HBC
Harvard Chan Bioinformatics Core

Pseudobulk and related approaches for scRNA-seq analysis

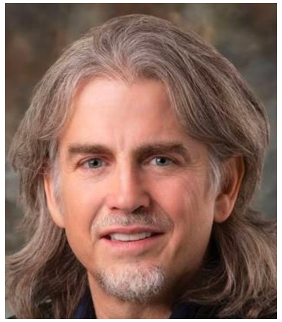
https://tinyurl.com/hbc-pseudobulk-quarto

Harvard Chan Bioinformatics Core

Introductions!

Shannan Ho Sui
Director

Lorena Pantano
Director of Bioinformatics Platform



John Quackenbush
Faculty Advisor

Upen Bhattarai

Will Gammerdinger

Noor Sohail

Alex Bartlett

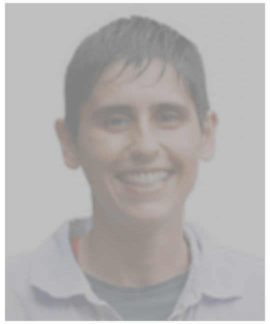
Elizabeth Partan

Emma Berdan

Zhu Zhuo

James Billingsley

Shannan Ho Sui
Director



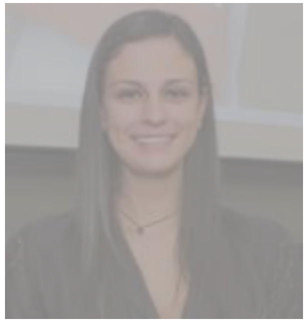
Lorena Pantano
Director of Bioinformatics Platform

John Quackenbush
Faculty Advisor

Upen Bhattarai

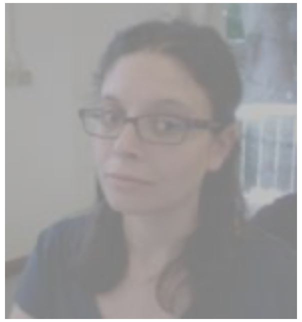
Will Gammerdinger

Noor Sohail



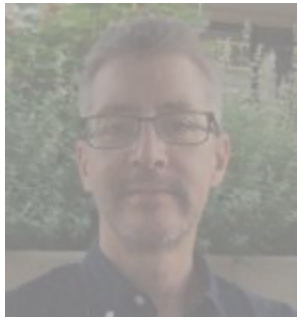
Alex Bartlett

Elizabeth Partan



Emma Berdan

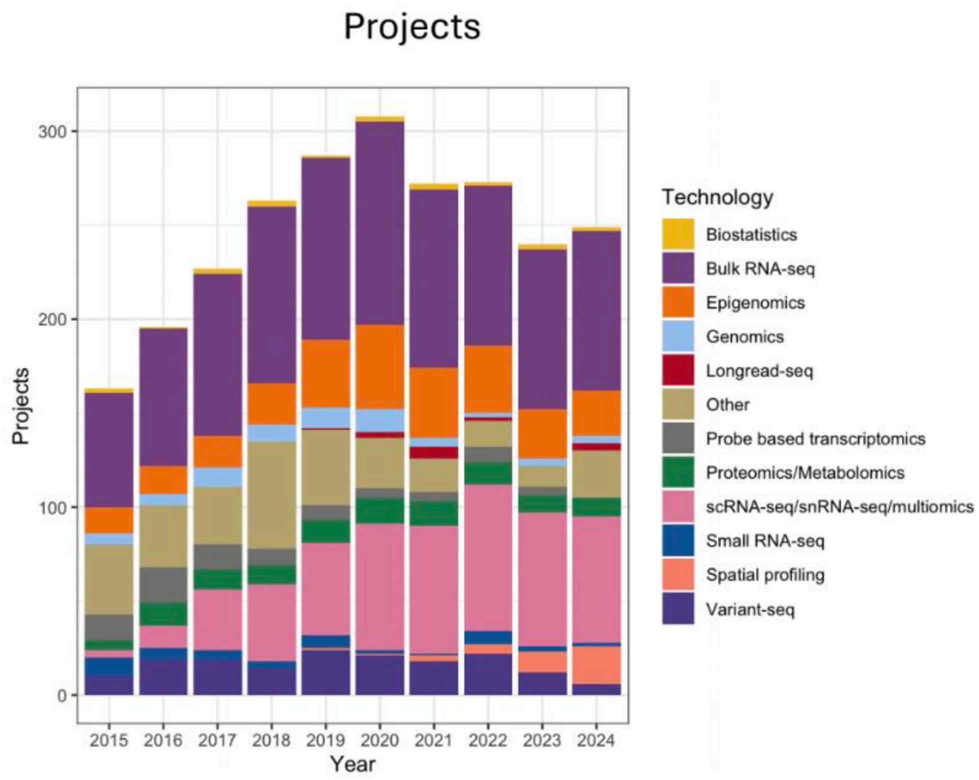
Zhu Zhuo



James Billingsley

Consulting

- Transcriptomics: Bulk, single cell, small RNA
- Epigenomics: ChIP-seq, CUT&RUN, ATAC-seq, DNA methylation
- Variant discovery: WGS, resequencing, exome-seq and CNV
- Multiomics integration
- Spatial biology
- Experimental design and grant support



https://hsph.harvard.edu/research/bioinformatics/consulting/

Consulting

- Transcriptomics: Bulk, single cell, small RNA
- Epigenomics: ChIP-seq, CUT&RUN, ATAC-seq, DNA methylation
- Variant discovery: WGS, resequencing, exome-seq and CNV
- Multiomics integration
- Spatial biology
- Experimental design and grant support

HARVARD
T.H. CHAN
SCHOOL OF PUBLIC HEALTH

NIEHS

HARVARD
CATALYST

THE HARVARD CLINICAL
AND TRANSLATIONAL
SCIENCE CENTER

HARVARD
MEDICAL SCHOOL

Training

- Hands-on workshops design to reflect best practices, reproducibility and an emphasis on experimental design
  - Basic Data Skills
    - Shell
    - R
  - Advanced Topics: Analysis of high-throughput sequencing data
    - Chromatin Biology
    - Bulk RNA-seq
    - Differential Gene Expression
    - scRNA-seq
    - Variant Calling
  - Current Topics in Bioinformatics

https://hsph.harvard.edu/research/bioinformatics/training/

Training

- Hands-on workshops design to reflect best practices, reproducibility and an emphasis on experimental design
  - Basic Data Skills
    - Shell
    - R
  - Advanced Topics: Analysis of high-throughput sequencing data
    - Chromatin Biology
    - Bulk RNA-seq
    - Differential Gene Expression
    - scRNA-seq
    - Variant Calling
  - Current Topics in Bioinformatics

