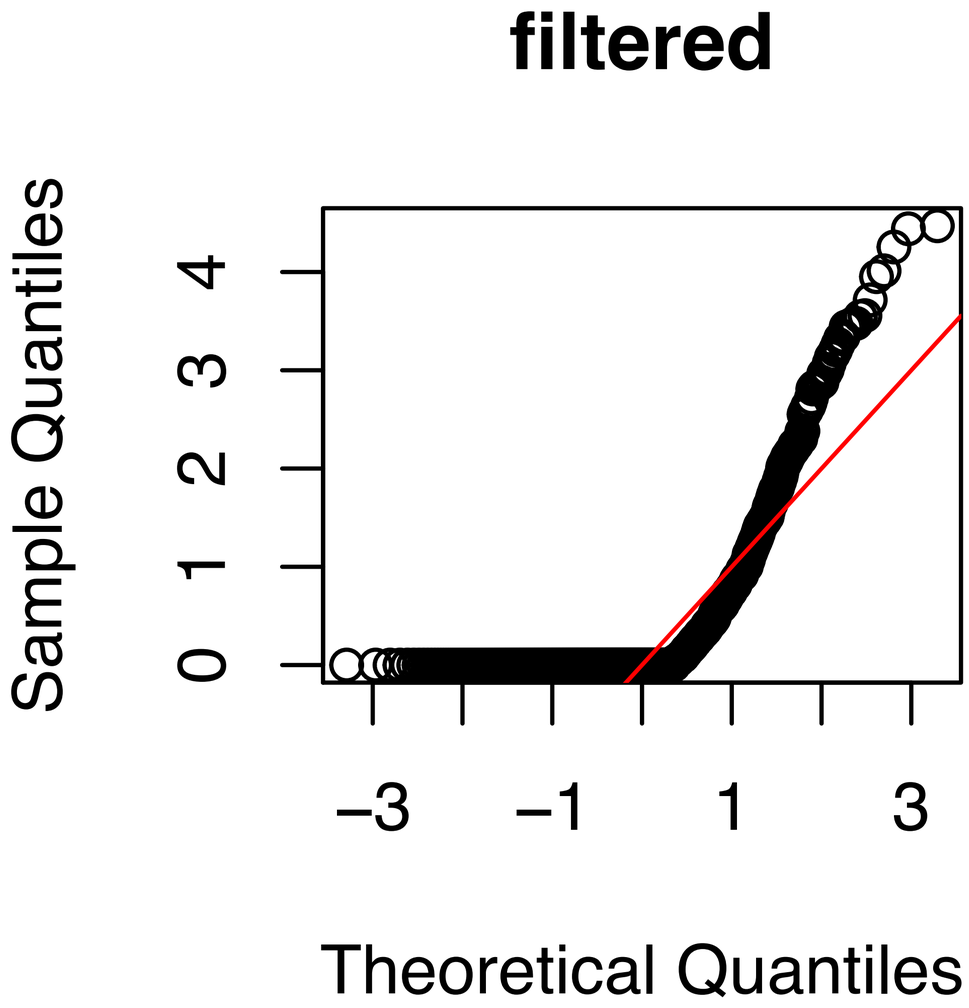


What could happen after the filtering?



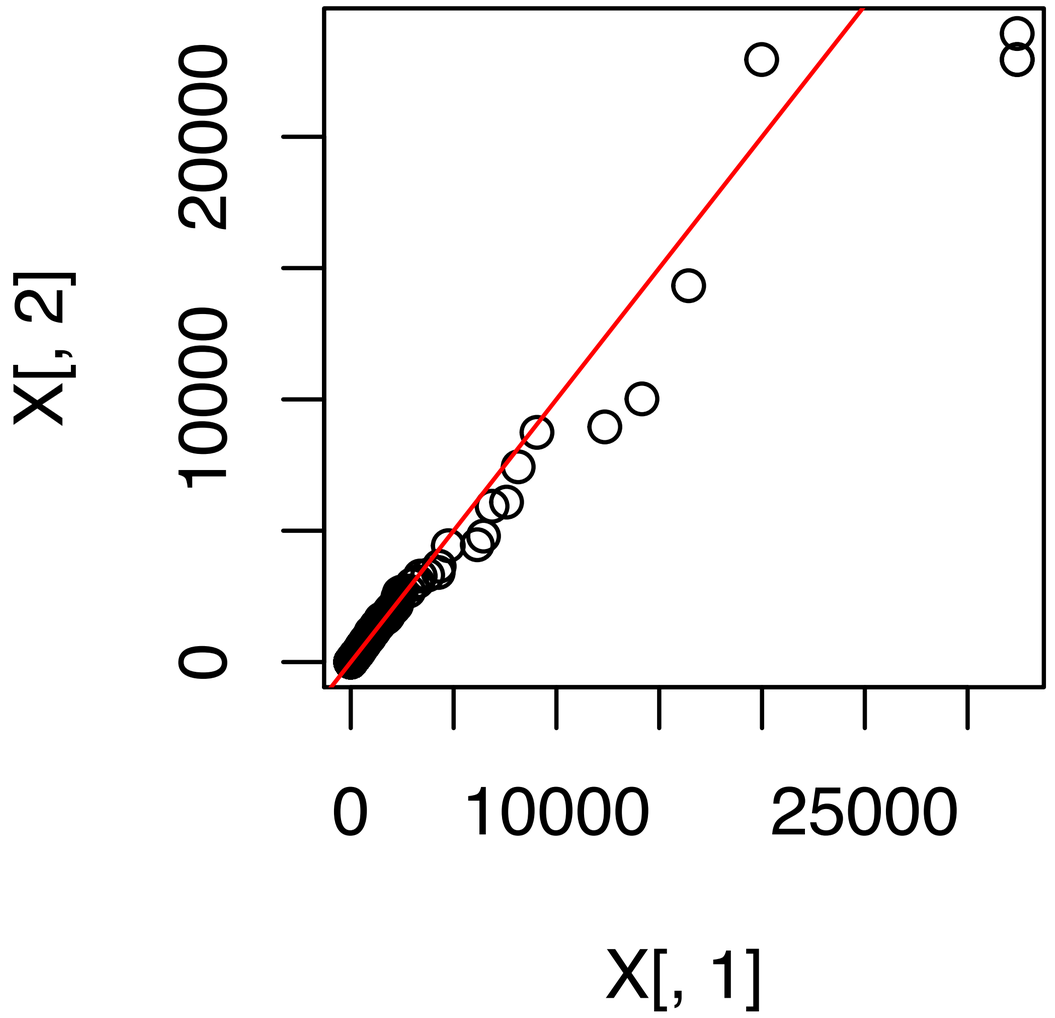
What could happen after the filtering? Normality of "log(1 + x)"



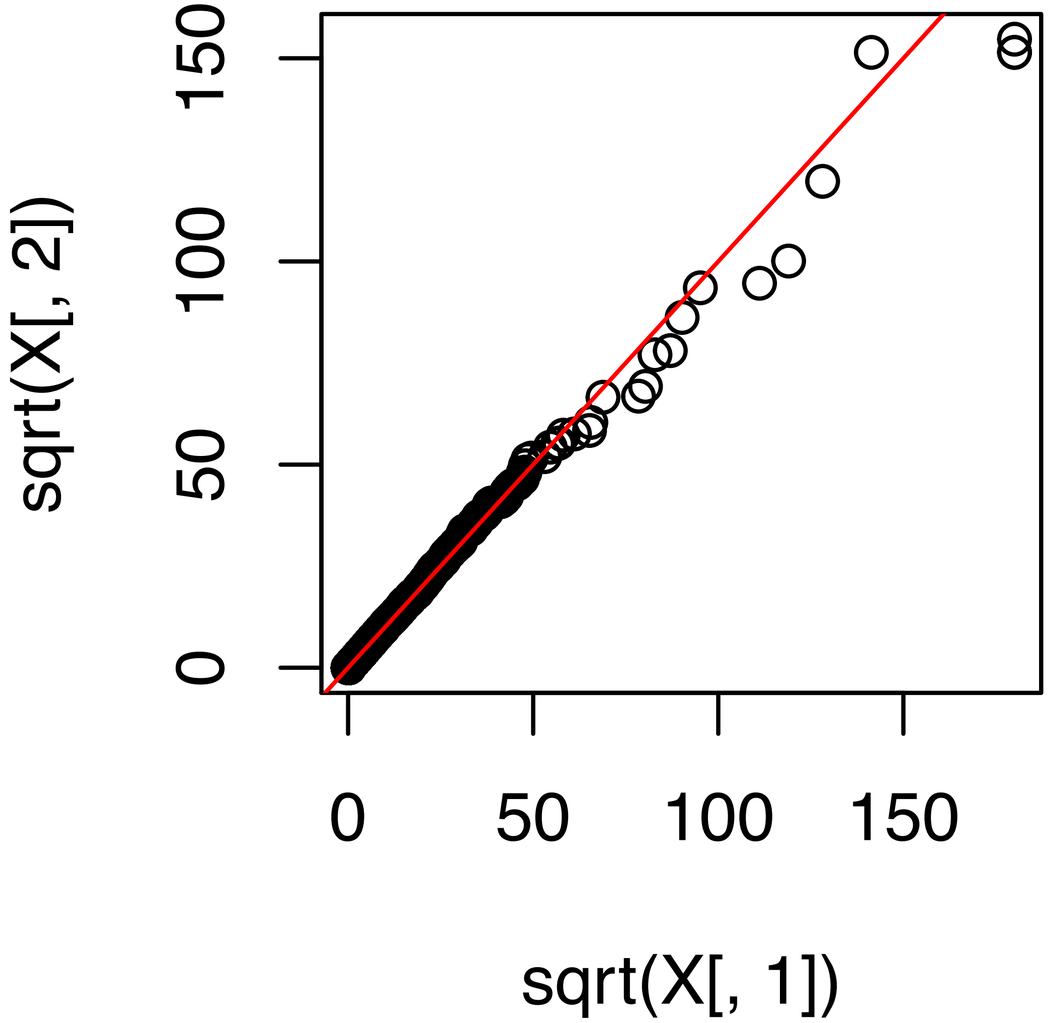


What about relationships between two columns (samples)?

```
qqplot(X[,1], X[,2])    # raw X1 and X2
abline(a=0,b=1,col=2)   # diagnoal line
```




```
qqplot(sqrt(X[,1]), sqrt(X[,2]))
abline(a=0,b=1,col=2)   # diagnoal line
```



Can we compare them in a more systematic way?

```
R.1 <- cor(X, method="pearson")
R.2 <- cor(X, method="spearman")
```

- What is the dimensionality of R?

```
dim(R.1)
```

```
## [1] 127 127
```

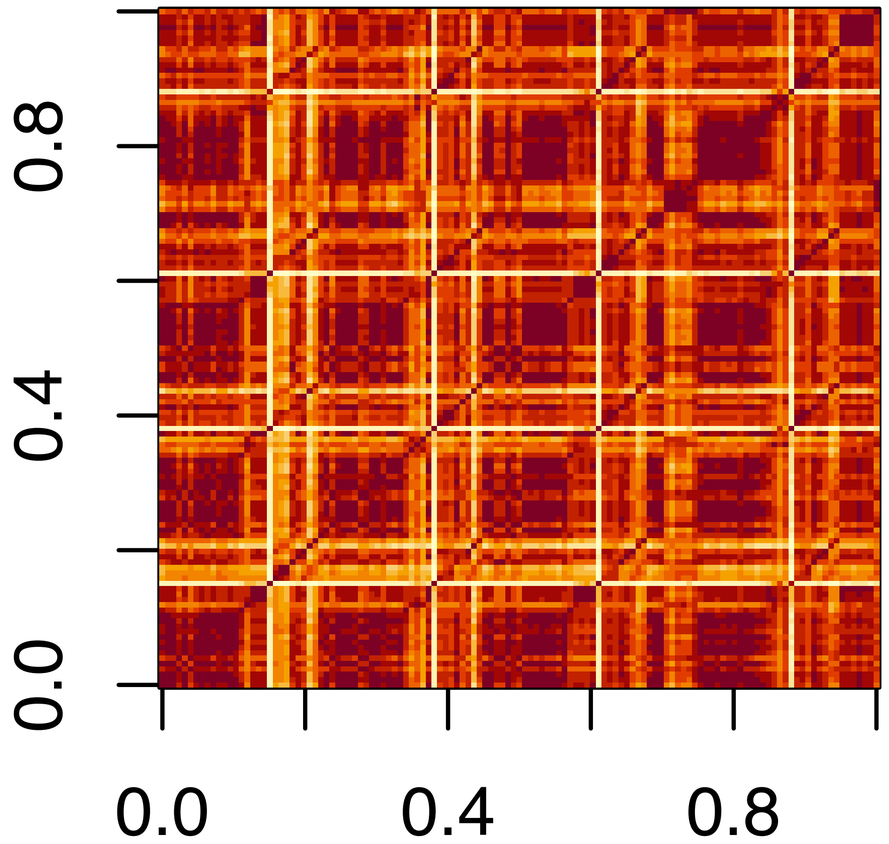
```
R.1[1:3, 1:3]
```

```
##                 DZQV_CD8_naive DZQV_CD8_CM DZQV_CD8_EM
## DZQV_CD8_naive       1.0000000   0.9659527   0.9335116
## DZQV_CD8_CM          0.9659527   1.0000000   0.9767444
## DZQV_CD8_EM          0.9335116   0.9767444   1.0000000
```

- What's the difference between the "pearson" and "spearman" methods?

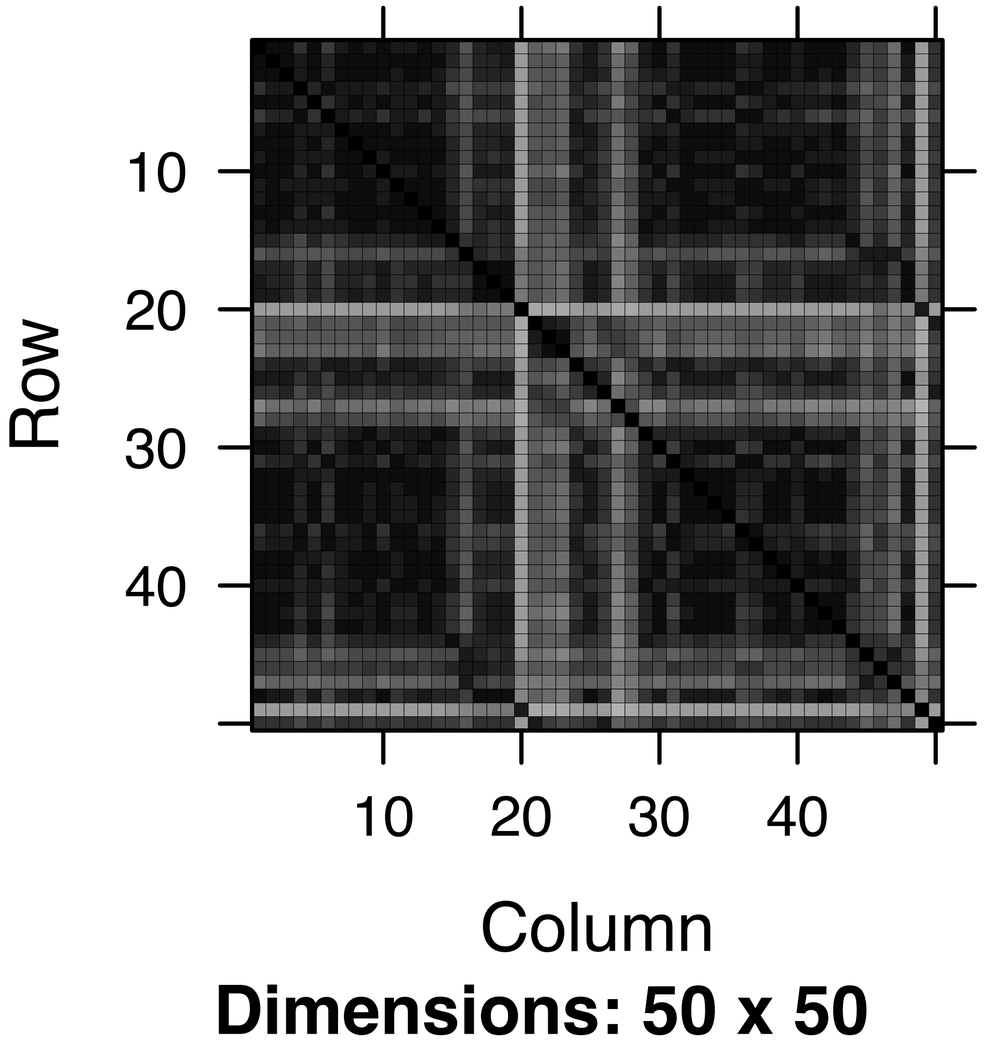
Quick visualization – base version 1

```
image(R.1)
```