HARVARD T.H. CHAN SCHOOL OF PUBLIC HEALTH

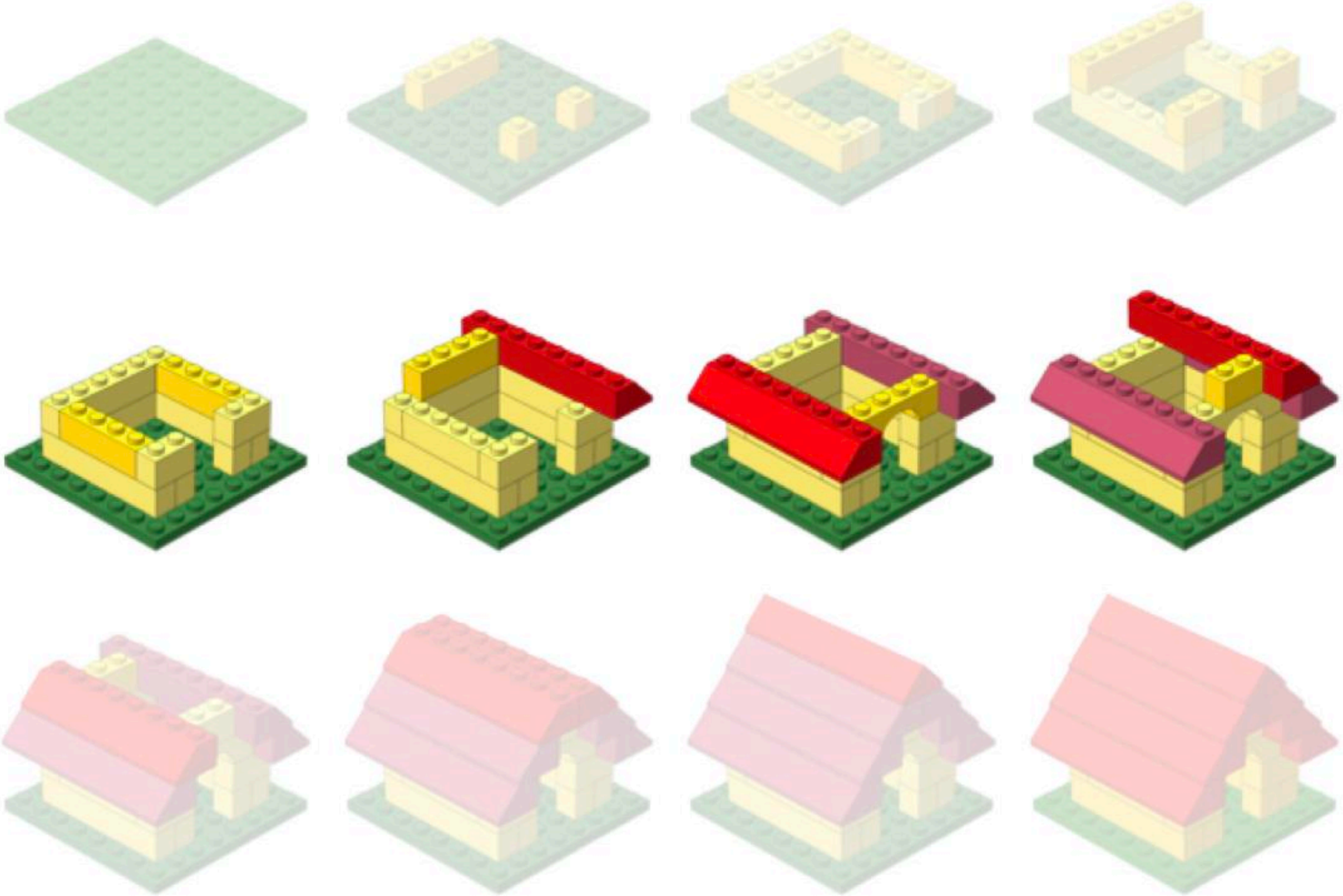
DF/HCC DANA-FARBER / HARVARD CANCER CENTER

HARVARD CATALYST THE HARVARD CLINICAL AND TRANSLATIONAL SCIENCE CENTER

HARVARD MEDICAL SCHOOL

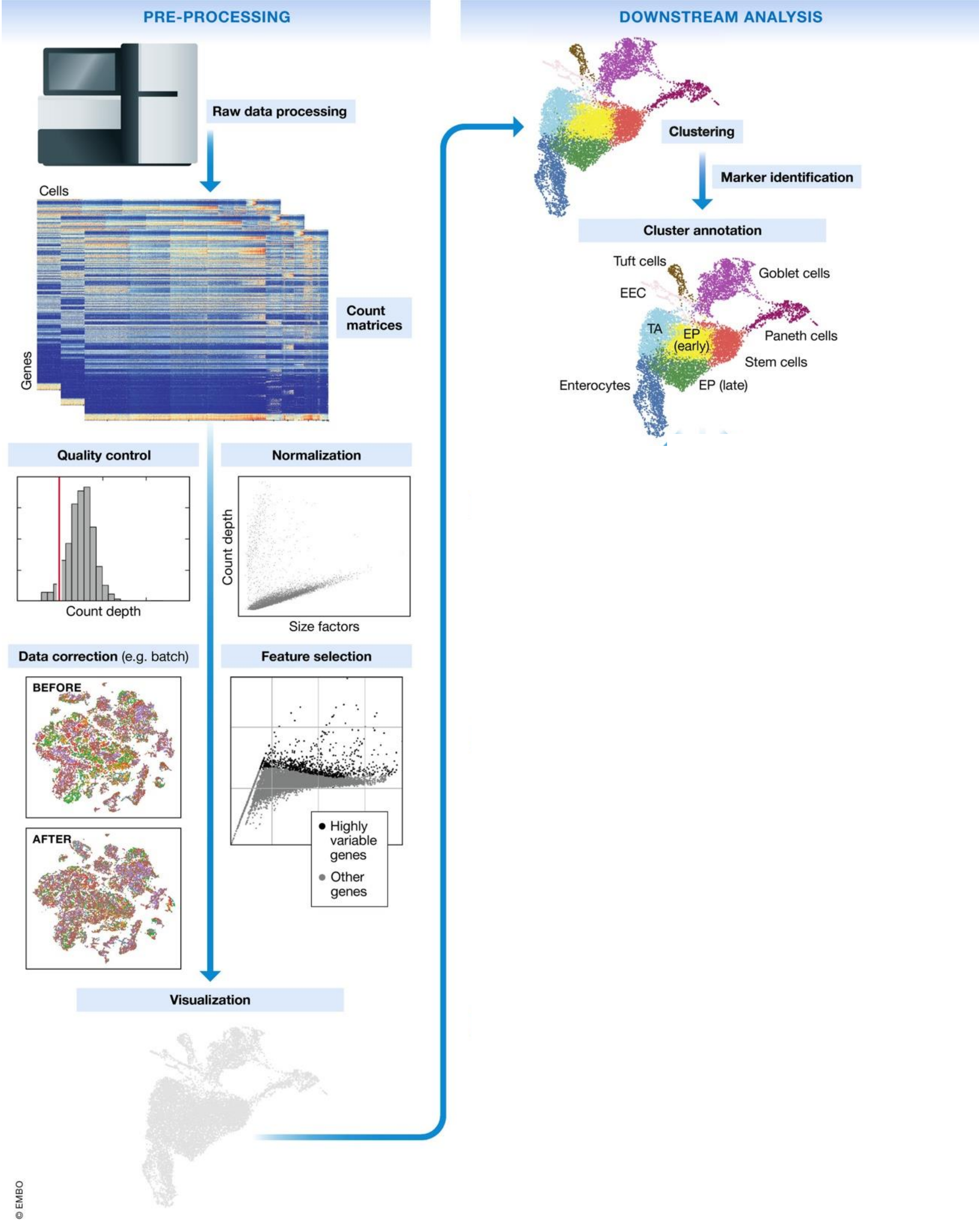
https://hsph.harvard.edu/research/bioinformatics/training/