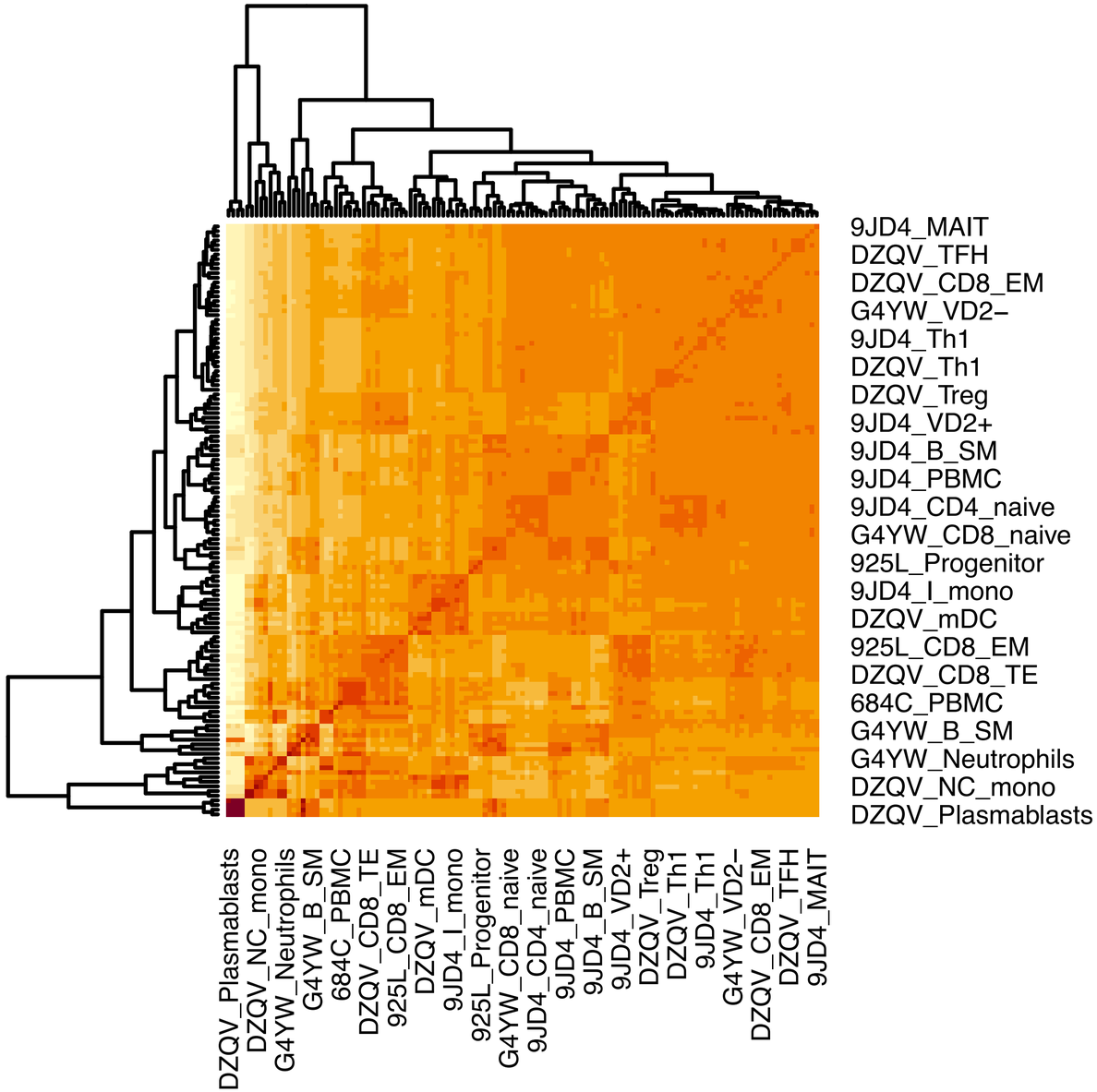


Quick visualization – base version 2

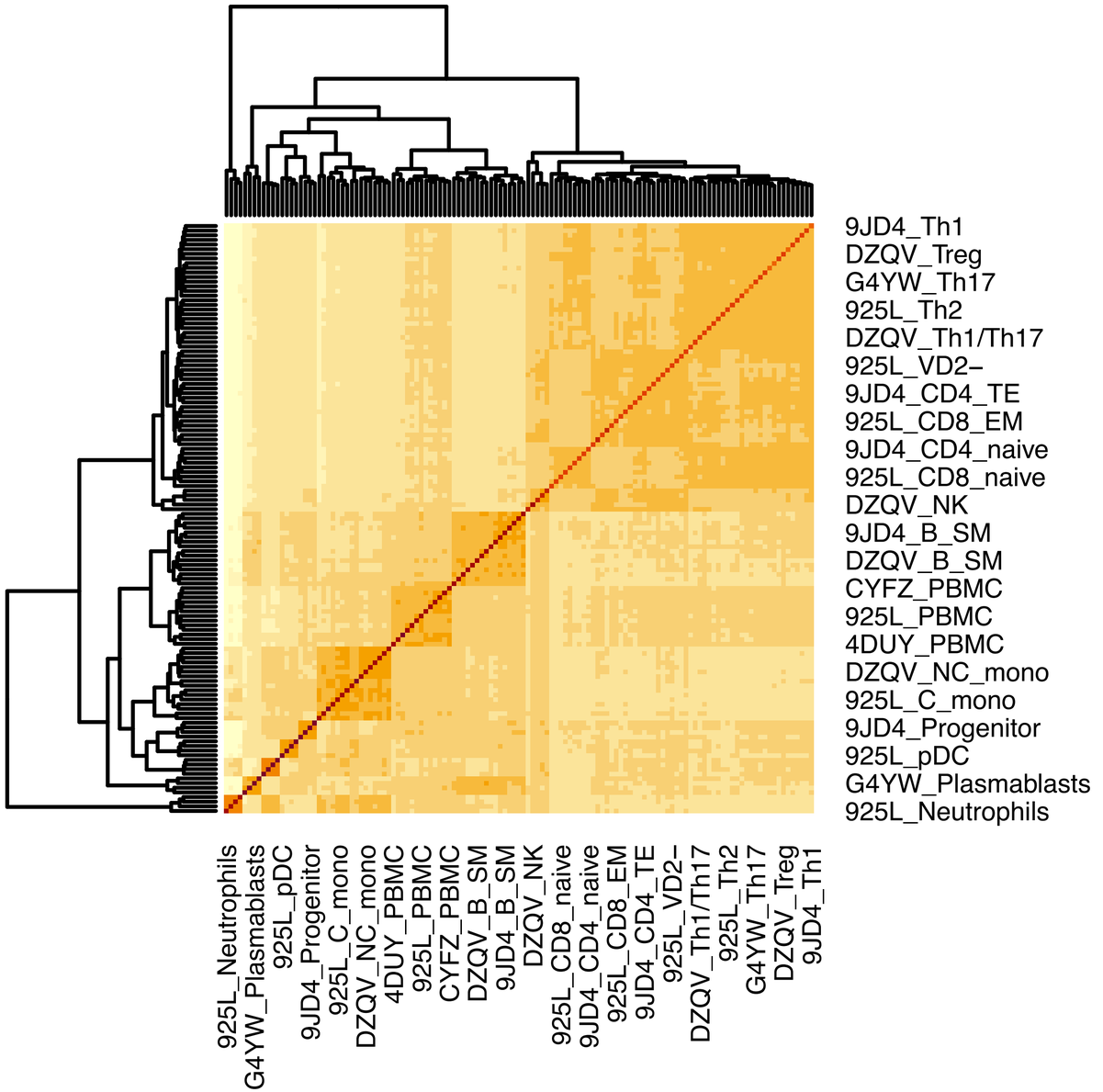
```
image(Matrix::Matrix(R.1[1:50, 1:50]))
```



We may use `heatmap` function for a small matrix (Pearson)



We may use `heatmap` function for a small matrix (Spearman)

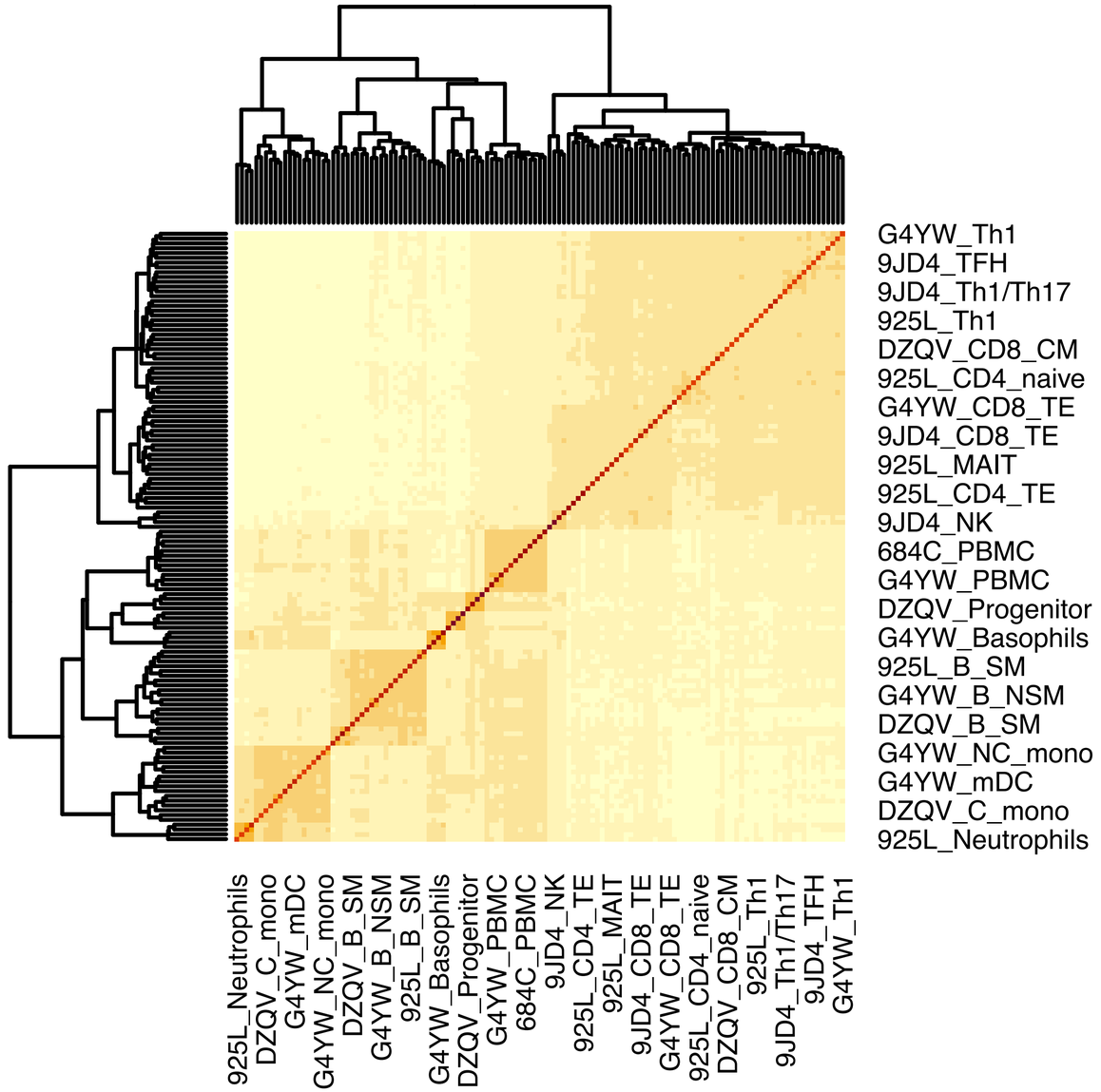


Wait, didn't we filter out some features?

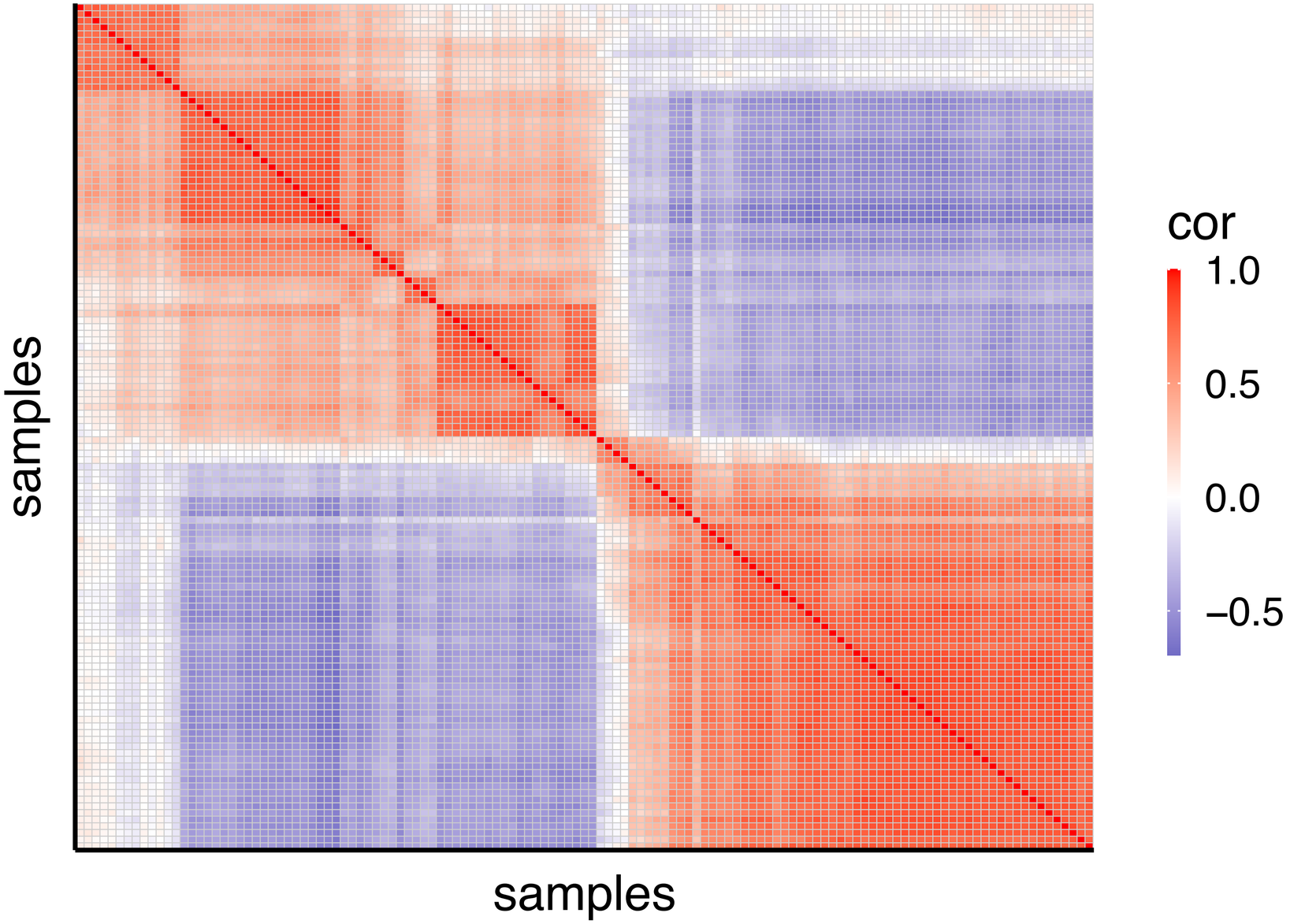
```
x.valid <- X[as.character(unlist(valid.rows)), ]
```

```
R <- cor(x.valid, method="spearman")
```

After removing the list of unwanted features



Just focusing on the top features



Today's lecture: EDA & Exp Design

- **Exploratory Data Analysis**
  - First steps to data analysis
  - Traditional (low-dimensional) approaches
  - High-dimensional methods
  - Tips on how to organize your data/project
- **Experimental Design**
  - Observational vs. Experimental studies
  - Identification of unwanted variation (SVA)
  - Causal inference: matching, stratification, inverse propensity

High-dimensional EDA methods

Clustering

- Group similar features into representative ones
- Group samples
- Lect 15

Dimensionality reduction

- Synthesize new features as a (linear) combination of existing features
- Map high-dim. data points onto latent space (Lect 12-)

What can help us in the high-dimensional setting? Essentially, our only hope is that the data is endowed with ***some form of low-dimensional structure****, one which makes it simpler than the high-dimensional view might suggest.*

Wainwright, High-dimensional Statistics

```
top.vars <-
    x.melt.valid %>%
    group_by(`Var2`) %>%
    top_n(n=15, wt=`value`) %>%
    ungroup() %>%
    select(Var1) %>%
    unique()
```

```
col.names <-
    x.melt.valid %>%
    select(`Var2`) %>%
    unique() %>%
    mutate(type = substr(`Var2`, 6, 99)) %>%
    mutate(m = `type`) %>%
    separate(`m`, c("major.type","remove"), sep="[_-]") %>%
    select(-`remove`)
```

```
head(col.names, 3)
```

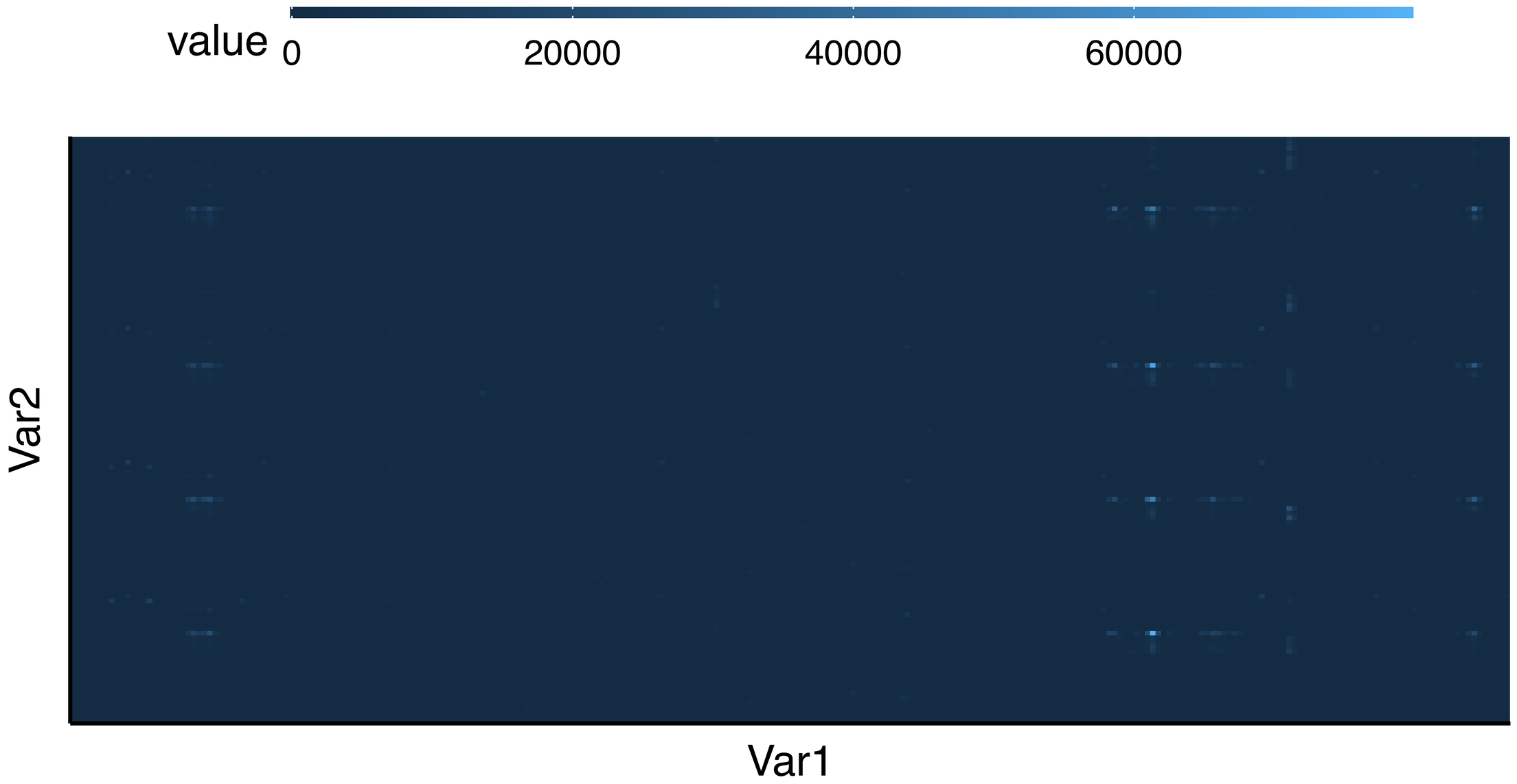
```
## # A tibble: 3 x 3
##   Var2           type      major.type
##   <fct>          <chr>     <chr>
## 1 DZQV_CD8_naive CD8_naive CD8
## 2 DZQV_CD8_CM    CD8_CM    CD8
## 3 DZQV_CD8_EM    CD8_EM    CD8
```

Drawing a heatmap for the top features

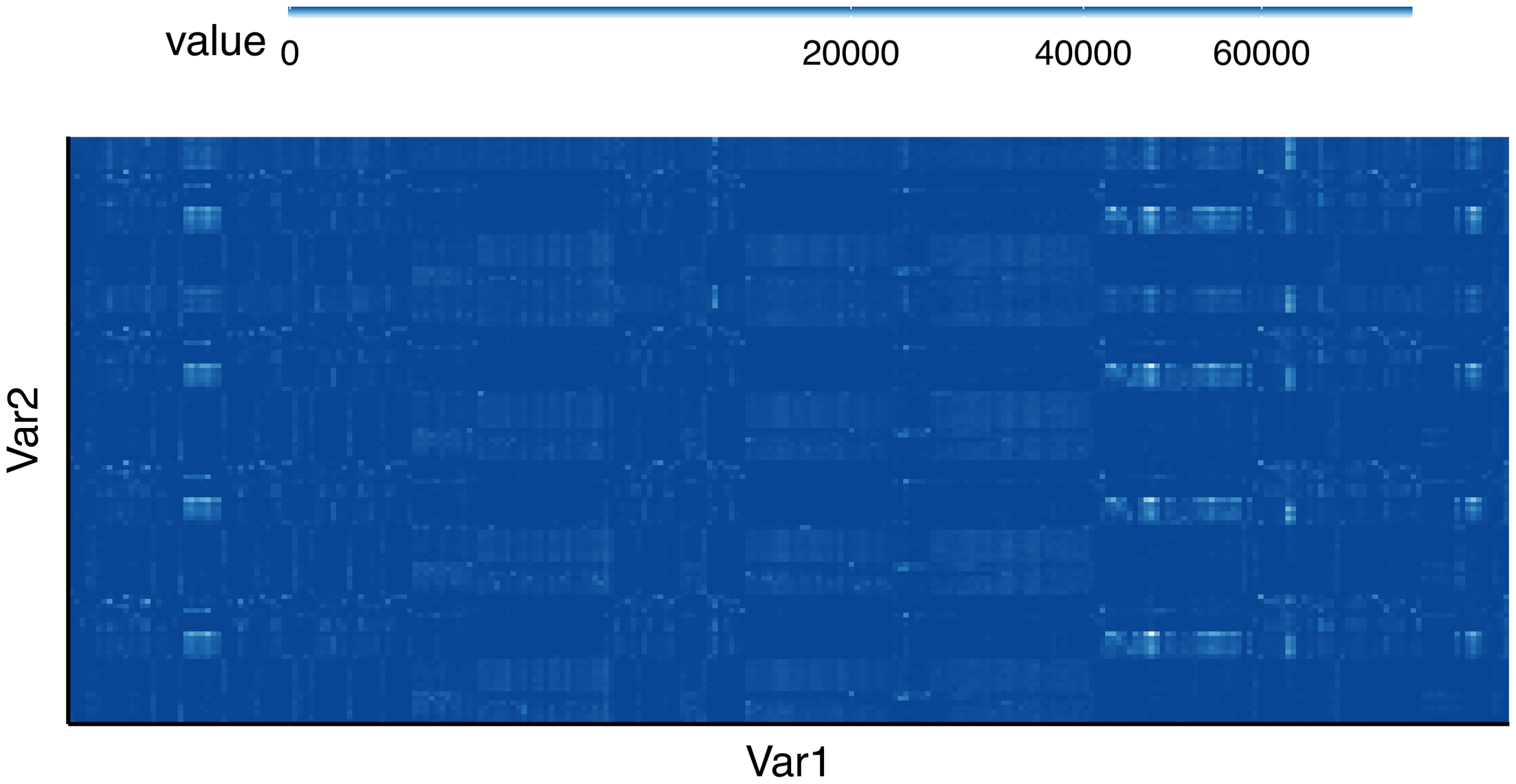
```
heat.df <- top.vars %>%
    left_join(x.melt.valid, by="Var1")
```

```
p0 <-
    ggplot(heat.df, aes(Var1, Var2, fill=value)) +
    geom_tile() +
    theme(axis.text=element_blank(), axis.ticks=element_blank()) +
    theme(legend.position = "top",
          legend.key.height = unit(.2, "lines"),
          legend.key.width = unit(4, "lines"))
```