Workshop scope



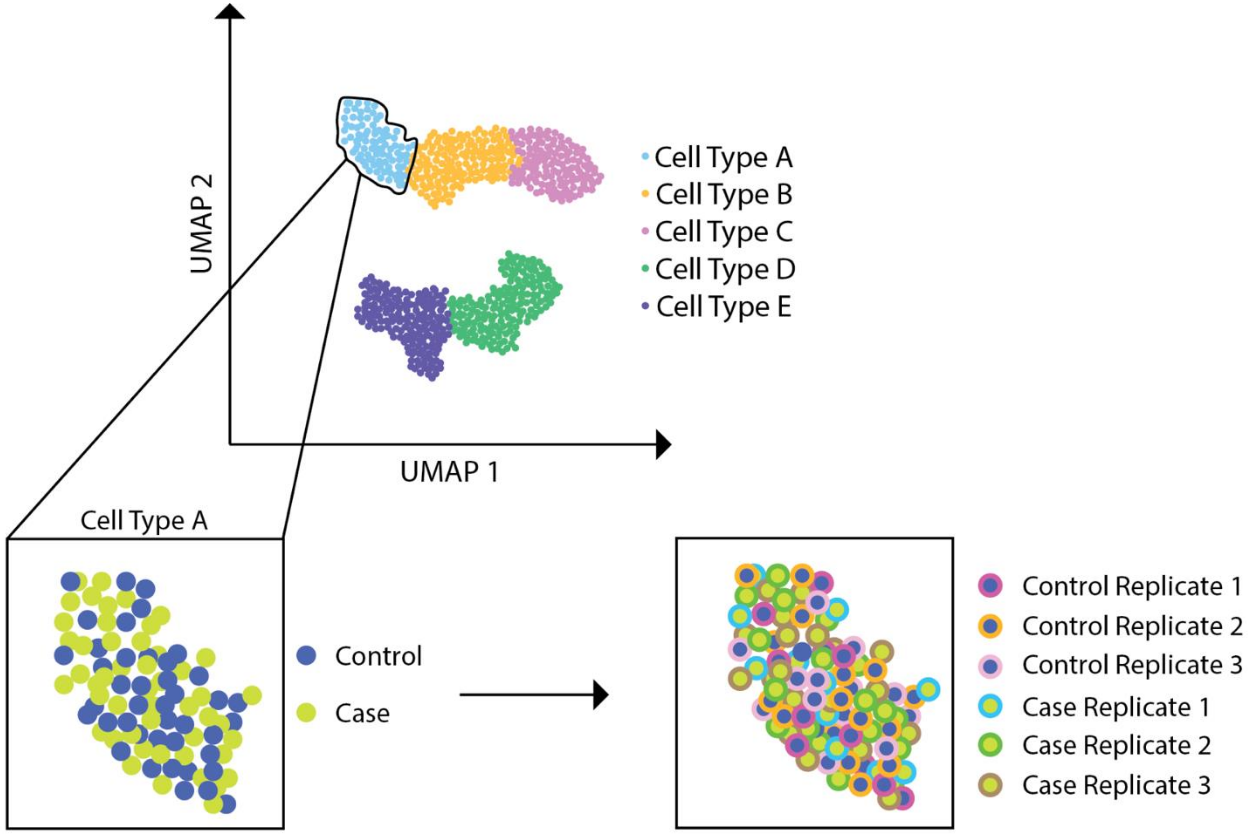
Bioinformatic Data Analysis

What is NOT covered in this workshop

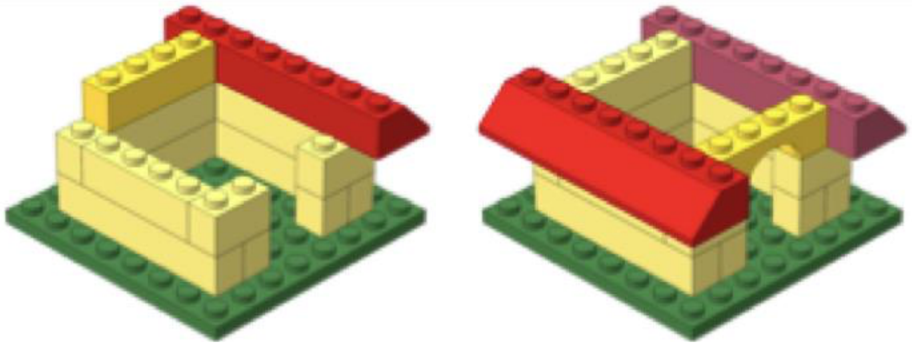


Luecken, MD and Theis, FJ. Current best practices in single-cell RNA-seq analysis: a tutorial, Mol Syst Biol 2019 (doi: https://doi.org/10.15252/msb.20188746)

What IS covered in this workshop



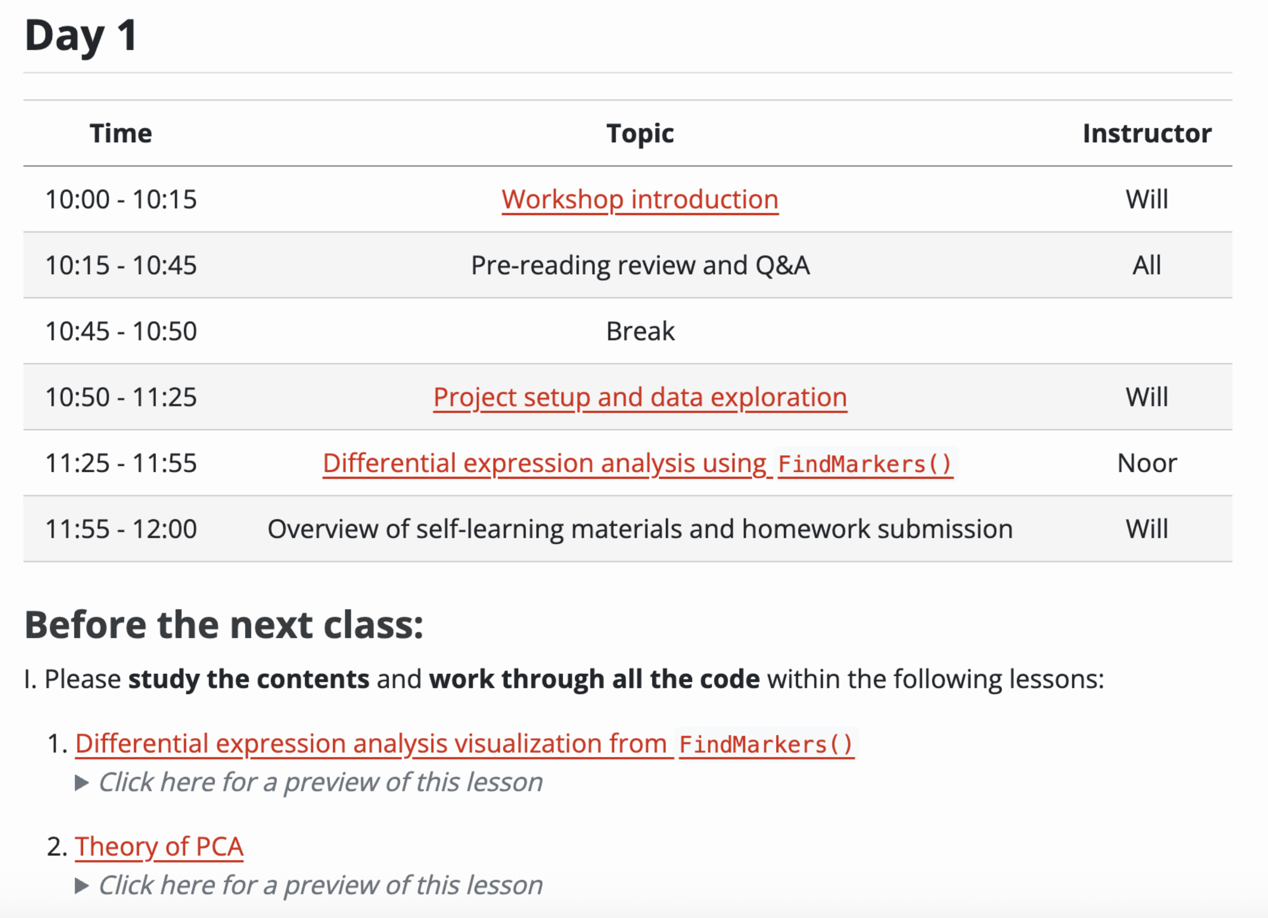
Workshop Scope



- Understanding considerations for when to use different DGE algorithms on scRNA-seq data
- Using FindMarkers to evaluate significantly DE genes
- Aggregating single cell expression data into a pseudobulk counts matrix to run a DESeq2 workflow
- Evaluating expression patterns of differentially expressed genes at the pseudobulk and single cell level
- Application of methods for evaluating differential proportions of cells between conditions

Logistics

Course schedule



Day 1

Time	Topic	Instructor
10:00 - 10:15	Workshop introduction	Will
10:15 - 10:45	Pre-reading review and Q&A	All
10:45 - 10:50	Break	
10:50 - 11:25	Project setup and data exploration	Will
11:25 - 11:55	Differential expression analysis using `FindMarkers()`	Noor
11:55 - 12:00	Overview of self-learning materials and homework submission	Will

Before the next class:

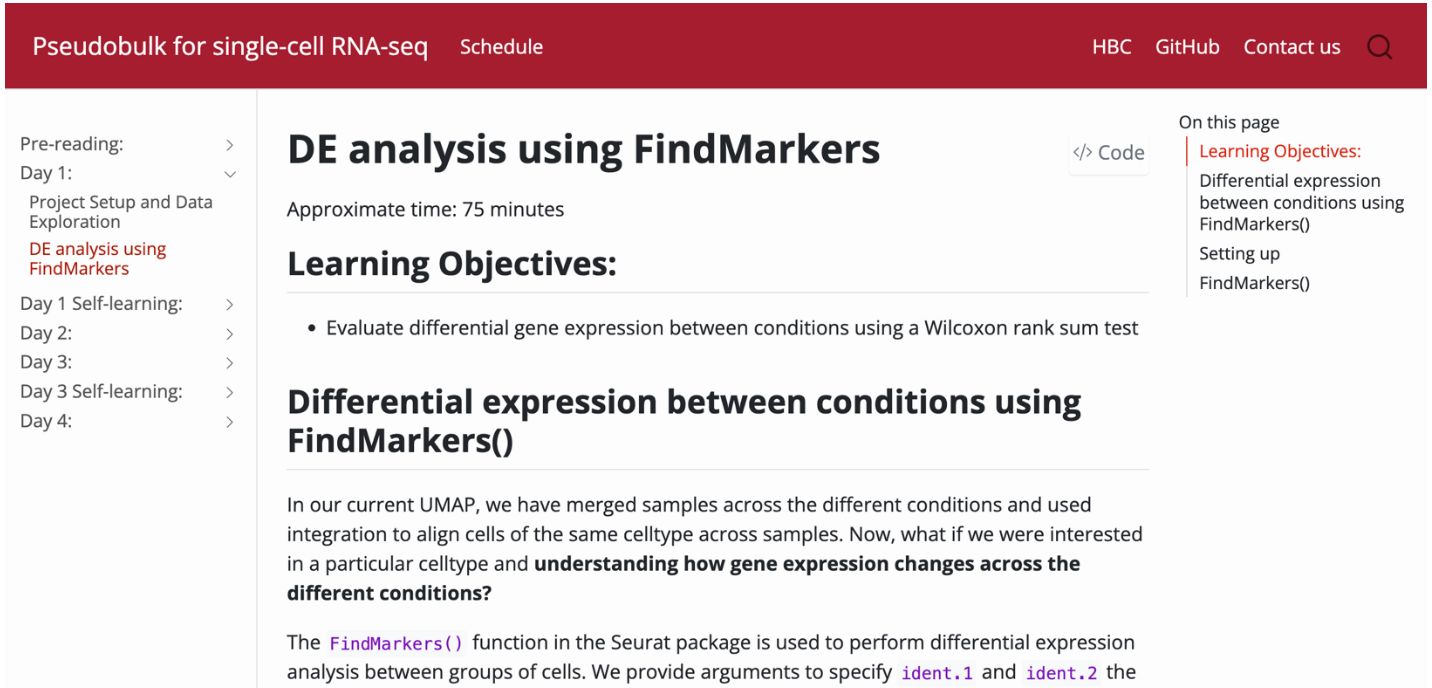
I. Please **study the contents** and **work through all the code** within the following lessons:

1. Differential expression analysis visualization from `FindMarkers()`
   ▶ *Click here for a preview of this lesson*
2. Theory of PCA
   ▶ *Click here for a preview of this lesson*

https://tinyurl.com/hbc-pseudobulk-quarto

Course materials

- We continuously update our materials to reflect changes in the field/software



DE analysis using FindMarkers

Approximate time: 75 minutes

Learning Objectives:

- Evaluate differential gene expression between conditions using a Wilcoxon rank sum test