A vanilla version



A simple transformation will provide a more informative view

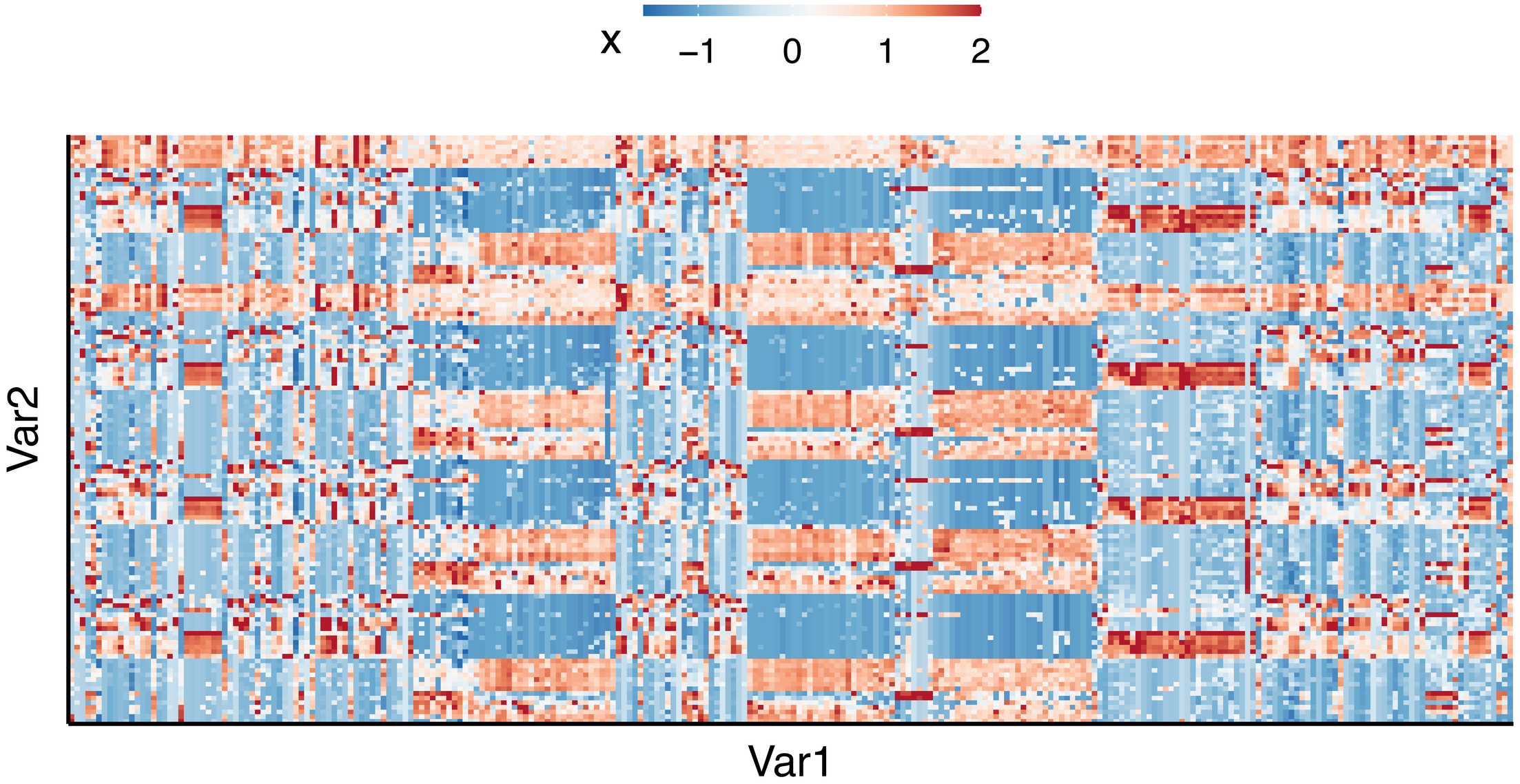


Can we normalize across samples?

```
heat.df <- heat.df %>%
    mutate(x.log = log(1 + `value`)) %>%
    group_by(`Var1`) %>%
    mutate(x.log.std = (x.log - mean(x.log))/sd(x.log)) %>%
    ungroup()
```

```
.aes <- aes(Var1, Var2, fill=pmin(pmax(`x.log.std`, -2), 2))
p1 <-
    ggplot(heat.df, .aes) +
    geom_tile() +
    theme(axis.text=element_blank(), axis.ticks=element_blank()) +
    theme(legend.position = "top",
          legend.key.height = unit(.2, "lines"))
```

Can we normalize across samples



Sorting columns by cell type (Var2)

```
heat.df <- heat.df %>% left_join(col.names)
```

```
v2.order <-
    heat.df %>%
    select(`type`, `Var2`) %>%
    arrange(`type`) %>%
    unique() %>%
    select(`Var2`) %>%
    unlist() %>%
    as.character()
```

```
head(v2.order, 3)
```

```
## [1] "DZQV_B_Ex" "925L_B_Ex" "9JD4_B_Ex"
```

Sorting columns by cell type (Var2)

Recover the data matrix after sorting the rows by v2.order

```
M <- heat.df %>%
    select(Var1, Var2, x.log.std) %>%
    mutate(Var2 = factor(Var2, v2.order)) %>%
    spread(Var1, value=x.log.std, fill = 0)
```

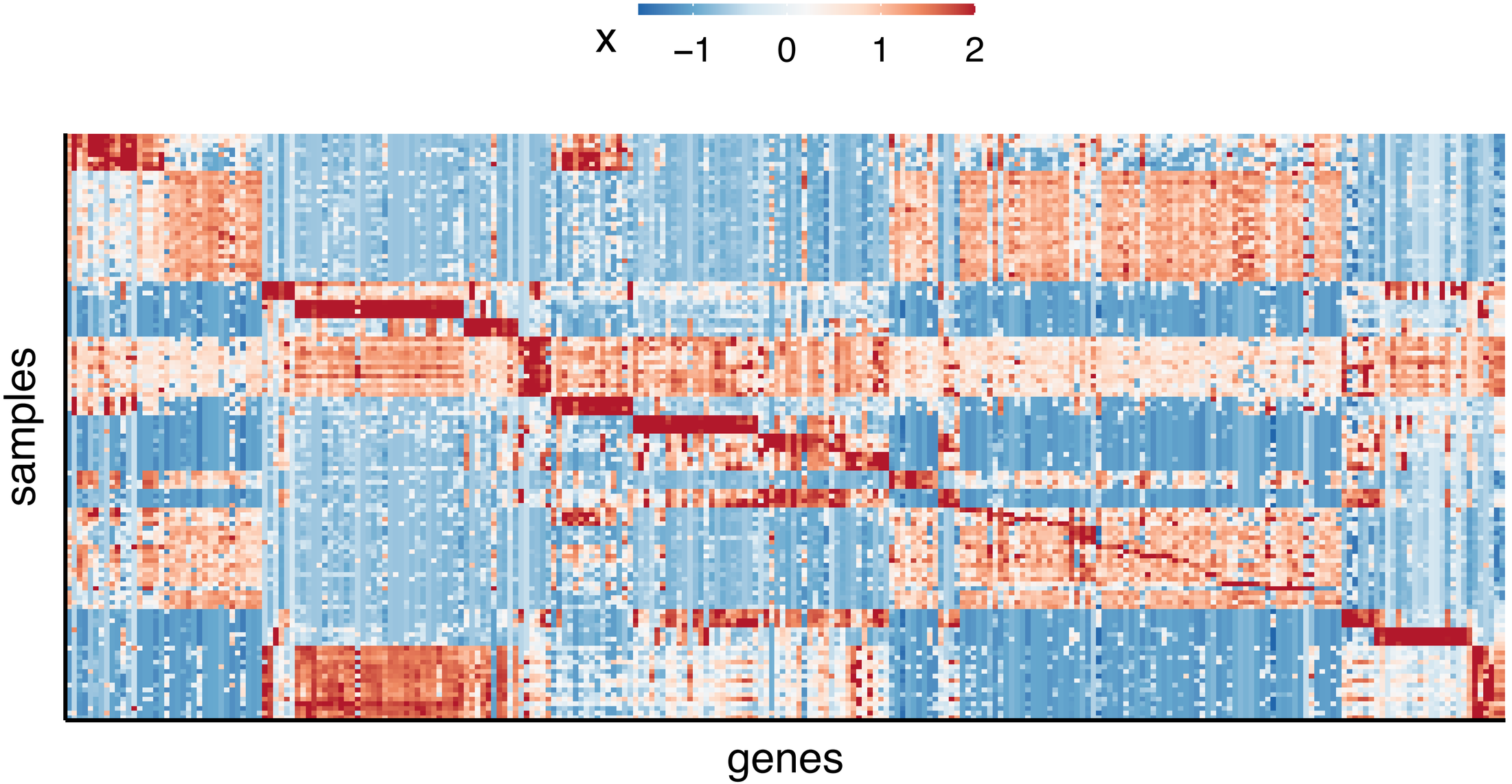
Diagnoalize the order of columns by maximum value positions

```
argmax.v2.index <- apply(M[, -1], 2, which.max)
.order <- order(argmax.v2.index, decreasing=TRUE)
v1.order <- colnames(M)[-1][.order]
```

Plotting the heatmap after sorting the rows and columns

```
heat.df <- heat.df %>%
    mutate(v1.sorted = factor(Var1, v1.order)) %>%
    mutate(v2.sorted = factor(Var2, v2.order))

.aes <- aes(v1.sorted, v2.sorted, fill=pmin(pmax(`x.log.std`, -2), 2))

p2 <-
    ggplot(heat.df, .aes) +
    geom_tile() +
    theme(axis.text=element_blank(), axis.ticks=element_blank()) +
    theme(legend.position = "top",
          legend.key.height = unit(.2, "lines"))
```

Plotting the heatmap after sorting the rows and columns

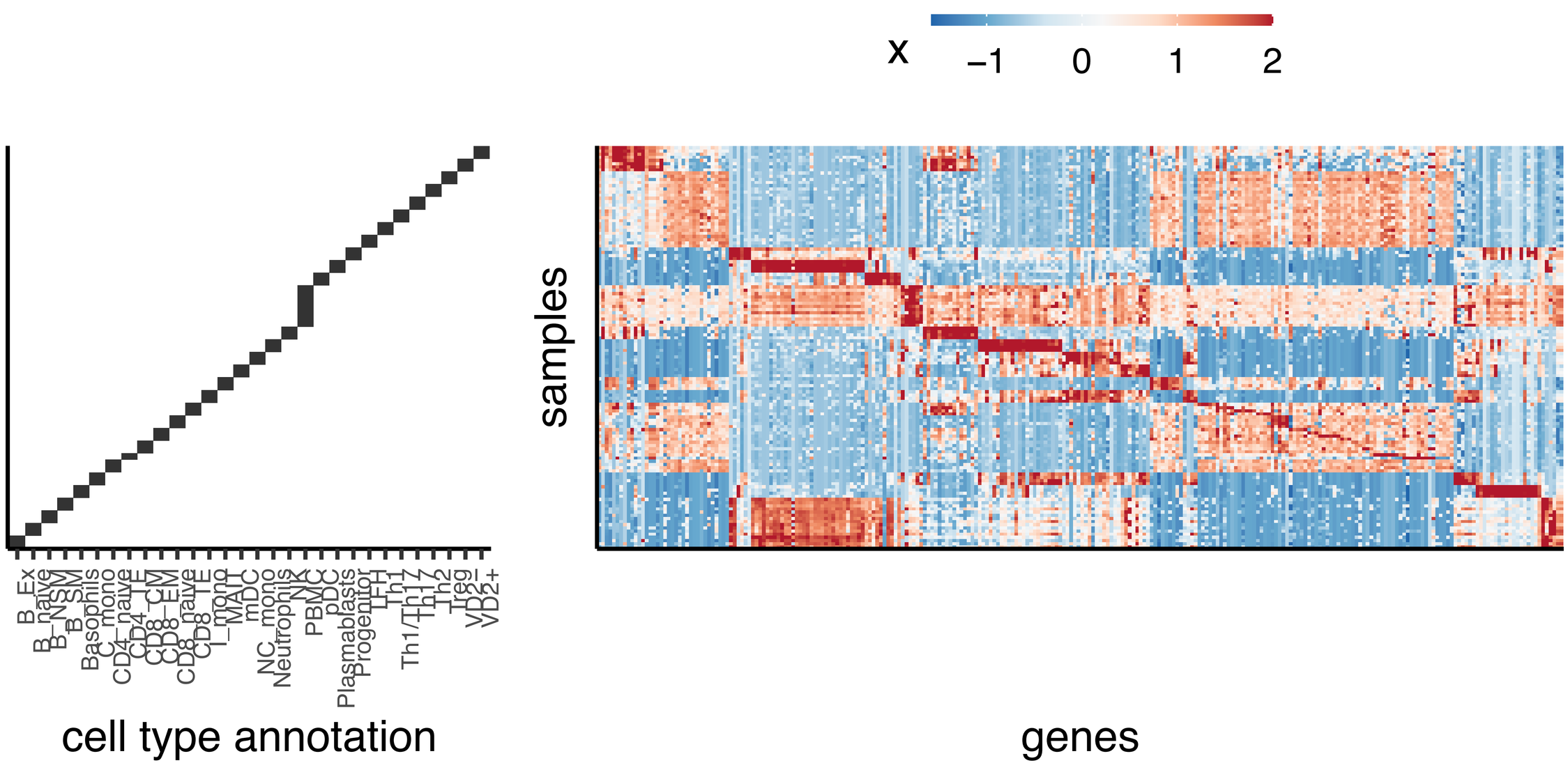


Take one step forward compare ...

```
annot.df <-
    col.names %>%
    mutate(v1.sorted = factor(type)) %>%
    mutate(v2.sorted = factor(Var2, v2.order))
```

```
.aes <- aes(v1.sorted, v2.sorted)

p2.1 <-
    ggplot(annot.df, .aes) + geom_tile() + xlab("cell type annotation") +
    theme(axis.title.y = element_blank(),
          axis.text.y = element_blank(),
          axis.ticks.y = element_blank(),
          axis.text.x = element_text(size=6, angle=90, vjust=1, hjust=1))

p2.2 <- p2 +
    scale_fill_distiller("x", palette = "RdBu", direction=-1) +
    ylab("samples") + xlab("genes")
```

Compare this heatmap with known cell type annotations



High-dimensional EDA methods

Clustering

- Group similar features into representative ones
- Group samples
- Lect 15

Dimensionality reduction

- Synthesize new features as a (linear) combination of existing features
- Map high-dim. data points onto latent space (Lect 12-)

What can help us in the high-dimensional setting? Essentially, our only hope is that the data is endowed with ***some form of low-dimensional structure****, one which makes it simpler than the high-dimensional view might suggest.*