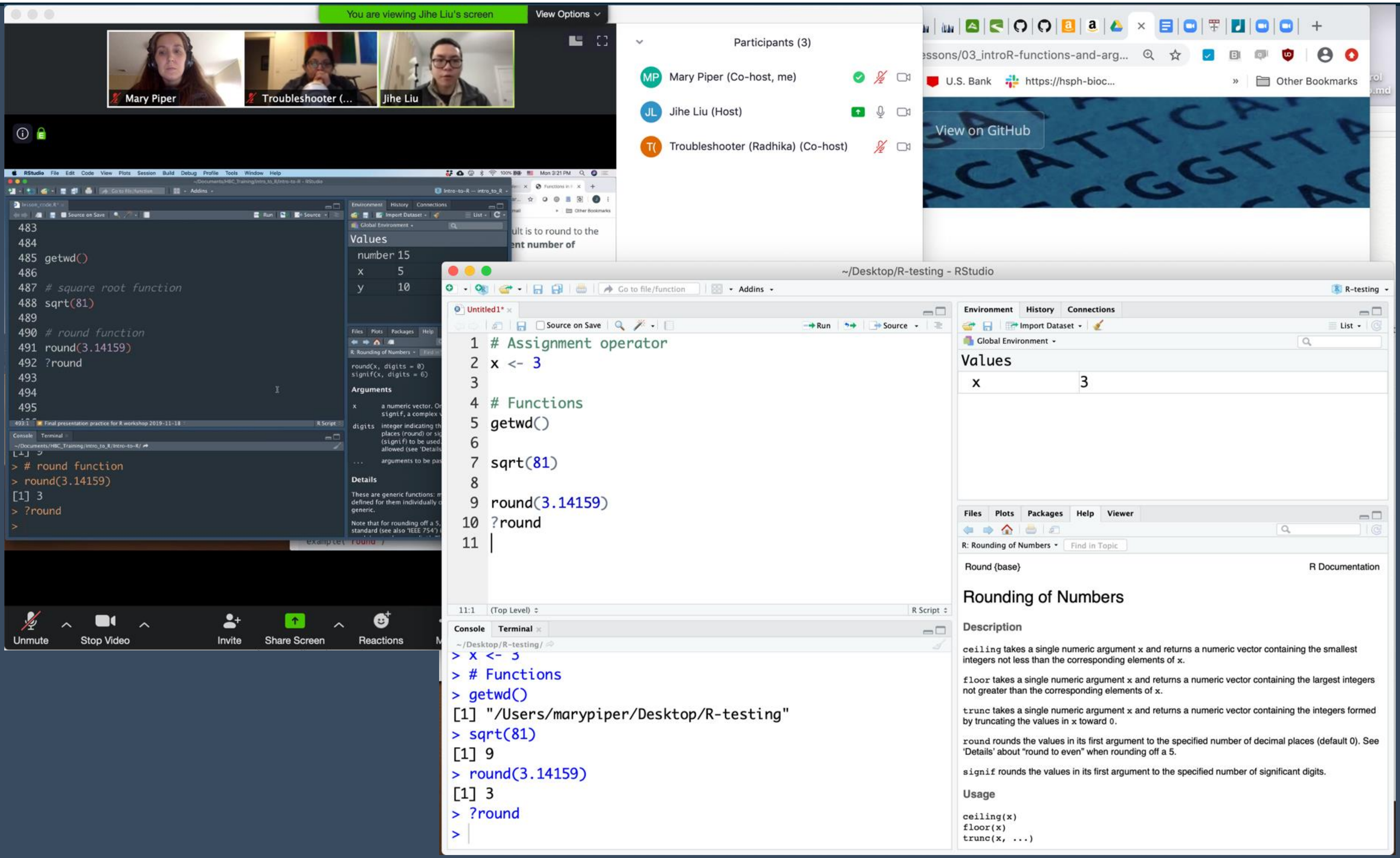
Differential expression between conditions using FindMarkers()

In our current UMAP, we have merged samples across the different conditions and used integration to align cells of the same celltype across samples. Now, what if we were interested in a particular celltype and **understanding how gene expression changes across the different conditions?**

The `FindMarkers()` function in the Seurat package is used to perform differential expression analysis between groups of cells. We provide arguments to specify `ident.1` and `ident.2` the

https://tinyurl.com/hbc-pseudobulk-quarto

Single Screen & 3 Windows



```
getwd()

# square root function
sqrt(81)

# round function
round(3.14159)
?round
```

```
> # round function
> round(3.14159)
[1] 3
> ?round
>
```

Values

number	15
x	5
y	10

```
# Assignment operator
x <- 3

# Functions
getwd()

sqrt(81)

round(3.14159)
?round
```

Values

x	3

```
> x <- 3
> # Functions
> getwd()
[1] "/Users/marypiper/Desktop/R-testing"
> sqrt(81)
[1] 9
> round(3.14159)
[1] 3
> ?round
>
```

Round {base} R Documentation

Rounding of Numbers

Description

`ceiling` takes a single numeric argument `x` and returns a numeric vector containing the smallest integers not less than the corresponding elements of `x`.

`floor` takes a single numeric argument `x` and returns a numeric vector containing the largest integers not greater than the corresponding elements of `x`.

`trunc` takes a single numeric argument `x` and returns a numeric vector containing the integers formed by truncating the values in `x` toward `0`.