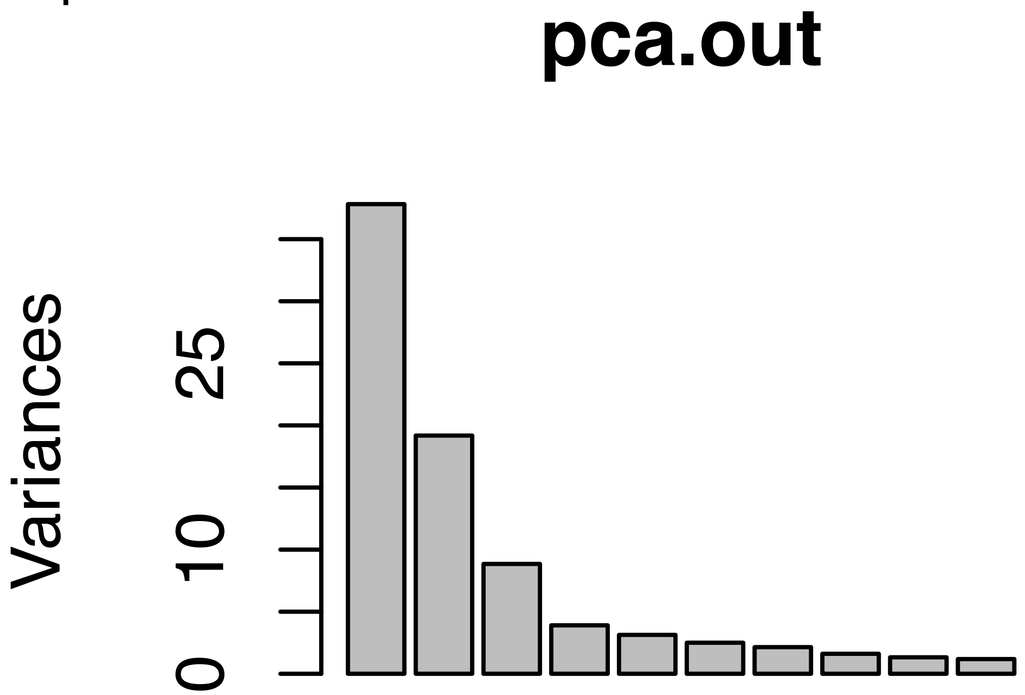
Wainwright, High-dimensional Statistics

Principal component analysis

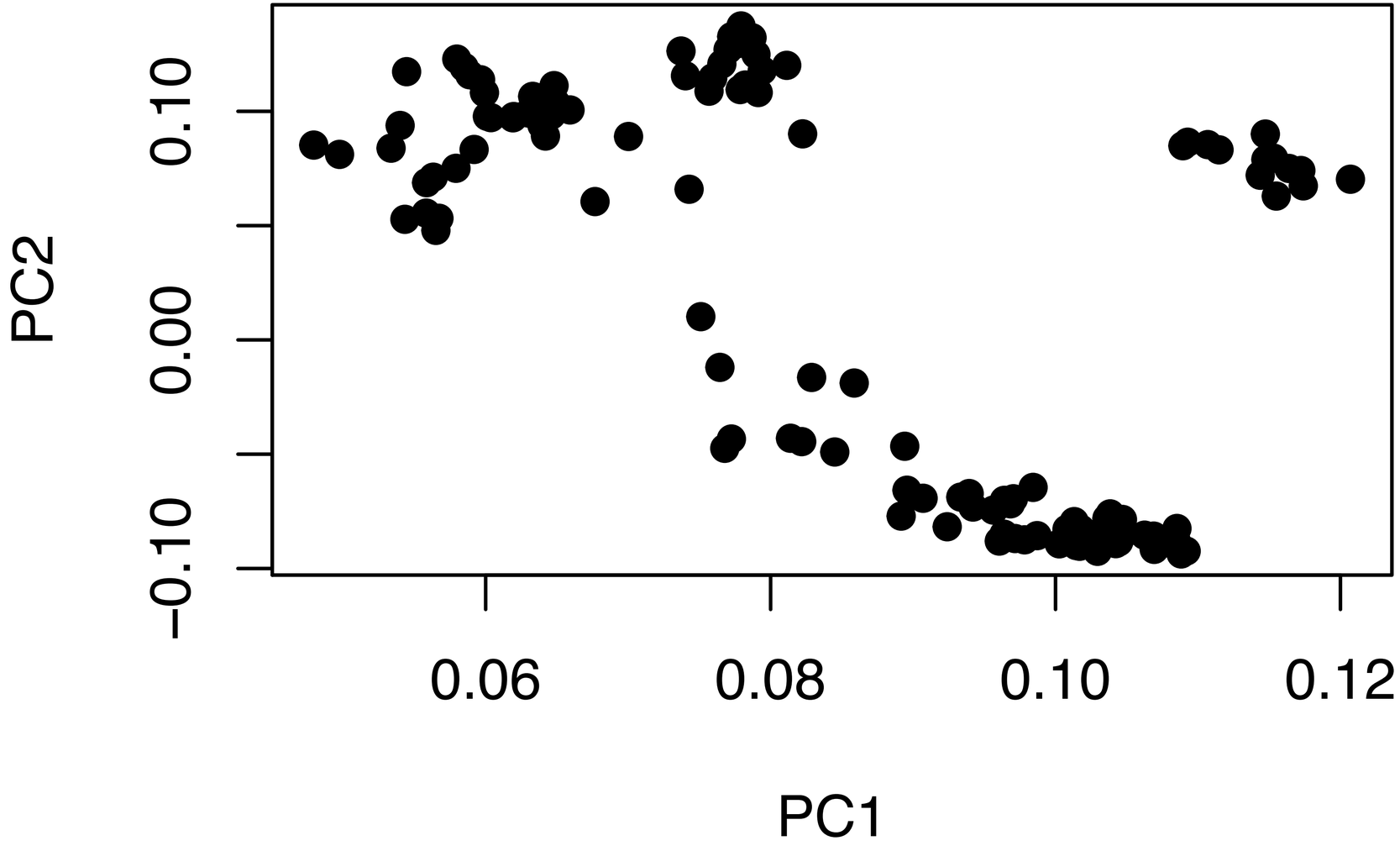
```
xx <- log(1 + x.valid)
## Note: measure PCs for the columns
pca.out <- prcomp(xx, scale.=TRUE)

## Extract the output
pcs <- pca.out$rotation[, 1:3]

head(pcs, 3)
```

```
##                          PC1          PC2           PC3
## DZQV_CD8_naive 0.09239707 -0.08174110  0.012668092
## DZQV_CD8_CM    0.10112809 -0.08644983  0.005001030
## DZQV_CD8_EM    0.10360903 -0.07807551 -0.001014498
```

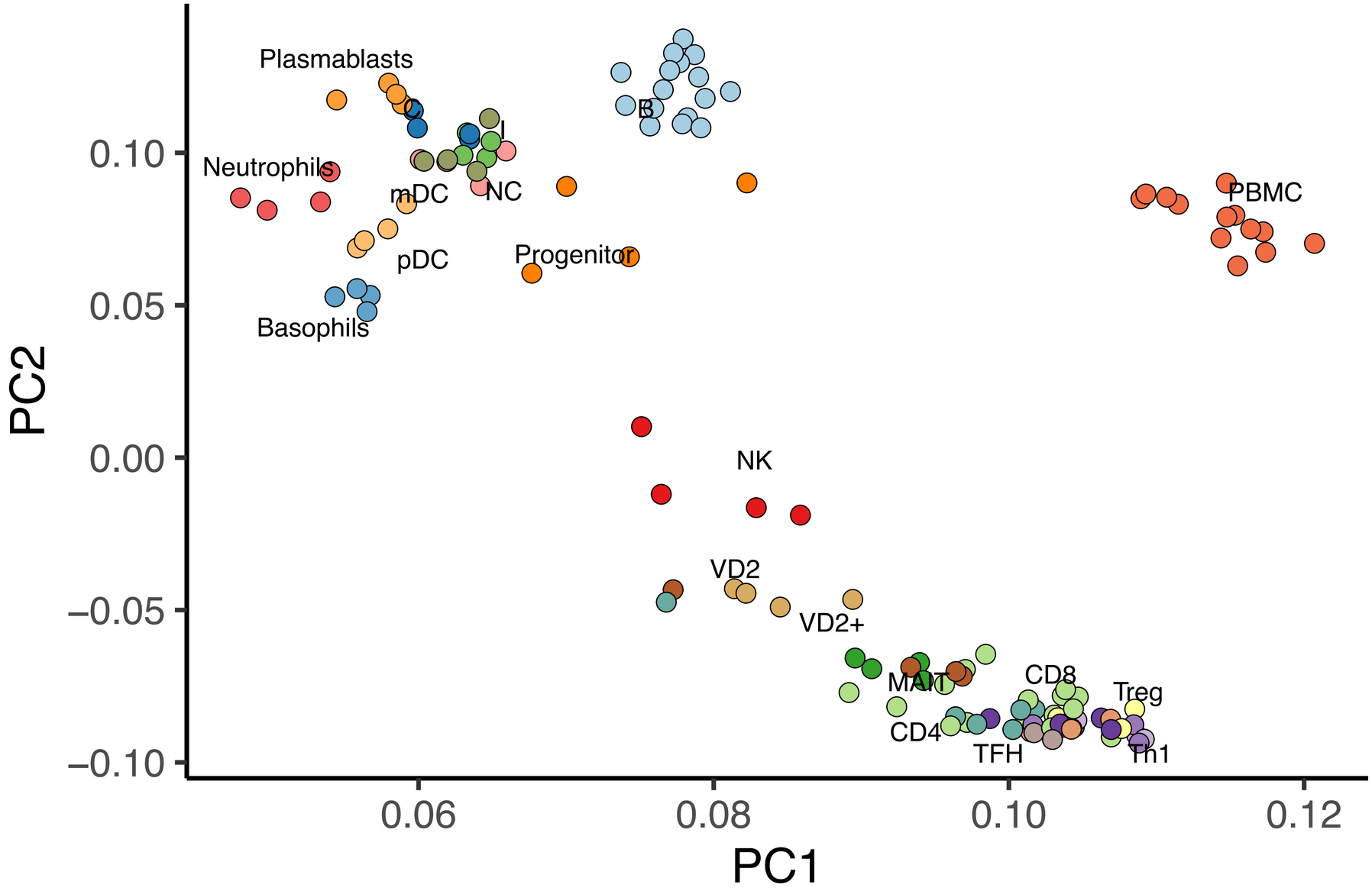
How much variance was explained by each component?



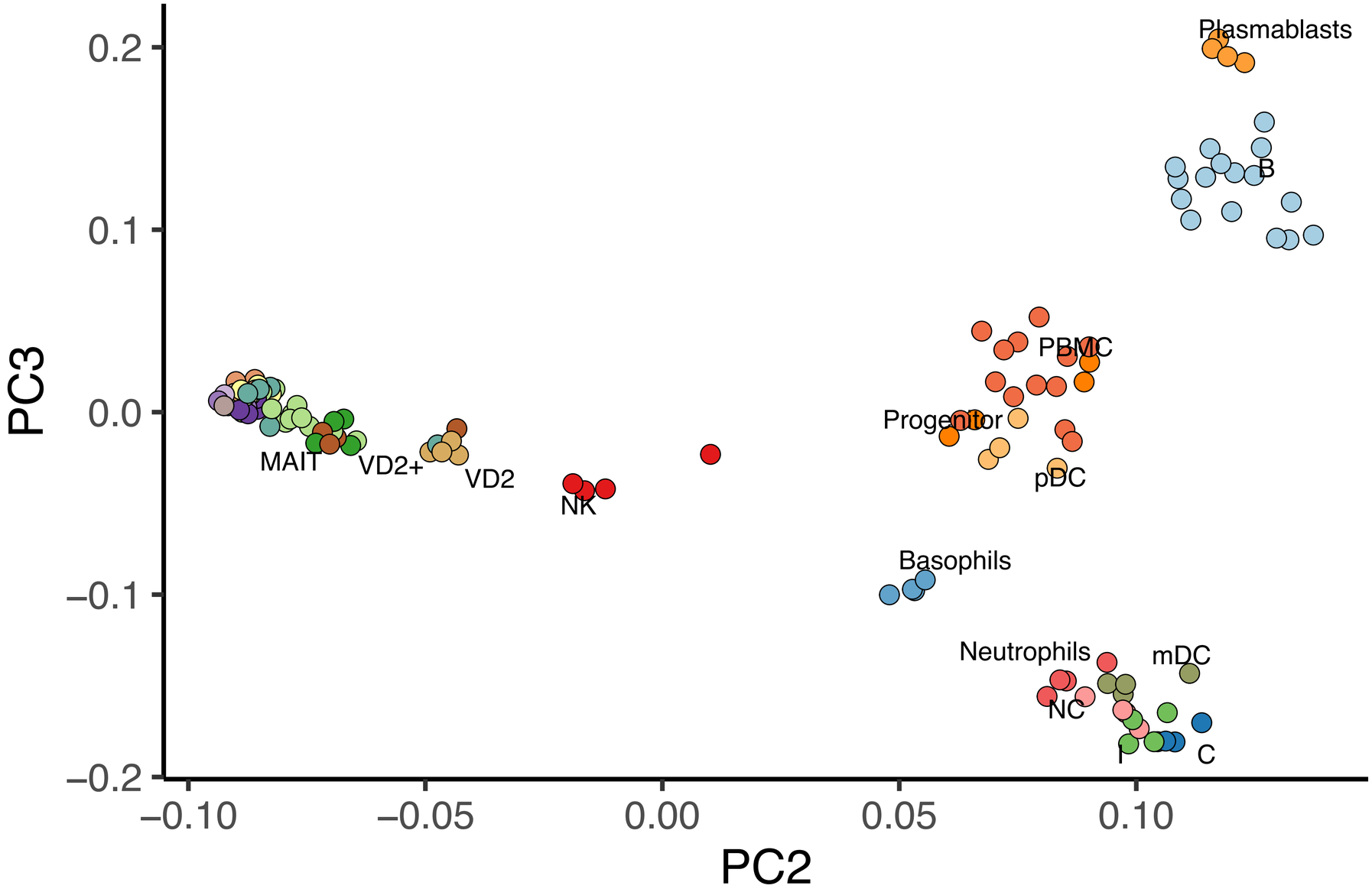
Top principal components



Top principal components



Top principal components

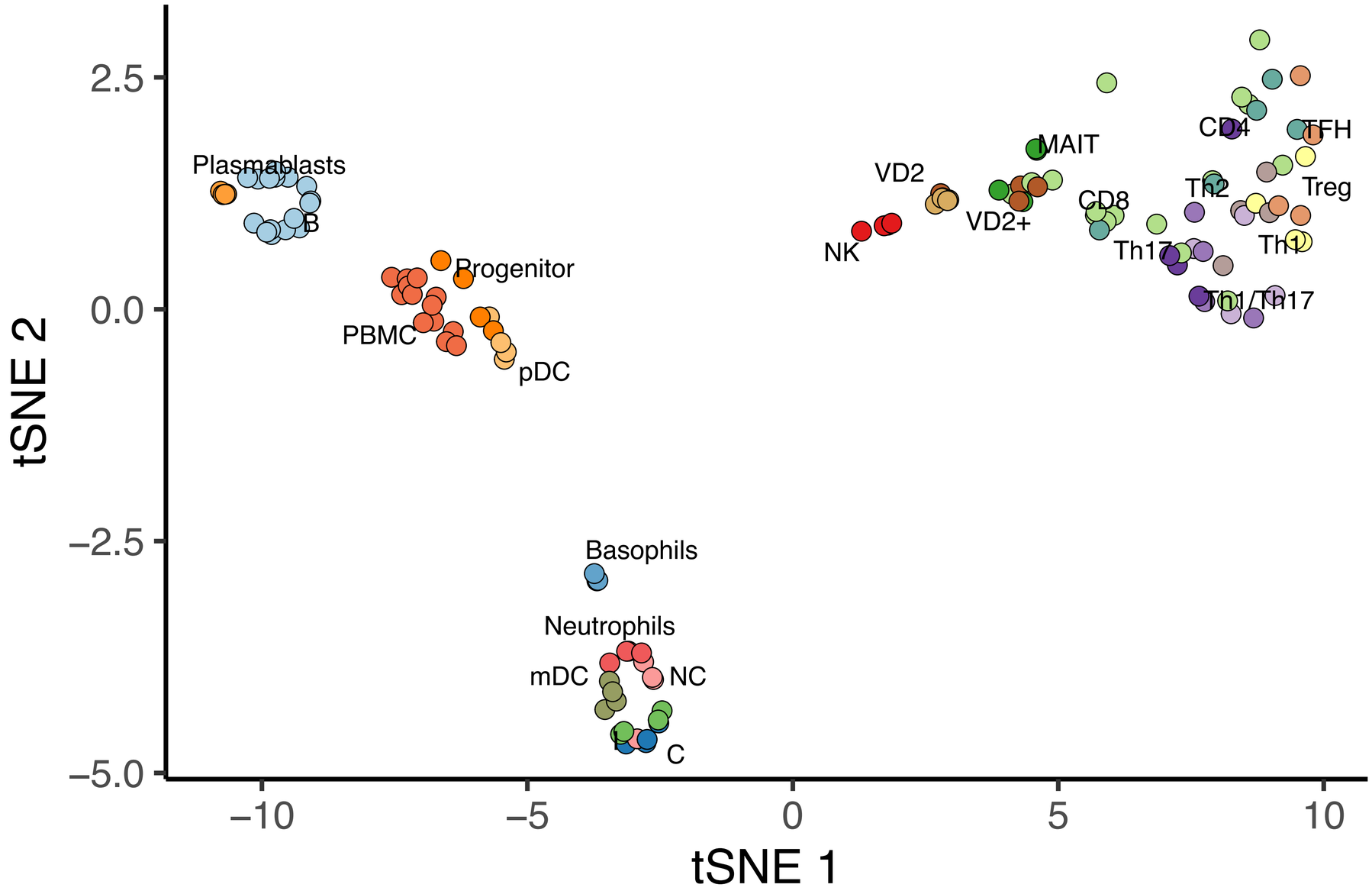


t-SNE: Other dimensionality reduction methods

```
tsne <- Rtsne::Rtsne(pcs, dims = 2)

.df <- tsne$Y %>%
    as_tibble() %>%
    mutate(Var2=rownames(pcs)) %>%
    left_join(col.names, by = "Var2")

.centers <- .df %>%
    group_by(major.type) %>%
    summarize(V1=mean(V1), V2=mean(V2)) %>%
    ungroup()
```

We will discuss PCA and tSNE later ...

t-SNE: Other dimensionality reduction methods



Today's lecture: EDA & Exp Design

- **Exploratory Data Analysis**
  - First steps to data analysis
  - Traditional (low-dimensional) approaches
  - High-dimensional methods
  - Tips on how to organize your data/project
- **Experimental Design**
  - Observational vs. Experimental studies
  - Identification of unwanted variation (SVA)
  - Causal inference: matching, stratification, inverse propensity

General rules for organizing your project

- Treat your data sacred (keep them as raw as possible).
- Never manipulate your data points manually!
- Literate programming (variable and function names should make sense)
- Strive to make your working directory organized
- Use one R markdown file for one figure.
- Do not postpone your interpretation until paper writing.
- Separate data analysis from plotting routines.

Tips for visualization

- Ask: What am I trying to show with this picture?
- Generate pictures that you can explain in words.
- Be kind to annotate all the axes' names.
- Additionally, annotate helpful information if needed.
- Avoid 3d or higher dimensional pictures (your paper is 2d)
- Build your own graphing library

Today's lecture: EDA & Exp Design

- **Exploratory Data Analysis**
  - First steps to data analysis
  - Traditional (low-dimensional) approaches
  - High-dimensional methods
  - Tips on how to organize your data/project
- **Experimental Design**
  - Observational vs. Experimental studies
  - Identification of unwanted factors
  - Causal inference: matching, stratification, inverse propensity

Why do we care about experimental design?

Make the best out of budget

Reduce variability along any unwanted axes/factors

vs.

Magnify variability along relevant axes (e.g., disease)

Causal discovery + validatation

Design your statistical model identifiable!

Measure causal effects of XXX

Observation vs. experimentation

- X and Y are correlated.
- I recovered from a fever (Y) after taking ibuprofen (X) last night.
- We found *BRCA* genes highly expressed in the metastasized breast cancer samples.
- Seeing the results Y given X

- X causes Y.
- Had I not taken medicine (X), I would still have a fever (Y).
- We knocked down *BRCA* genes and mitigated cancer progression in patient-derived xenograft models.
- Doing X to make changes in Y

Code available in our GitHub repo

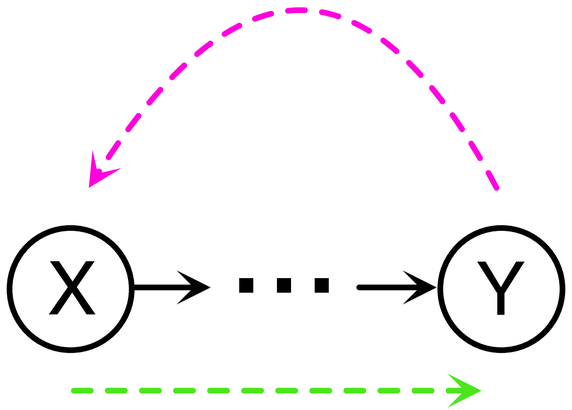
https://github.com/STAT540-UBC/lectures/blob/main/lect03-eda/causality.Rmd

Today’s lecture: EDA & Exp Design

- **Exploratory Data Analysis**
  - First steps to data analysis
  - Traditional (low-dimensional) approaches
  - High-dimensional methods
  - Tips on how to organize your data/project
- **Experimental Design**
  - Observational vs. Experimental studies
  - Identification of unwanted factors
  - Causal inference: matching, stratification, inverse propensity

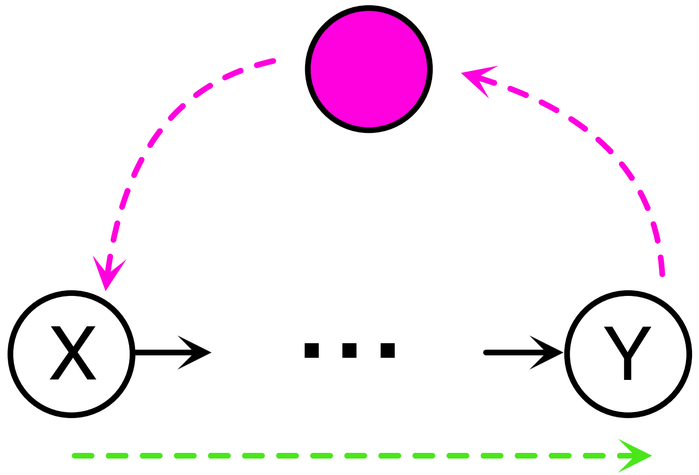
“Backdoor” exercise

Question: Which nodes should be “conditioned” and/or “adjusted” to block a reverse path from Y to X?

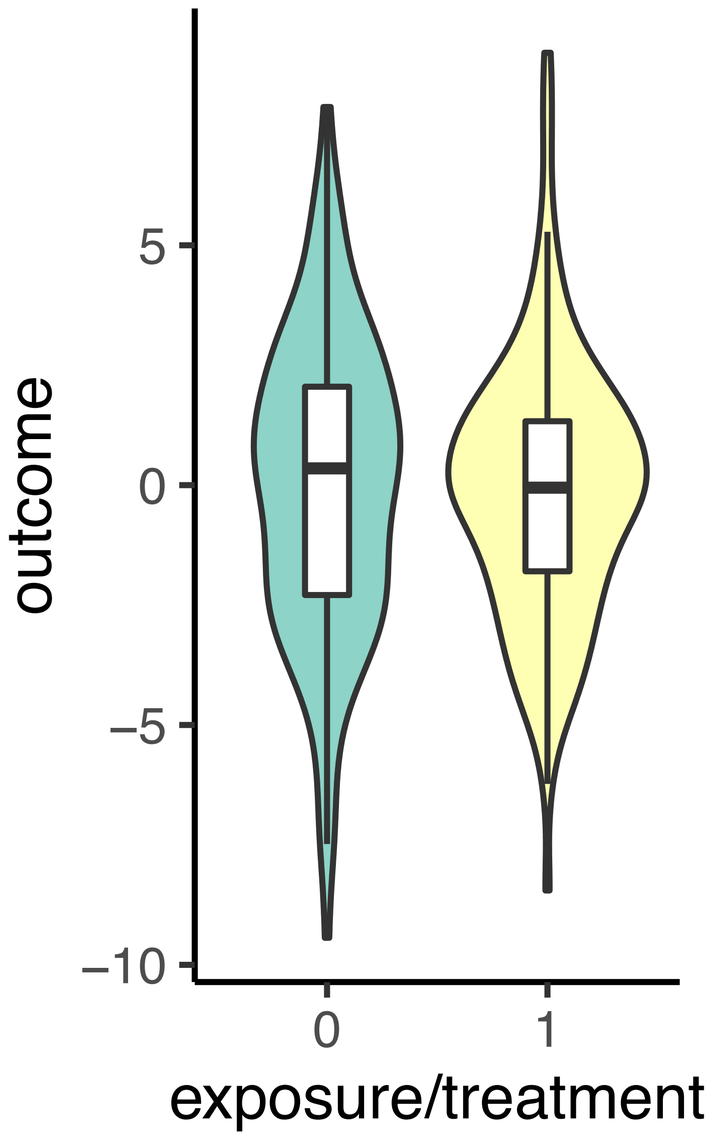


“Backdoor” exercise

Question: Which nodes should be “conditioned” and/or “adjusted” to block a reverse path from Y to X?



A working example: confounder adjustment in case-control study

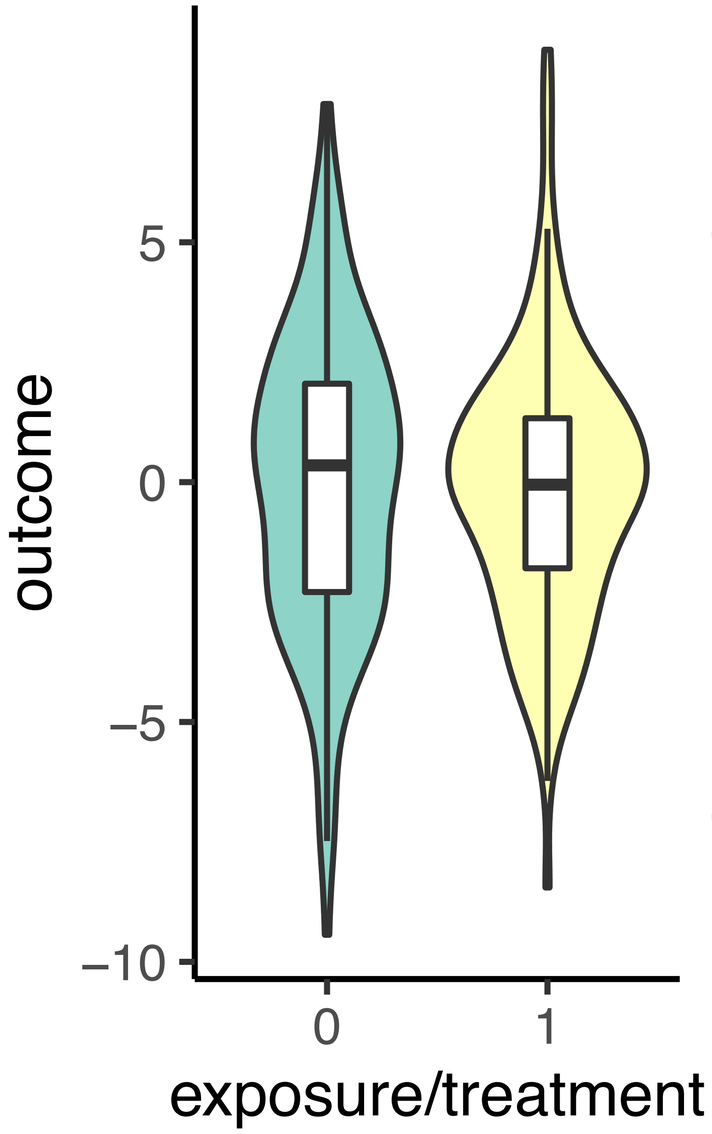


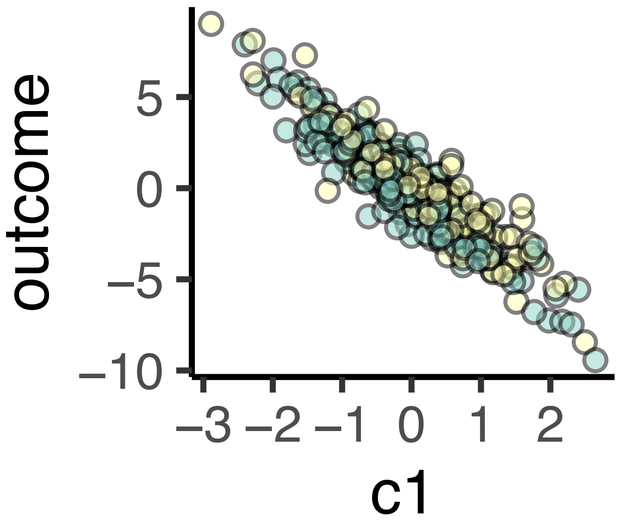
Key question: $X \overset{?}{\rightarrow} Y$

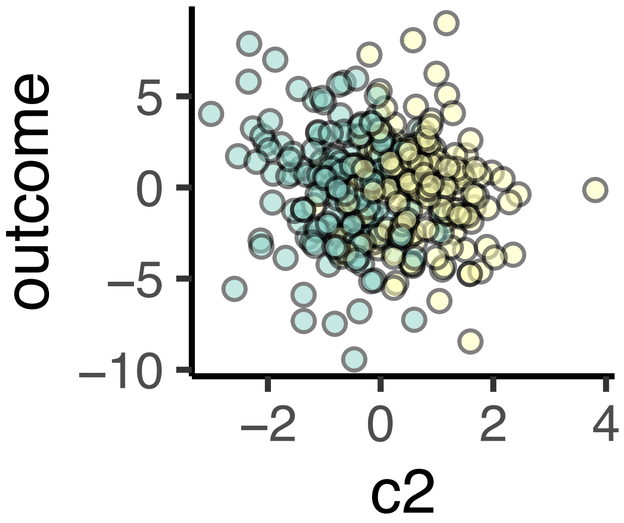
$\mathbf{X}$: exposure variable

$\mathbf{Y}$: outcome variable

A working example: confounder adjustment in case-control study







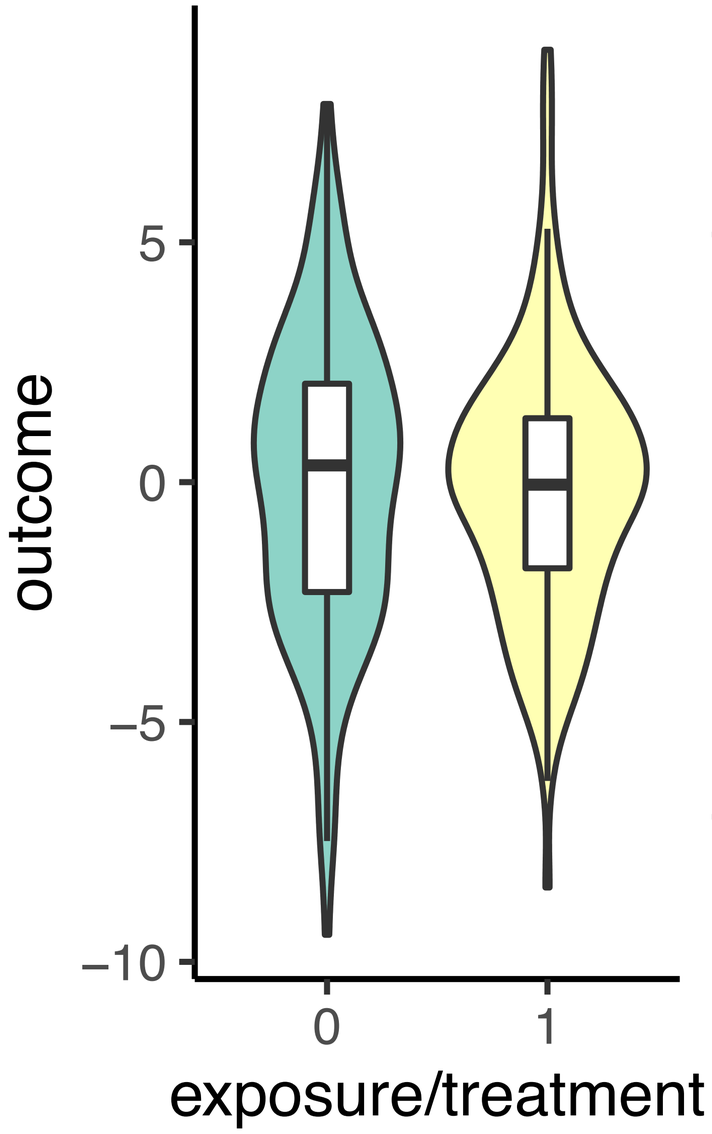
Key question: $X \overset{?}{\to} Y$

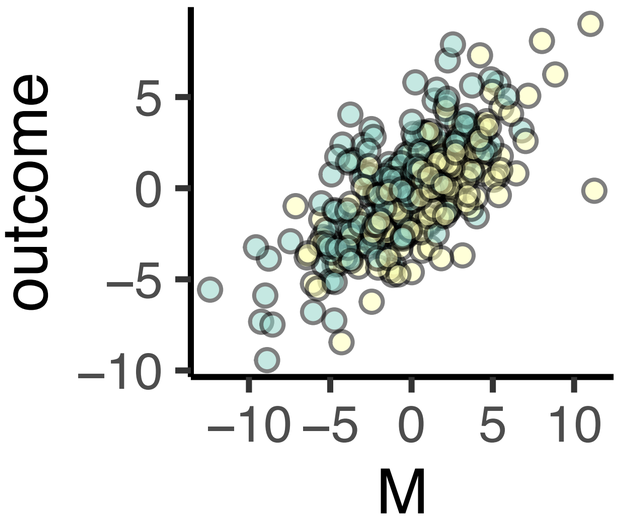
$\mathbf{X}$: exposure variable

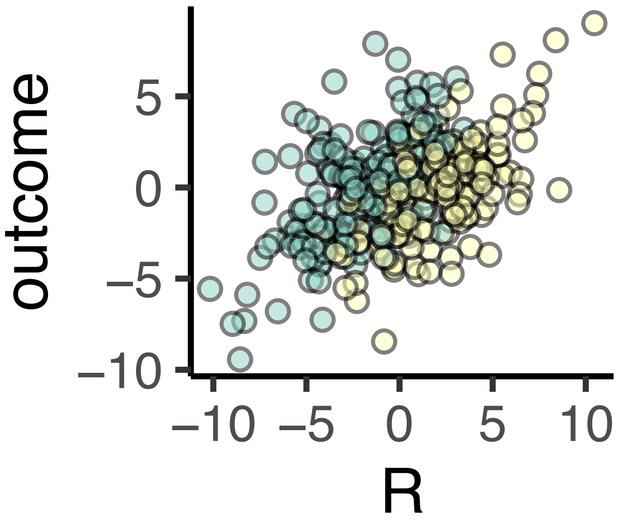
$\mathbf{Y}$: outcome variable

$\mathbf{C_1}$ and $\mathbf{C_2}$: covariates

A working example: confounder adjustment in case-control study







Key question: $X \overset{?}{\rightarrow} Y$

$\mathbf{X}$: exposure variable

$\mathbf{Y}$: outcome variable

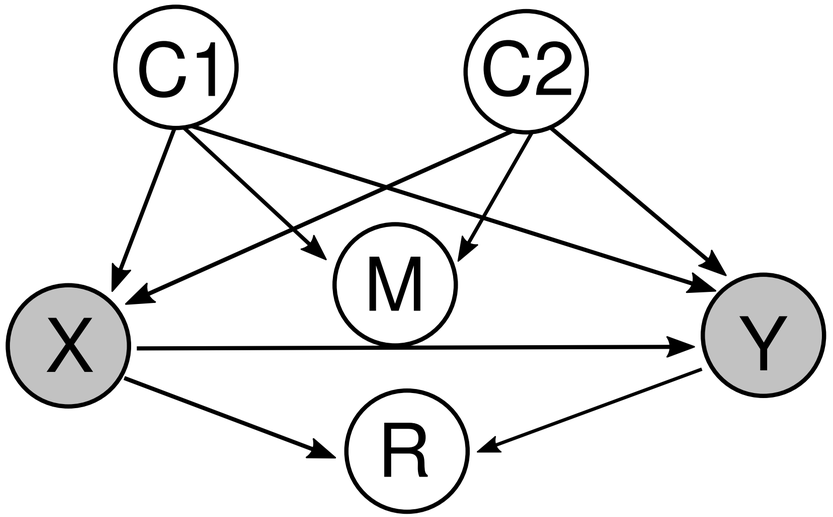
$\mathbf{C_1}$ and $\mathbf{C_2}$: covariates

$\mathbf{M}$: other covariate

$\mathbf{R}$: other covariate

Causal inference with a graphical model

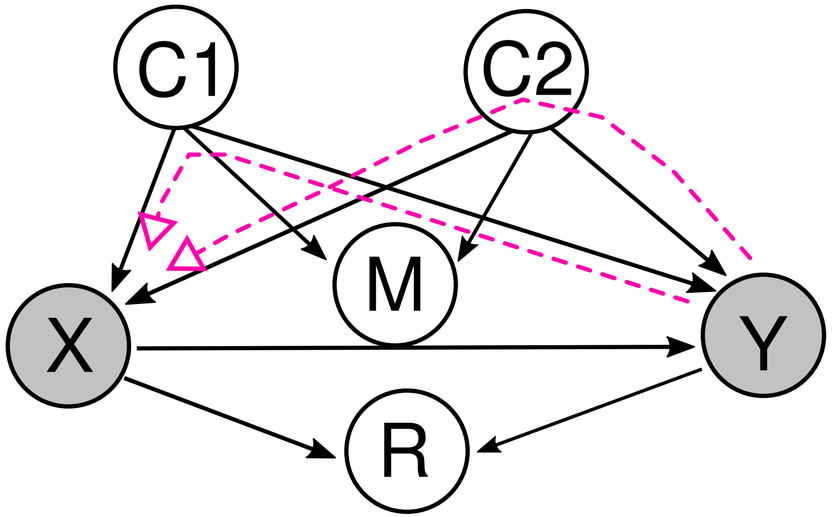
1. Build a causal structural model



What are potential backdoors?

Causal inference with a graphical model

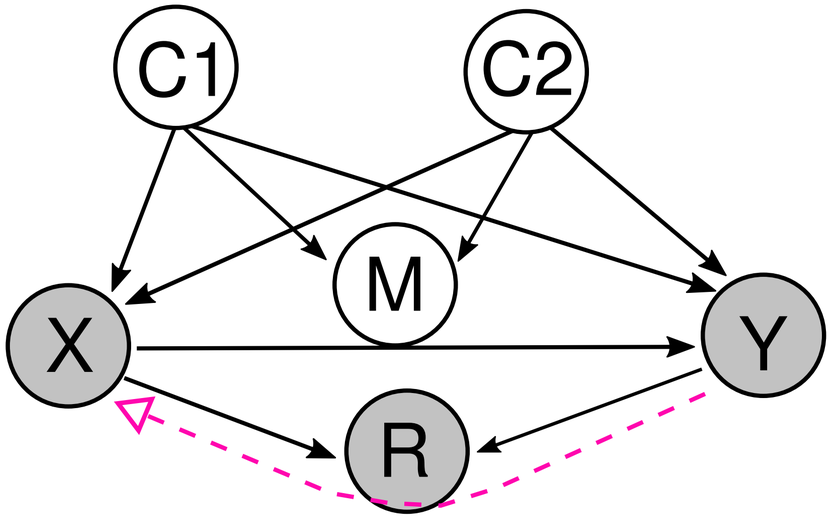
1. Build a causal structural model
2. Identify "back-door" paths/variables (*closing* $Y \to X$, *while opening* $X \to Y$)



How do we close them?

Causal inference with a graphical model

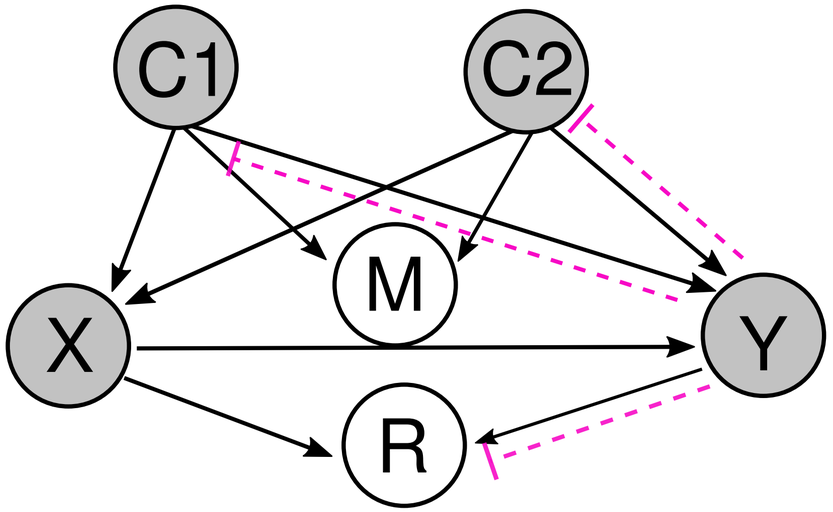
1. Build a causal structural model
2. Identify "back-door" paths/variables (*closing* $Y \to X$, *while opening* $X \to Y$)



What about this?

Causal inference with a graphical model

1. Build a causal structural model
2. Identify "back-door" paths/variables (*closing* $Y \to X$, *while opening* $X \to Y$)



Is this enough?