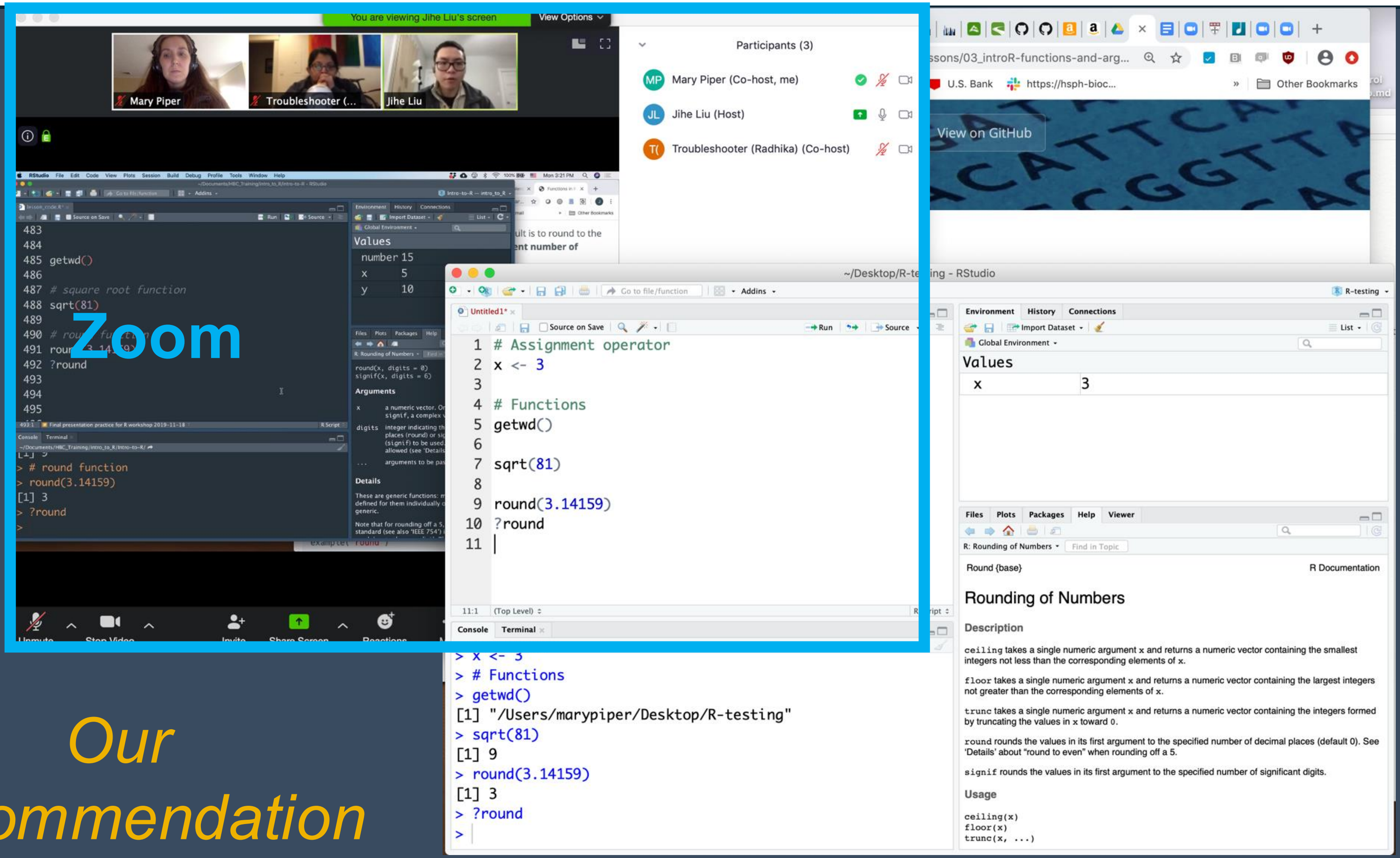
`round` rounds the values in its first argument to the specified number of decimal places (default 0). See 'Details' about "round to even" when rounding off a 5.

`signif` rounds the values in its first argument to the specified number of significant digits.

Usage

```
ceiling(x)
floor(x)
trunc(x, ...)
```

Single Screen & 3 Windows



Zoom

```
# Assignment operator
x <- 3

# Functions
getwd()

sqrt(81)

round(3.14159)
?round
```

```
> x <- 3
> # Functions
> getwd()
[1] "/Users/marypiper/Desktop/R-testing"
> sqrt(81)
[1] 9
> round(3.14159)
[1] 3
> ?round
>
```

Round {base} R Documentation

Rounding of Numbers

Description

`ceiling` takes a single numeric argument `x` and returns a numeric vector containing the smallest integers not less than the corresponding elements of `x`.

`floor` takes a single numeric argument `x` and returns a numeric vector containing the largest integers not greater than the corresponding elements of `x`.

`trunc` takes a single numeric argument `x` and returns a numeric vector containing the integers formed by truncating the values in `x` toward `0`.

`round` rounds the values in its first argument to the specified number of decimal places (default 0). See 'Details' about "round to even" when rounding off a 5.

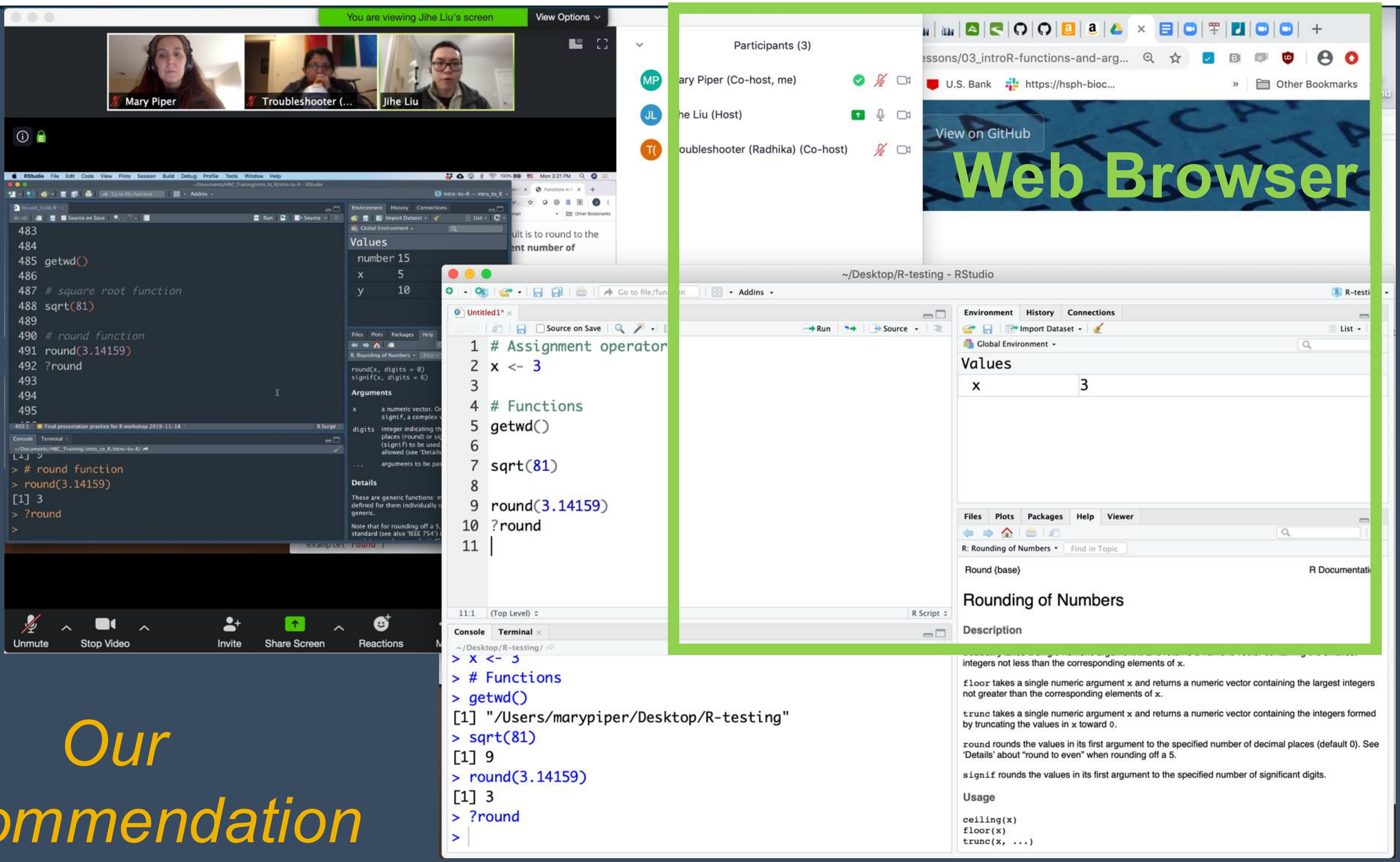
`signif` rounds the values in its first argument to the specified number of significant digits.