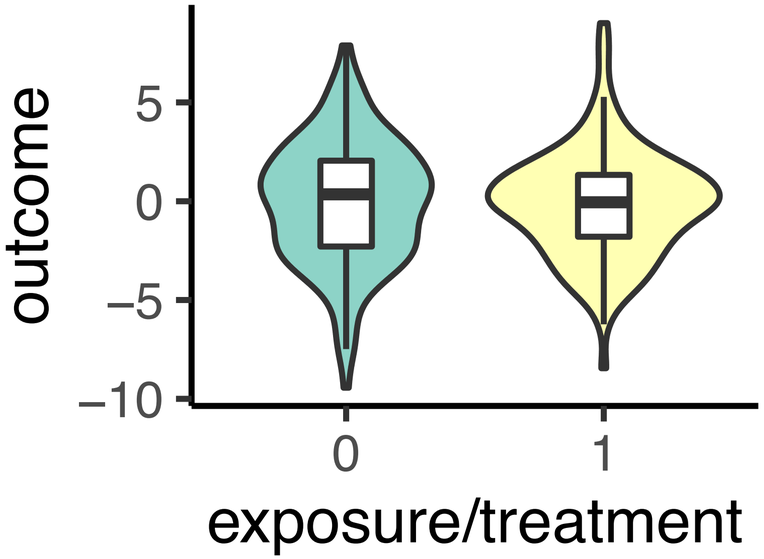
Causal inference with a graphical model

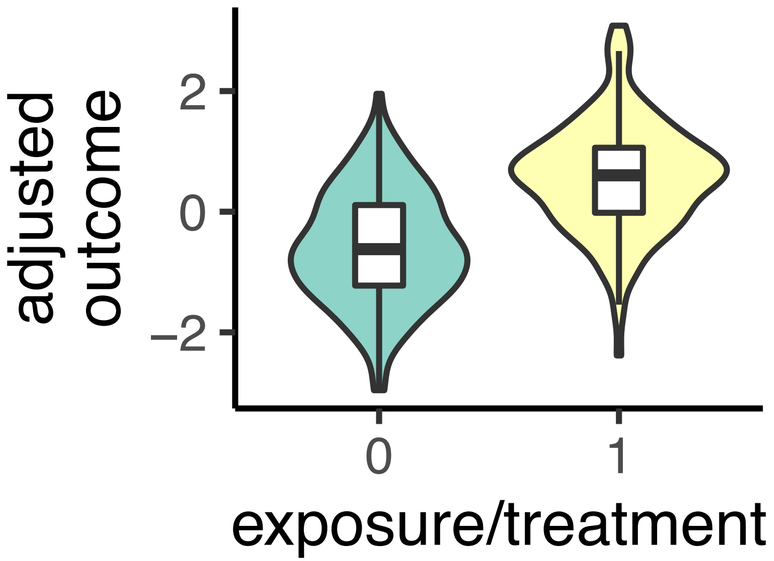
1. Build a causal structural model
2. Identify "back-door" paths/variables (*closing* $Y \rightarrow X$, *while opening* $X \rightarrow Y$)
3. Adjust "back-door" variables
4. Estimate causal effects

$$Y \leftarrow Y - \sum_{k=1}^{2} C_k \hat{\beta}_k$$

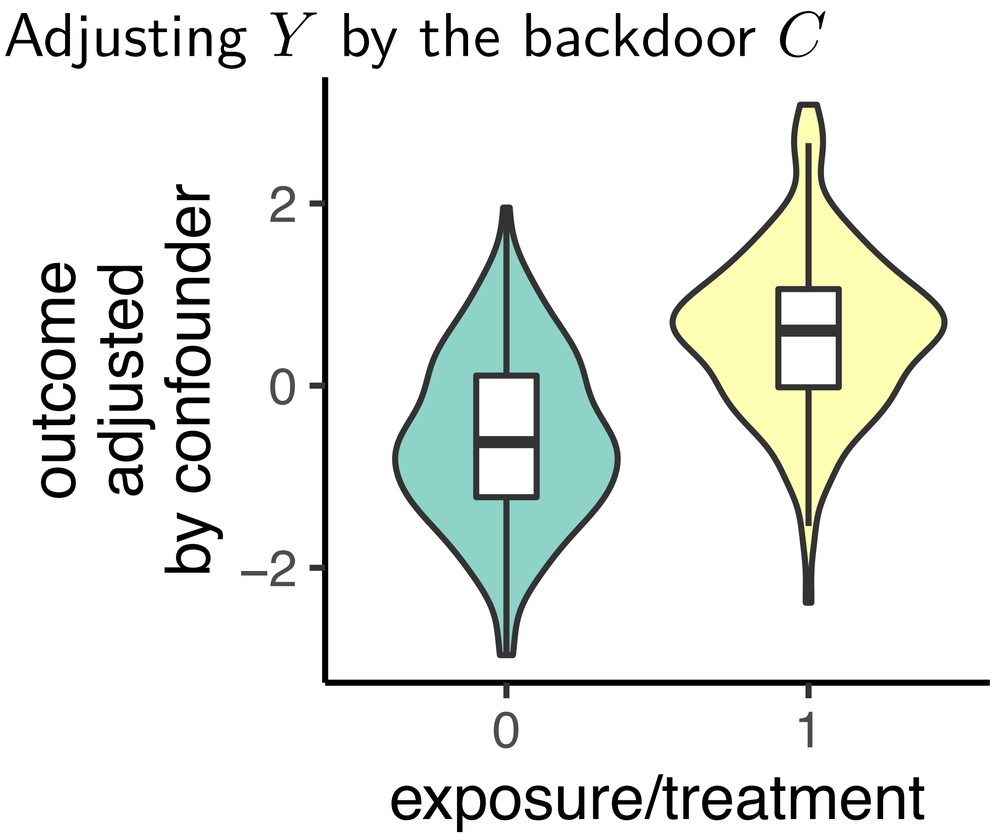
which approximates

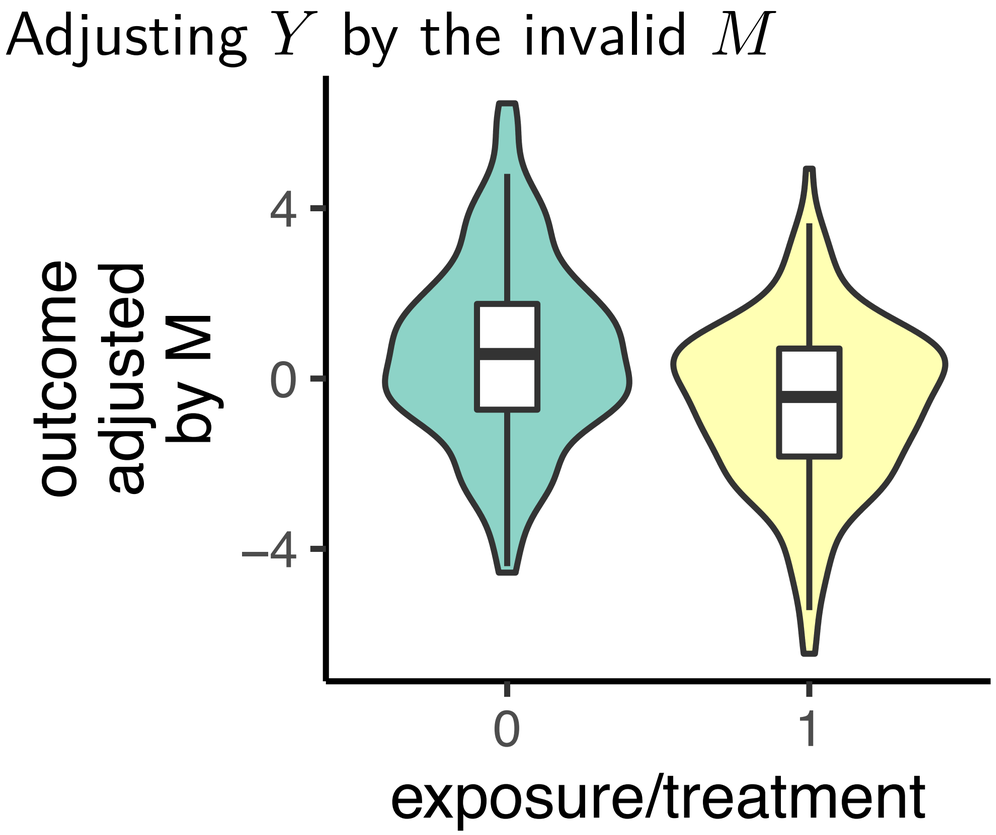
$$p(Y|X) = \int_C p(Y|X, C)p(C)dC$$



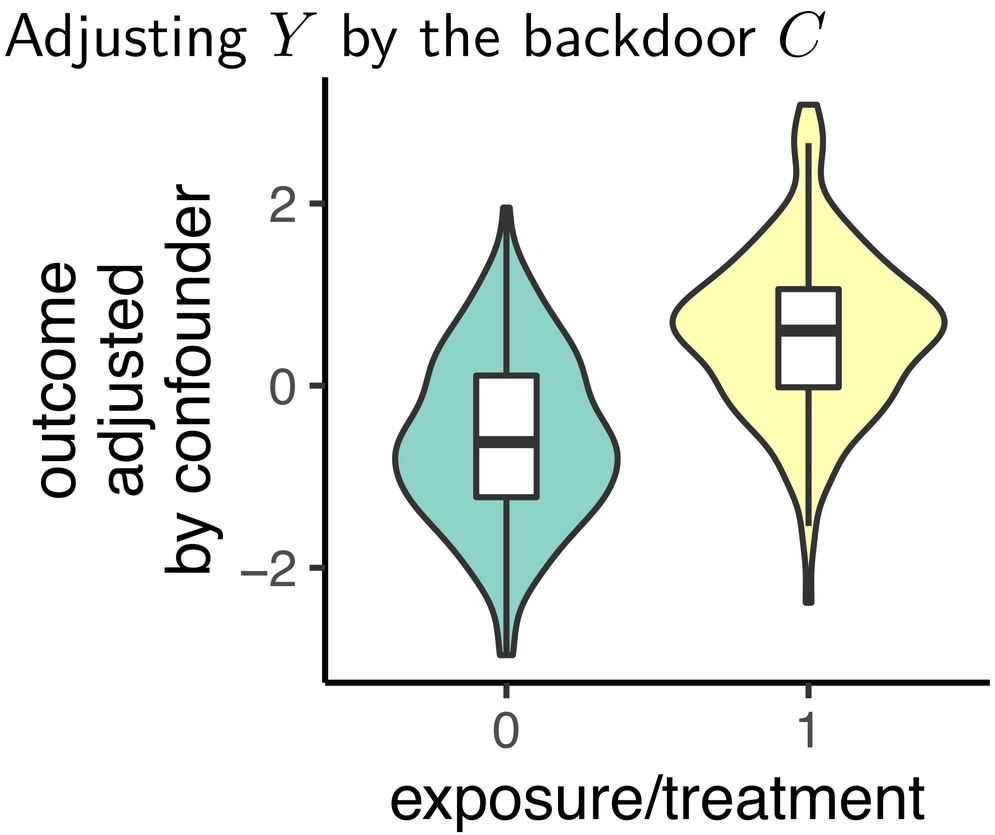


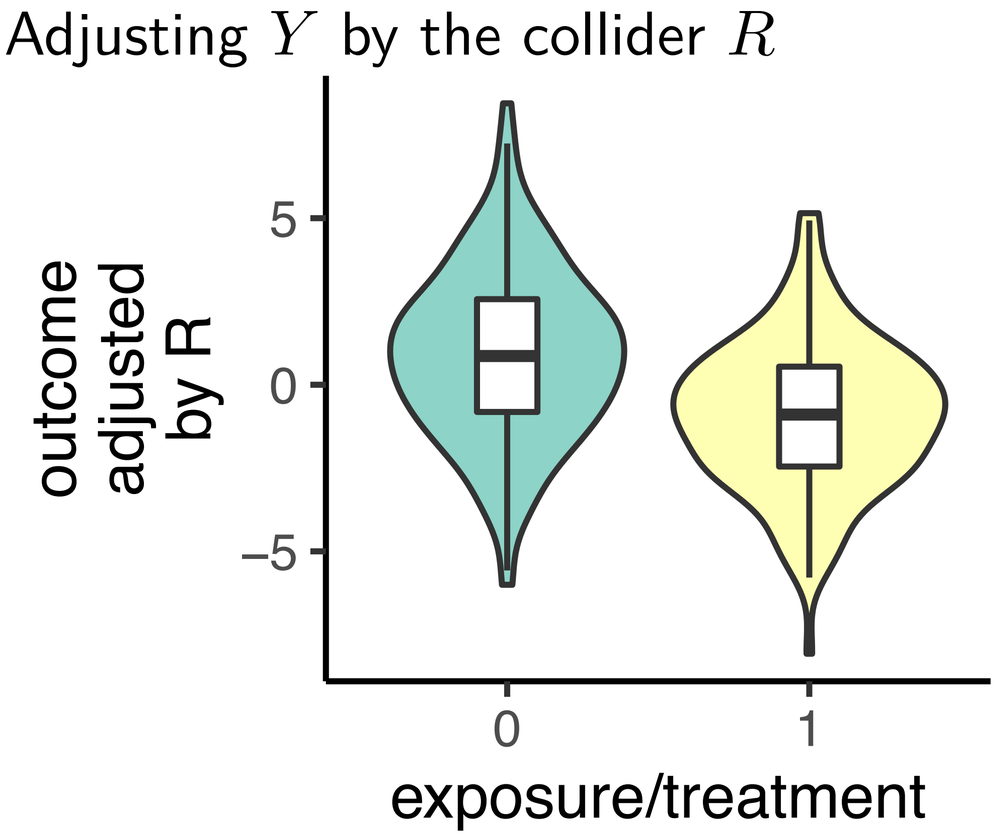
What would happen if we close a wrong "backdoor" variable?





What would happen if we close a wrong "backdoor" variable?

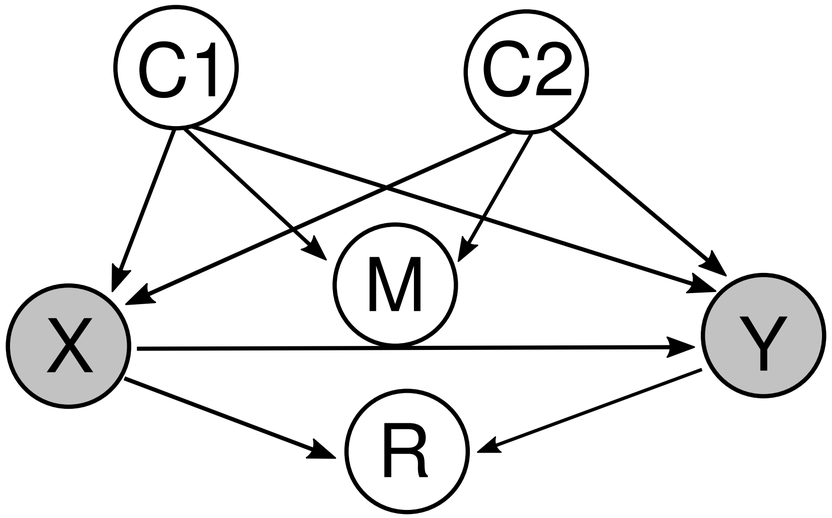




Today’s lecture: EDA & Exp Design

- **Exploratory Data Analysis**
  - First steps to data analysis
  - Traditional (low-dimensional) approaches
  - High-dimensional methods
  - Tips on how to organize your data/project
- **Experimental Design**
  - Observational vs. Experimental studies
  - Identification of unwanted variation
  - Causal inference: matching, stratification, inverse propensity

The same example with causal relationship from X to Y

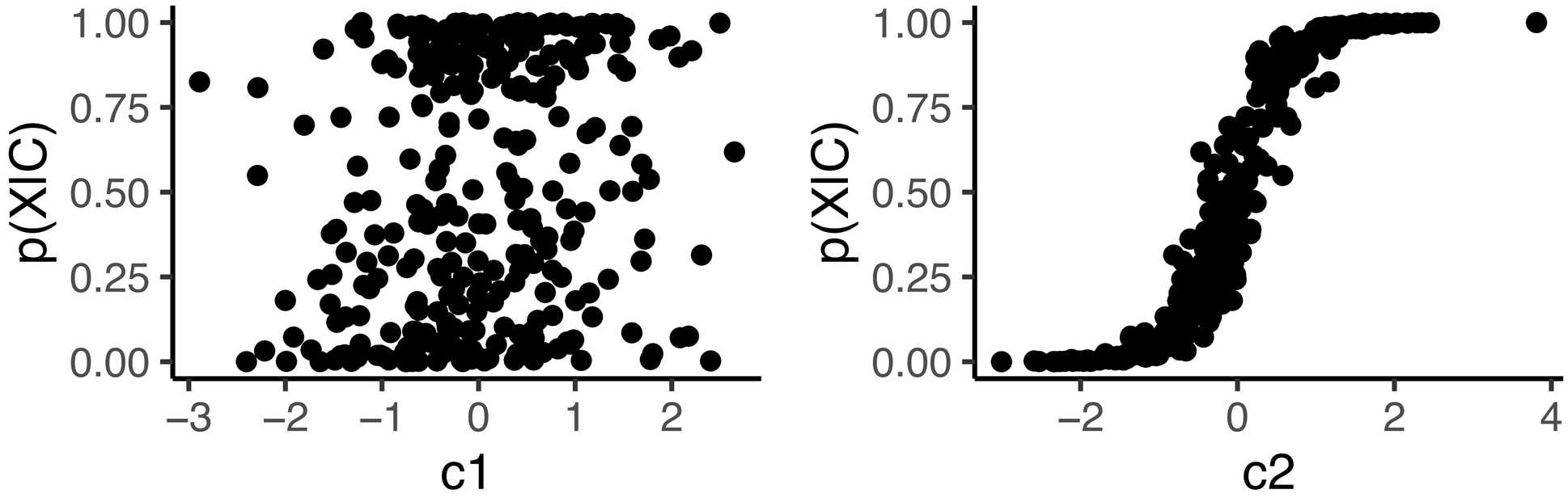


Inverse Propensity Weighting

What is the model for X given confoudners (the backdoor variables)?

Propensity $=$ probability of assignment $X = 1$:

$$p(X|C_1, C_2) \approx \frac{1}{1 + \exp(-\beta_0 - \beta_1 C_1 - \beta_2 C_2)}$$



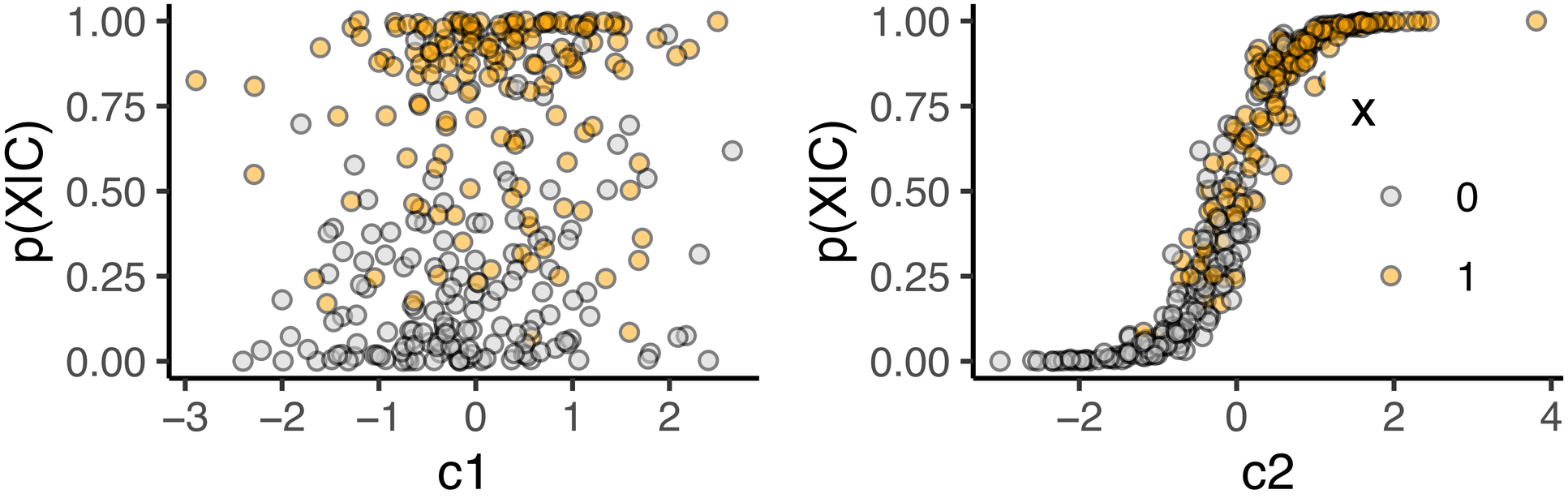
Horvitz and Thomson, *JASA*, 1952

Inverse Propensity Weighting

What is the model for X given confoudners (the backdoor variables)?