Usage

```
ceiling(x)
floor(x)
trunc(x, ...)
```

Our Recommendation

Single Screen & 3 Windows



```
getwd()

# square root function
sqrt(81)

# round function
round(3.14159)
?round
```

```
> # round function
> round(3.14159)
[1] 3
> ?round
>
```

Values

number	15
x	5
y	10

```
# Assignment operator
x <- 3

# Functions
getwd()

sqrt(81)

round(3.14159)
?round
```

Values

x	3

```
> x <- 3
> # Functions
> getwd()
[1] "/Users/marypiper/Desktop/R-testing"
> sqrt(81)
[1] 9
> round(3.14159)
[1] 3
> ?round
>
```

Round {base} R Documentati

Rounding of Numbers

Description

integers not less than the corresponding elements of x.

floor takes a single numeric argument x and returns a numeric vector containing the largest integers not greater than the corresponding elements of x.

trunc takes a single numeric argument x and returns a numeric vector containing the integers formed by truncating the values in x toward 0.

round rounds the values in its first argument to the specified number of decimal places (default 0). See 'Details' about "round to even" when rounding off a 5.

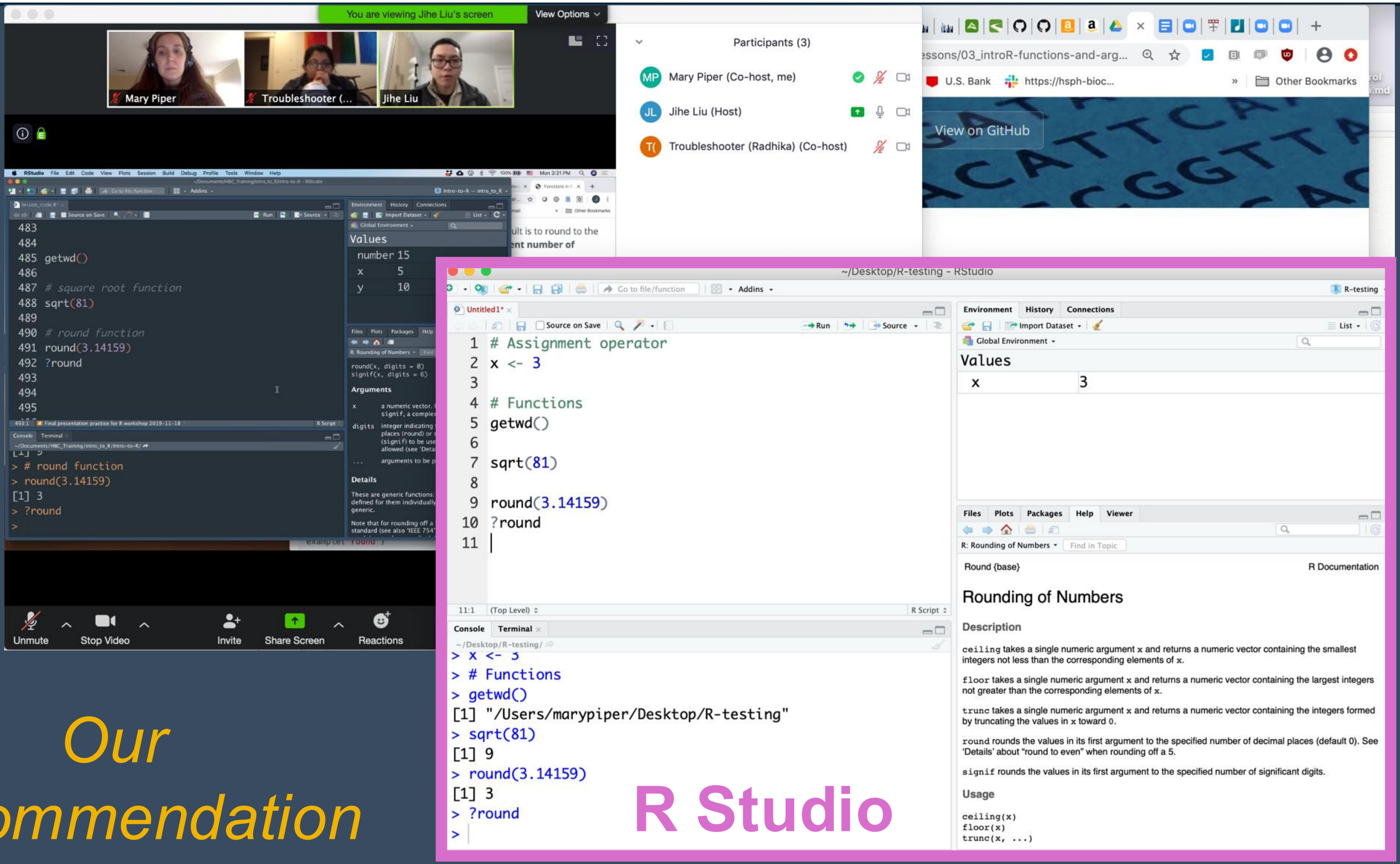
signif rounds the values in its first argument to the specified number of significant digits.

Usage

```
ceiling(x)
floor(x)
trunc(x, ...)
```

Our Recommendation

Single Screen & 3 Windows



```
getwd()

# square root function
sqrt(81)

# round function
round(3.14159)
?round
```

```
> # round function
> round(3.14159)
[1] 3
> ?round
>
```

Values
number 15
x 5
y 10

```
# Assignment operator
x <- 3

# Functions
getwd()

sqrt(81)

round(3.14159)
?round
```

Values
x 3

```
> x <- 3
> # Functions
> getwd()
[1] "/Users/marypiper/Desktop/R-testing"
> sqrt(81)
[1] 9
> round(3.14159)
[1] 3
> ?round
>
```

Round {base} R Documentation

Rounding of Numbers

Description

`ceiling` takes a single numeric argument x and returns a numeric vector containing the smallest integers not less than the corresponding elements of x.

`floor` takes a single numeric argument x and returns a numeric vector containing the largest integers not greater than the corresponding elements of x.

`trunc` takes a single numeric argument x and returns a numeric vector containing the integers formed by truncating the values in x toward 0.

`round` rounds the values in its first argument to the specified number of decimal places (default 0). See 'Details' about "round to even" when rounding off a 5.

`signif` rounds the values in its first argument to the specified number of significant digits.

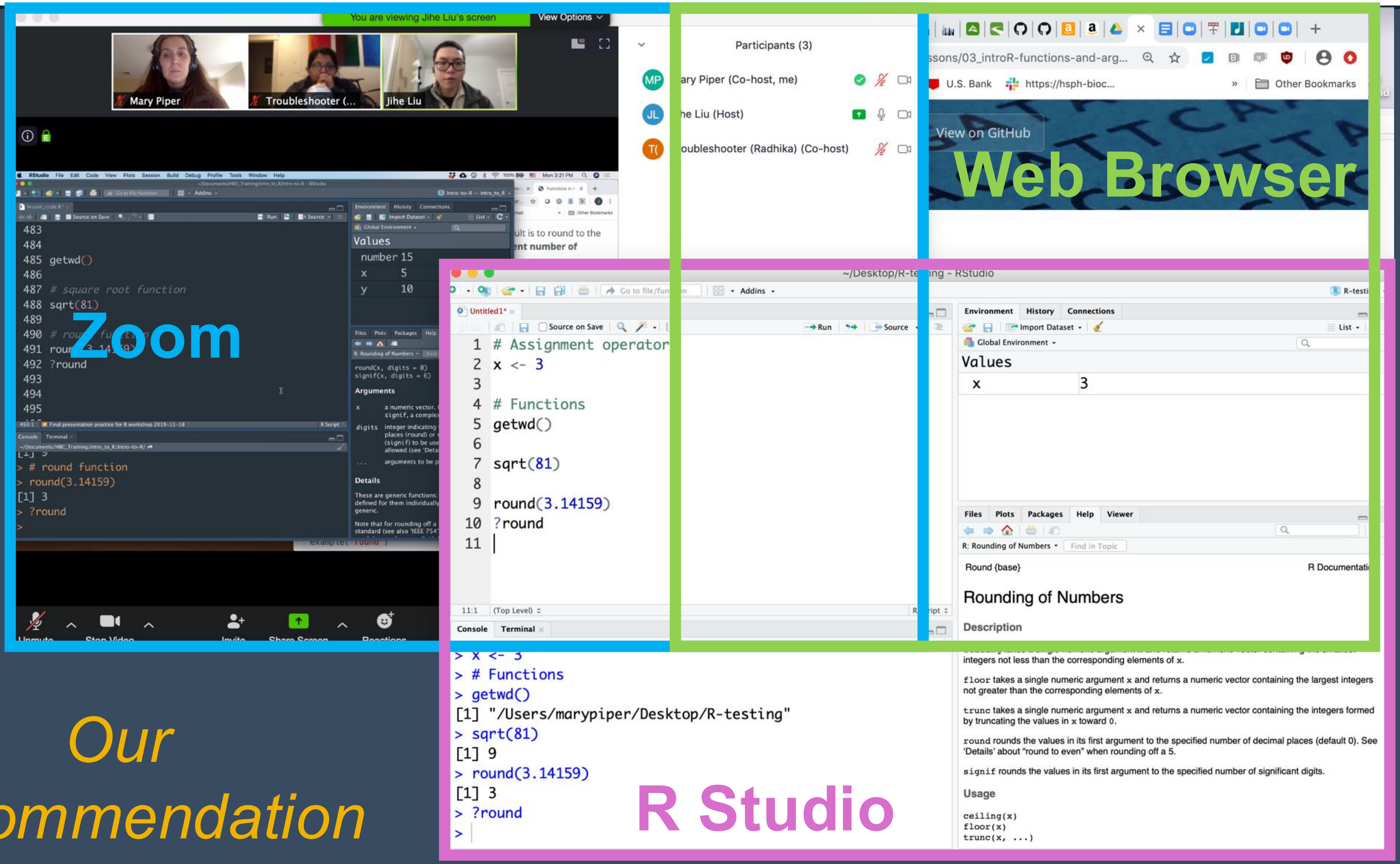
Usage

```
ceiling(x)
floor(x)
trunc(x, ...)
```

R Studio

Our Recommendation

Single Screen & 3 Windows



```r
# Assignment operator
x <- 3

# Functions
getwd()

sqrt(81)

round(3.14159)
?round
```

```
> x <- 3
> # Functions
> getwd()
[1] "/Users/marypiper/Desktop/R-testing"
> sqrt(81)
[1] 9
> round(3.14159)
[1] 3
> ?round
>
```

Round {base} R Documentati

Rounding of Numbers

Description

integers not less than the corresponding elements of x.

floor takes a single numeric argument x and returns a numeric vector containing the largest integers not greater than the corresponding elements of x.

trunc takes a single numeric argument x and returns a numeric vector containing the integers formed by truncating the values in x toward 0.

round rounds the values in its first argument to the specified number of decimal places (default 0). See 'Details' about "round to even" when rounding off a 5.

signif rounds the values in its first argument to the specified number of significant digits.

Usage

```
ceiling(x)
floor(x)
trunc(x, ...)
```

Our Recommendation

Course participation

- Mandatory review of self-learning lessons and assignments
- Attendance required for all classes
- Your questions and active participation drive learning
- **We look forward to all of your questions!**

Course participation

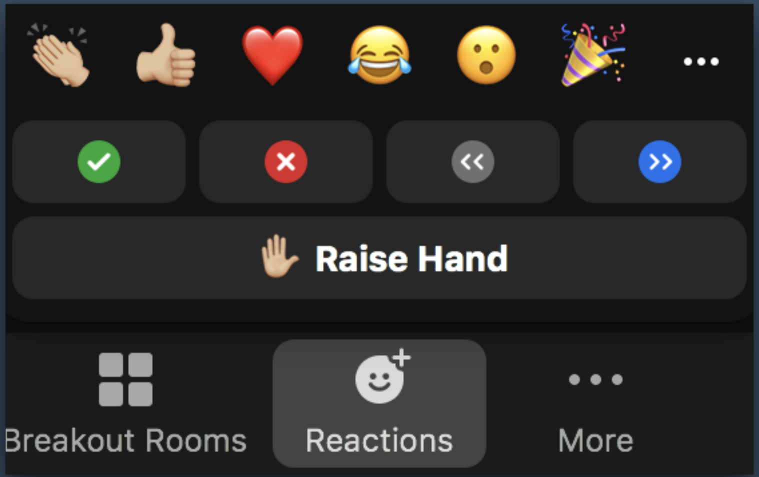
- At-home lessons and exercises after each session
- Cover material not previously discussed
- Provides us feedback to help pace the course appropriately
- 3-5 hours to complete
- Homework load is heavier in the beginning of this workshop series and tapers off

Using AI for Assignments

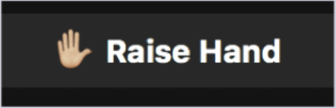
- Do
  - Try to resolve error messages with it
  - Test code written by AI on a dataset where you have expected results
  - Take the time to review the generated code line-by-line
- Don’t
  - Implement it in replacement to learning
  - Write code that you don’t understand
  - Assume the output from an AI process is correct

Odds & Ends

- Quit/minimize all applications that are not required for class
- Are you all set?
  - ✓ = "agree", "I'm all set"
  - ✗ = "disagree", "I need help"

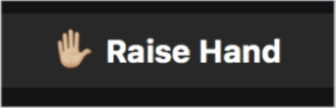


Odds & Ends

- Questions for the presenter?
  - Post the question in the Chat window OR
  - 

 when the presenter asks for questions
  - Let the Moderator know

Odds & Ends

- Questions for the presenter?
    - Post the question in the Chat window OR
    - 

when the presenter asks for questions
    - Let the Moderator know
- Technical difficulties with software?
    - Start a private chat with the Troubleshooter with a description of the problem

Contact Us

- *HBC training team:* hbctraining@hsph.harvard.edu
- *HBC consulting:* bioinformatics@hsph.harvard.edu