Propensity $=$ probability of assignment $X = 1$:

$$p(X|C_1, C_2) \approx \frac{1}{1 + \exp(-\beta_0 - \beta_1 C_1 - \beta_2 C_2)}$$



Horvitz and Thomson, *JASA*, 1952

Intuition behind IPW

What if we have assigned $X = 1$ unconfounded by C?

Intuition behind IPW

What if we have assigned $X = 1$ unconfounded by C?

In other words, what if samples could be drawn more from the underrepresented group?

Intuition behind IPW

What if we have assigned $X = 1$ unconfounded by C?

In other words, what if samples could be drawn more from the underrepresented group?

Likewise, what if samples could be dropped in the overrepresented group?

Inverse Propensity Weighting "inverse" confounded assignment

$$\hat{Y}_i^{(1)} = \frac{X_i Y_i}{\hat{p}(X_i = 1|C_i)}$$

$$\hat{Y}_i^{(0)} = \frac{(1 - X_i) Y_i}{1 - \hat{p}(X_i = 1|C_i)}$$

```
p.xc <-
    glm(x~cc, family="binomial") %>%
    predict() %>%
    sigmoid() %>%
    clamp() # avoid 0 or 1

ww <- x / p.xc + (1-x) / (1-p.xc)
```

Inverse Propensity Weighting "inverse" confounded assignment

$$\hat{Y}_i^{(1)} = \frac{X_i Y_i}{\hat{p}(X_i = 1|C_i)}$$

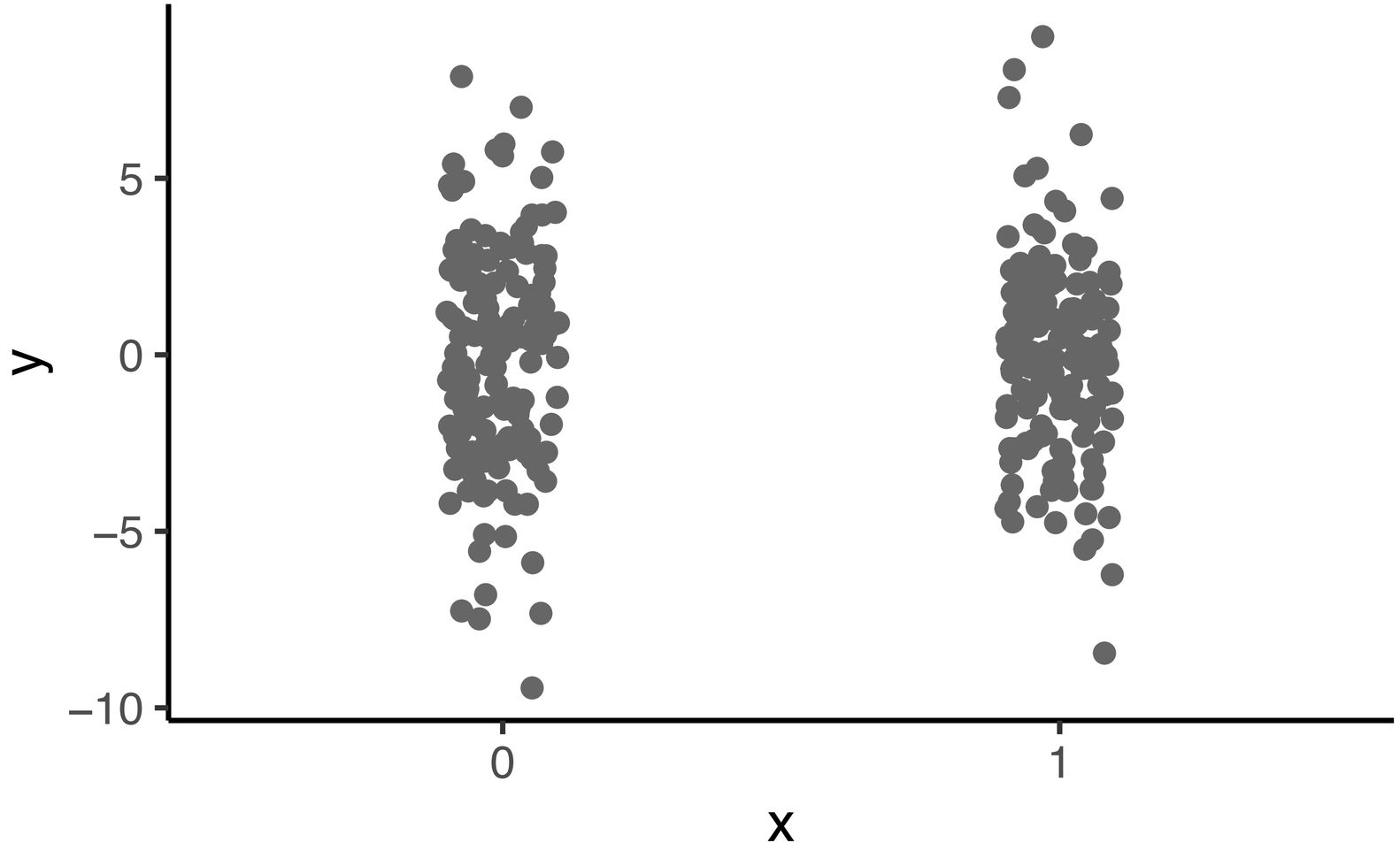
$$\hat{Y}_i^{(0)} = \frac{(1 - X_i) Y_i}{1 - \hat{p}(X_i = 1|C_i)}$$

equivalently give weights for $\forall$ i

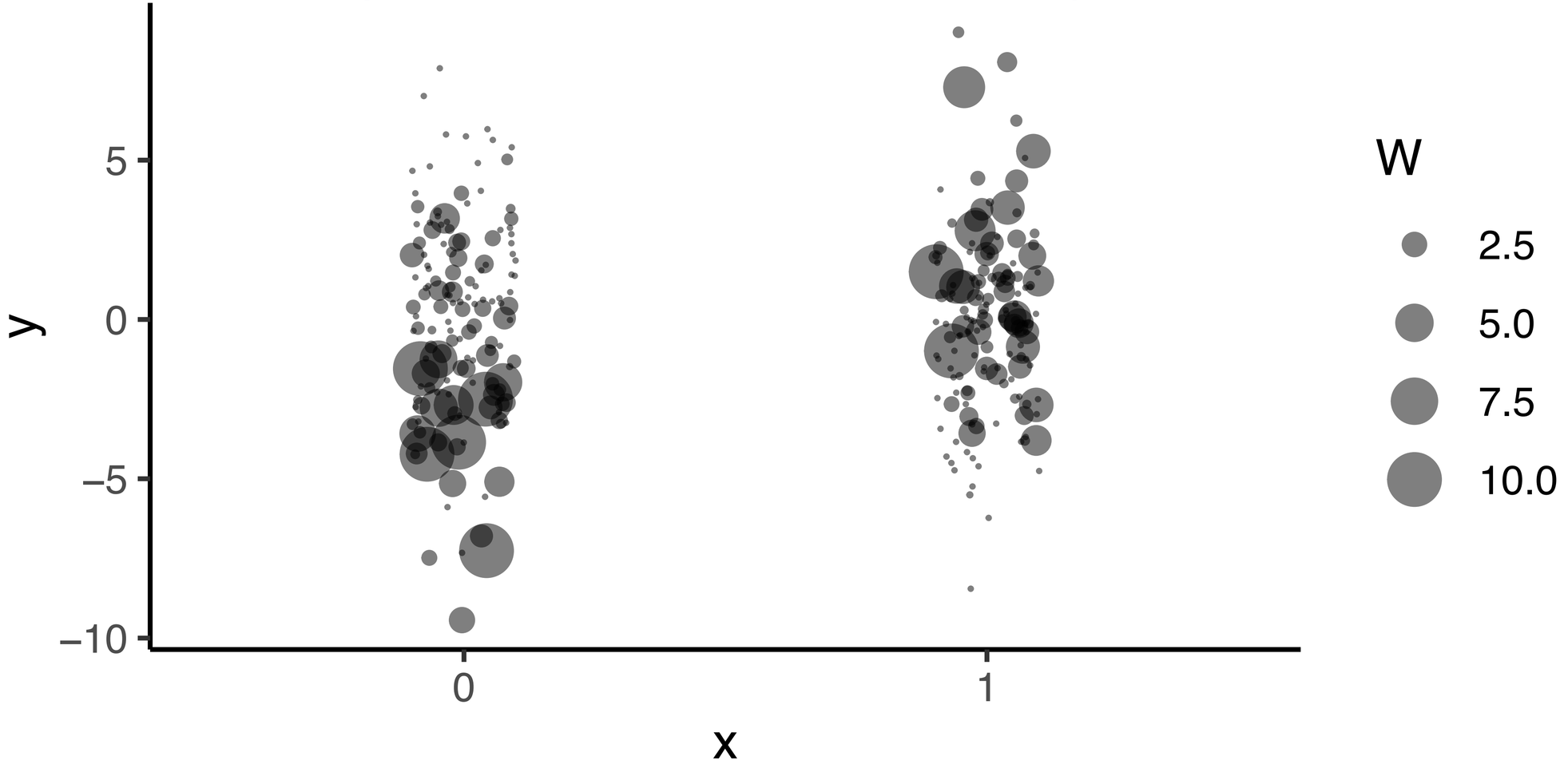
$$W_i \propto \begin{cases} 1/p(X_i = 1|C_i) & X_i = 1 \\ 1/p(X_i = 0|C_i) & X_i = 0 \end{cases}$$

```
p.xc <-
    glm(x~cc, family="binomial") %>%
    predict() %>%
    sigmoid() %>%
    clamp() # avoid 0 or 1

ww <- x / p.xc + (1-x) / (1-p.xc)
```

Take samples inversely proportional to propensity

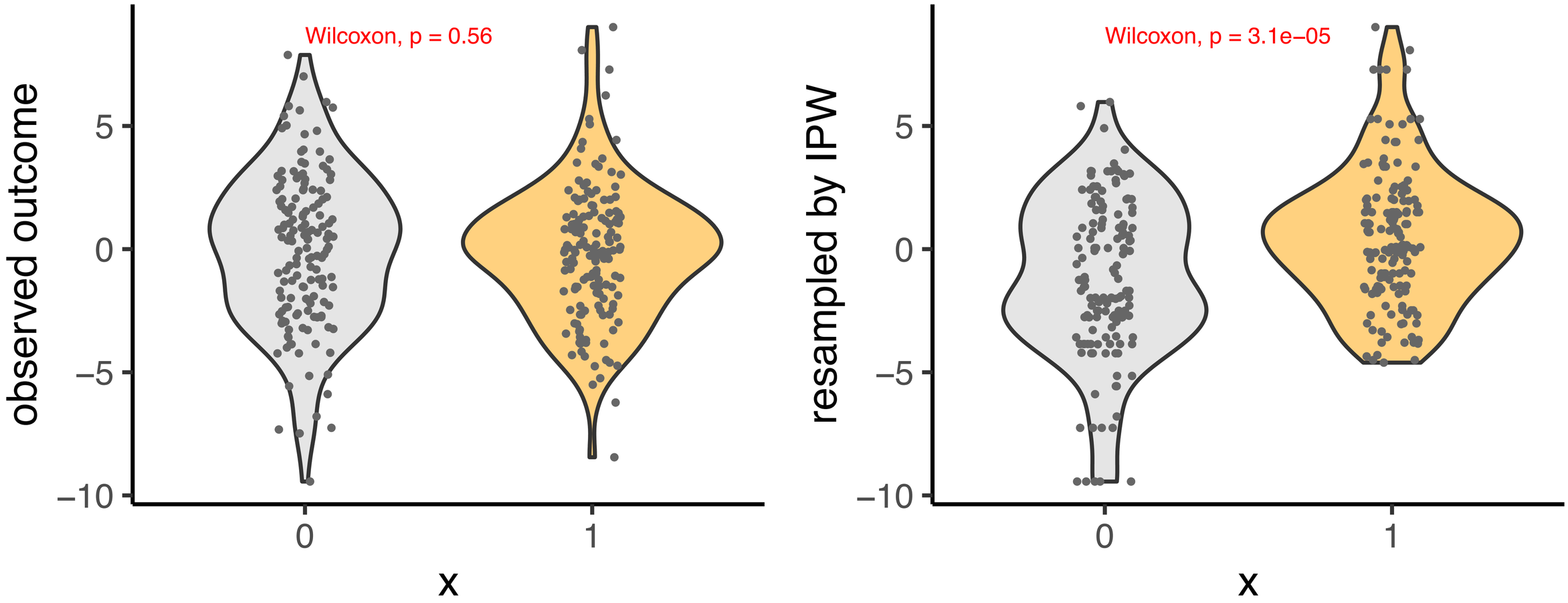


Take samples inversely proportional to propensity



$$W_i \propto \begin{cases} 1/p(X_i = 1|C_i) & X_i = 1 \\ 1/p(X_i = 0|C_i) & X_i = 0 \end{cases}$$

Take samples inversely proportional to propensity

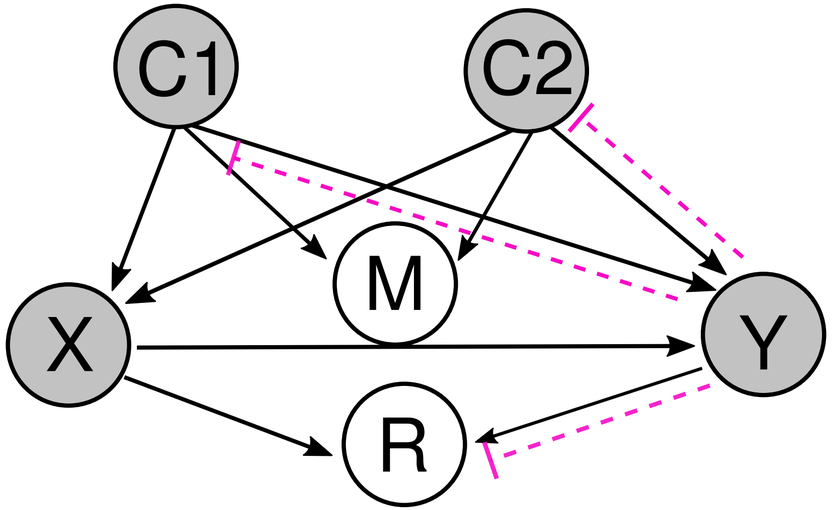


$$W_i \propto \begin{cases} 1/p(X_i = 1|C_i) & X_i = 1 \\ 1/p(X_i = 0|C_i) & X_i = 0 \end{cases}$$

Today's lecture: EDA & Exp Design

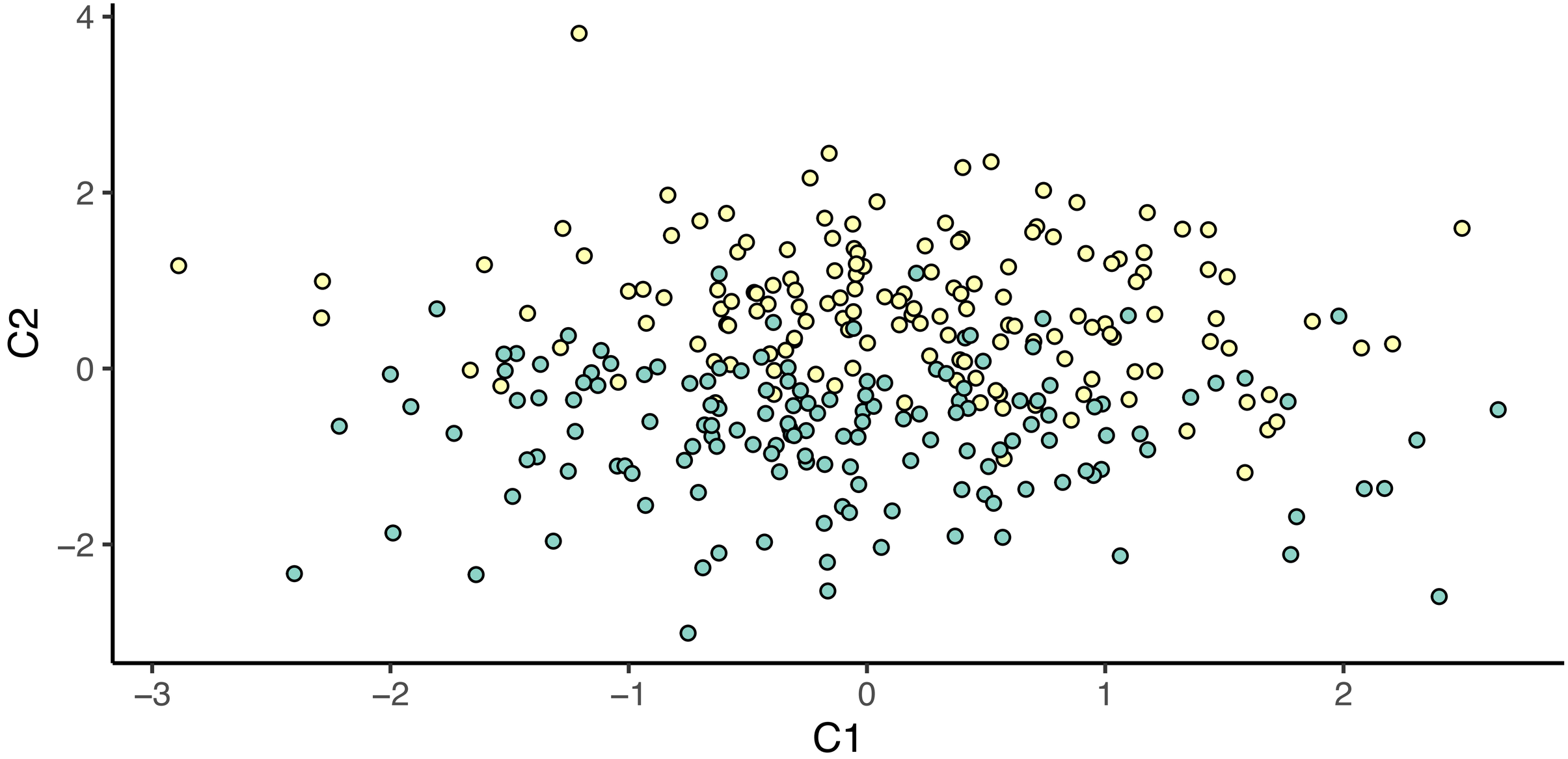
- **Exploratory Data Analysis**
  - First steps to data analysis
  - Traditional (low-dimensional) approaches
  - High-dimensional methods
  - Tips on how to organize your data/project
- **Experimental Design**
  - Observational vs. Experimental studies
  - Identification of unwanted variation
  - Causal inference: matching, stratification, inverse propensity
- **Tips on effective data visualization in science**

Estimating potential outcome by matching

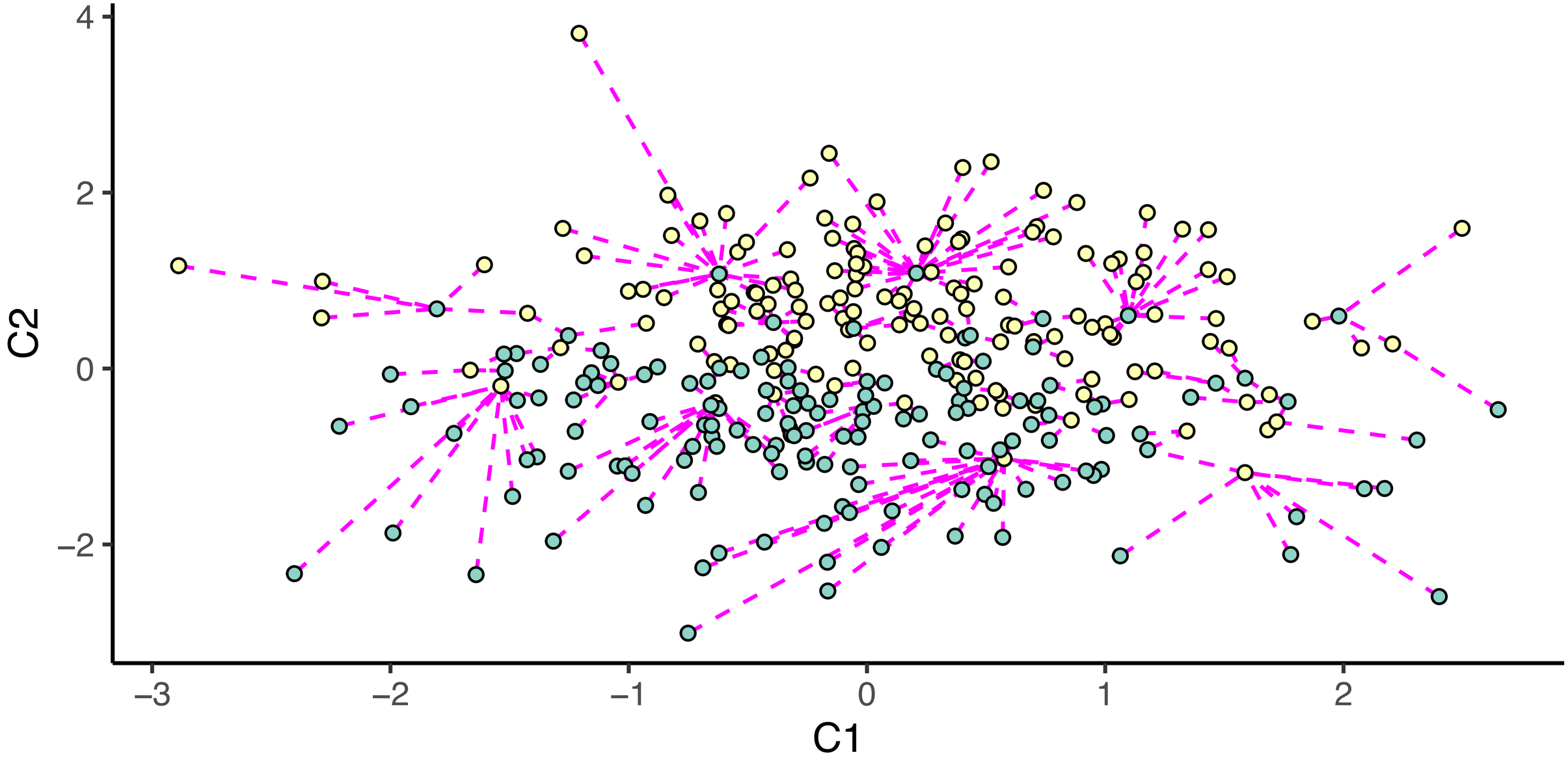


- Estimtae $\mathbb{E}\left[Y_i^{(0)}|C_{i1}, C_{i2}\right]$ for $X_i = 1$ to compare with $\mathbb{E}[Y_i|X_i = 1]$
- Estimtae $\mathbb{E}\left[Y_i^{(1)}|C_{i1}, C_{i2}\right]$ for $X_i = 0$ to compare with $\mathbb{E}[Y_i|X_i = 0]$

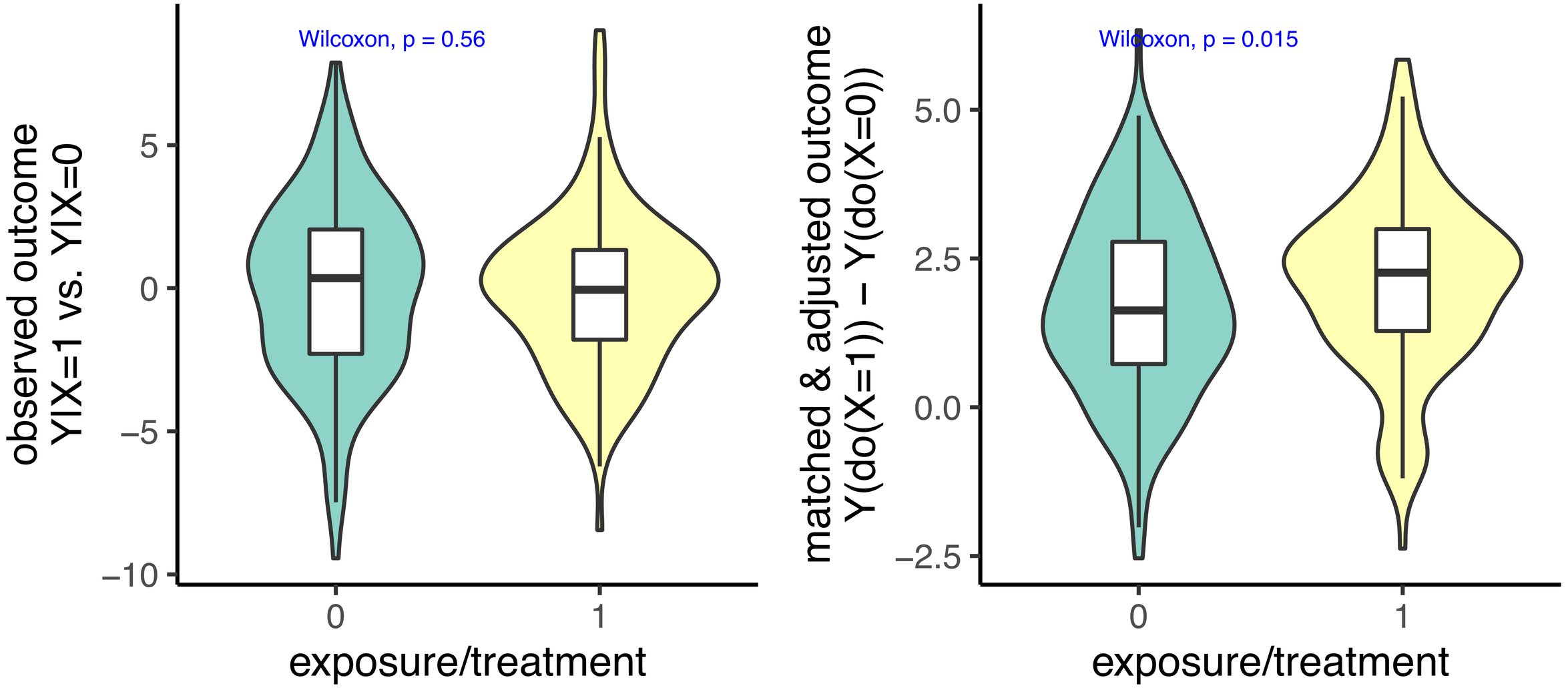
Estimating potential outcome by matching



Estimating potential outcome by matching



Estimating potential outcome by matching



Bayesian Additive Regression Tree (BART) approach

- G-formula $\rightarrow$ outcome regression models
- Regression model for $Y \sim S$ for each $X = 1$ and $X = 0$ using BART

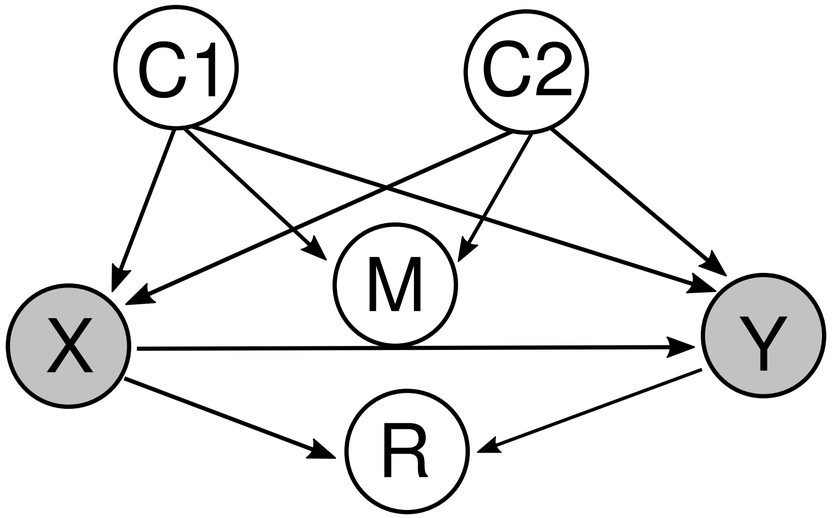
$$\mathbb{E}\left[Y^{(1)}\right] = \int_S \mathbb{E}[Y|X = 1, S]dS$$

$$\mathbb{E}\left[Y^{(0)}\right] = \int_S \mathbb{E}[Y|X = 0, S]dS$$

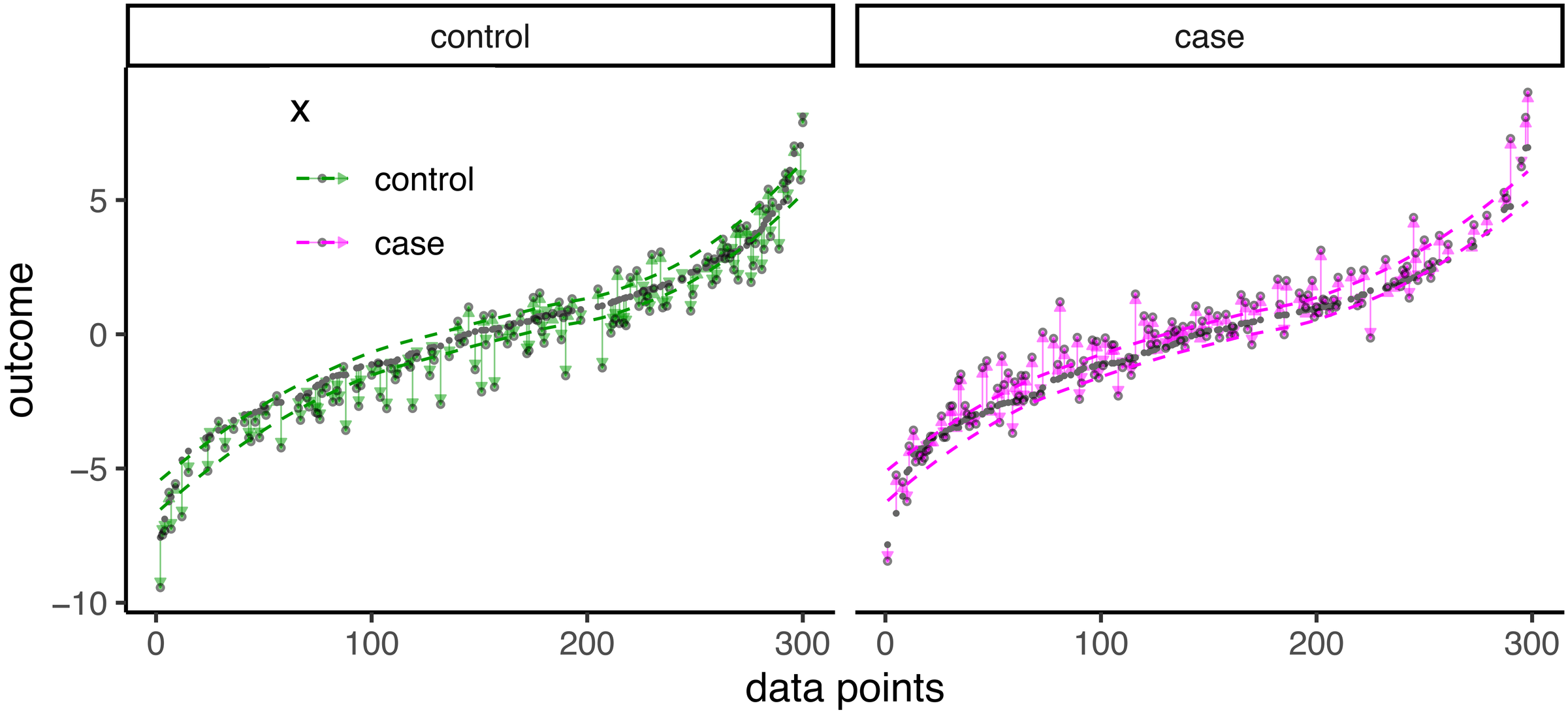
- Estimate causal effect: $\mathbb{E}\left[Y^{(1)}\right] - \mathbb{E}\left[Y^{(0)}\right]$

Hill, *Bayesian Nonparametric Modeling for Causal Inference* (2011)

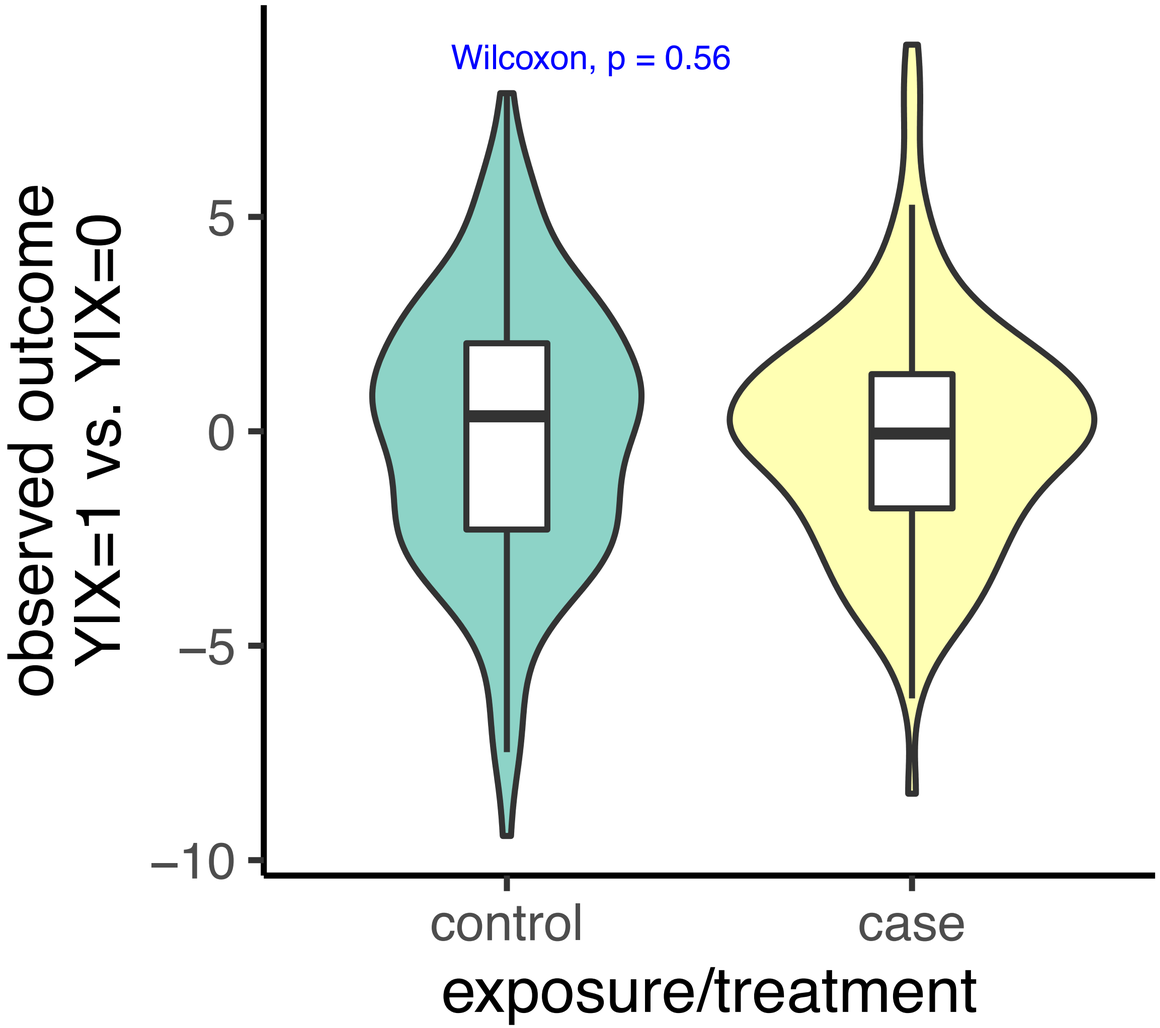
The same example with causal relationship from X to Y

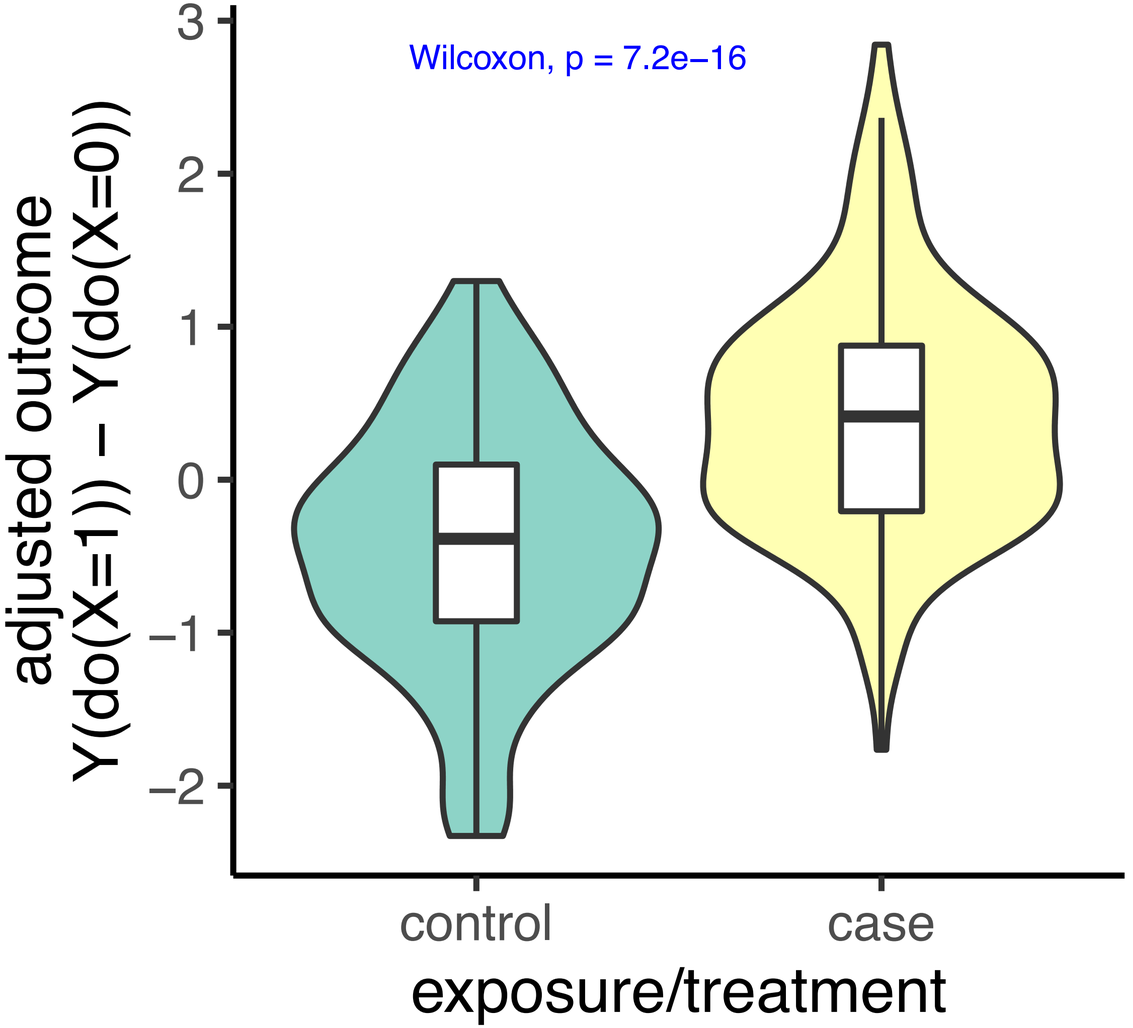


BART: a regression model to impute potential outcomes



BART: a regression model to impute potential outcomes





Today’s lecture: EDA & Exp Design

- **Exploratory Data Analysis**
  - First steps to data analysis
  - Traditional (low-dimensional) approaches
  - High-dimensional methods
  - Tips on how to organize your data/project
- **Experimental Design**
  - Observational vs. Experimental studies
  - Identification of unwanted variation (SVA)
  - Causal inference: matching, stratification, inverse propensity