

CellMixS

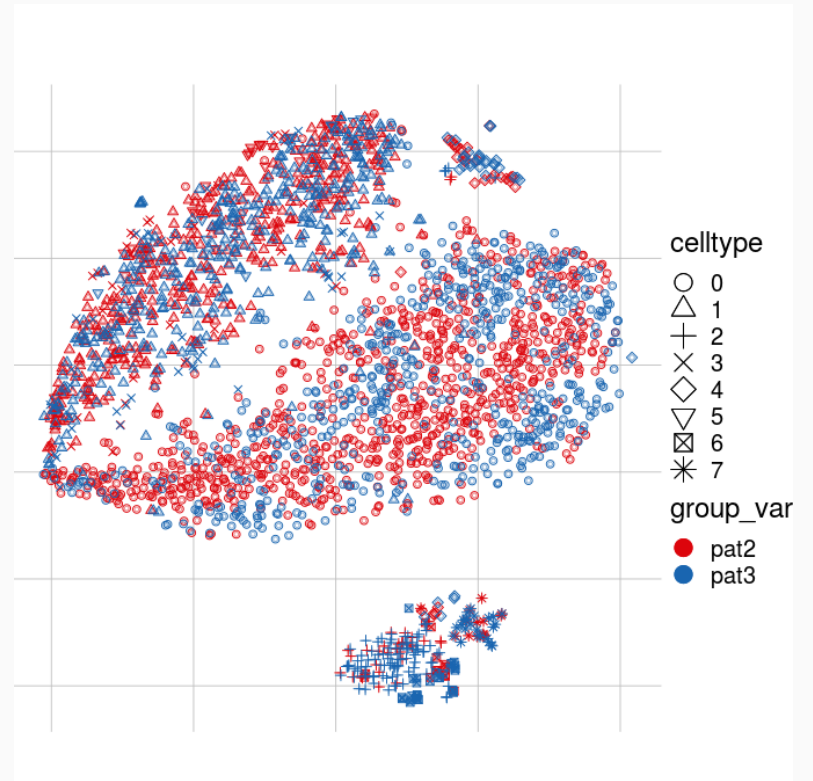
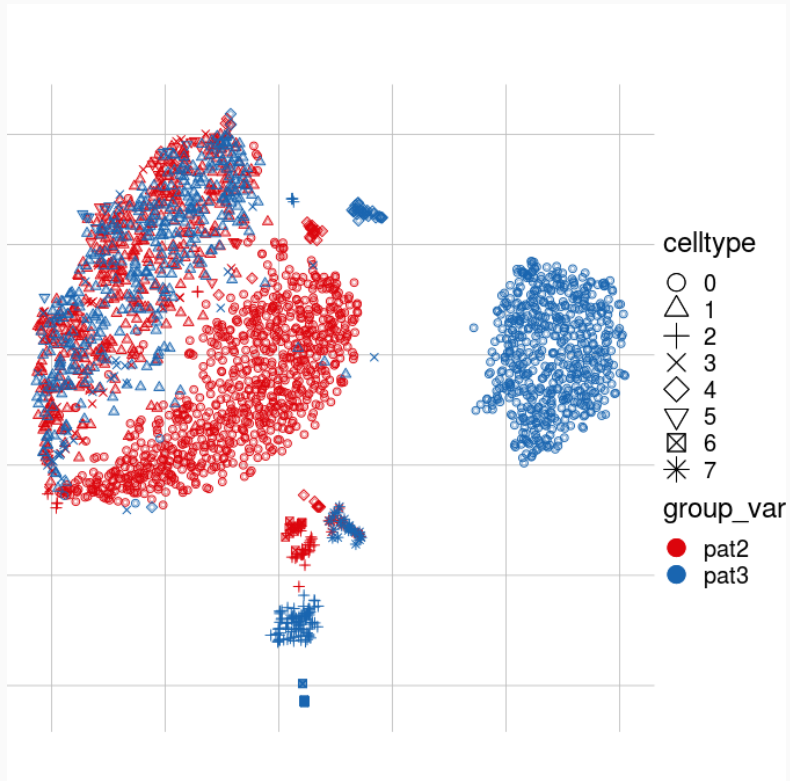
Explore data integration and batch effects

Almut Lütge

DMLS - University of Zürich

2019-12-08

Batch effects





```
library(CellMixS)
```

```
# Input data SingleCellExperiment object
```

```
sce1 <- readRDS("data/sim_batch_2.rds")
```

```
sce2 <- readRDS("data/sim_batch_1.5.rds")
```

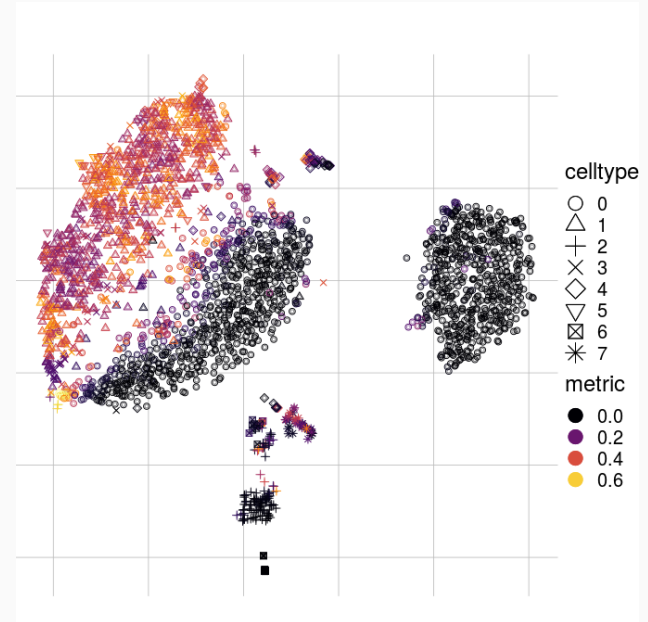
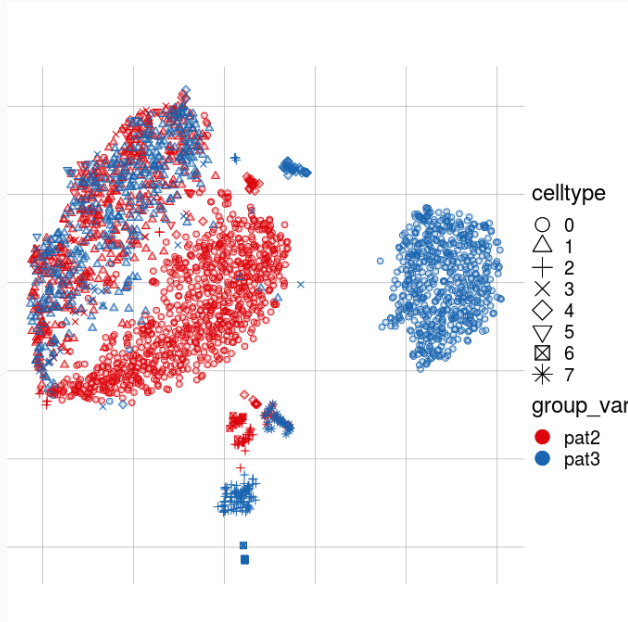
```
# Get Cellspecific Mixing Score
```

```
sce1 <- evalIntegration(metrics = "cms", sce1, group = "batch_id", k = 100)
```

```
sce2 <- evalIntegration(metrics = "cms", sce2, group = "batch_id", k = 100)
```

Cellspecific Mixing Score

Cellspecific Mixing Score



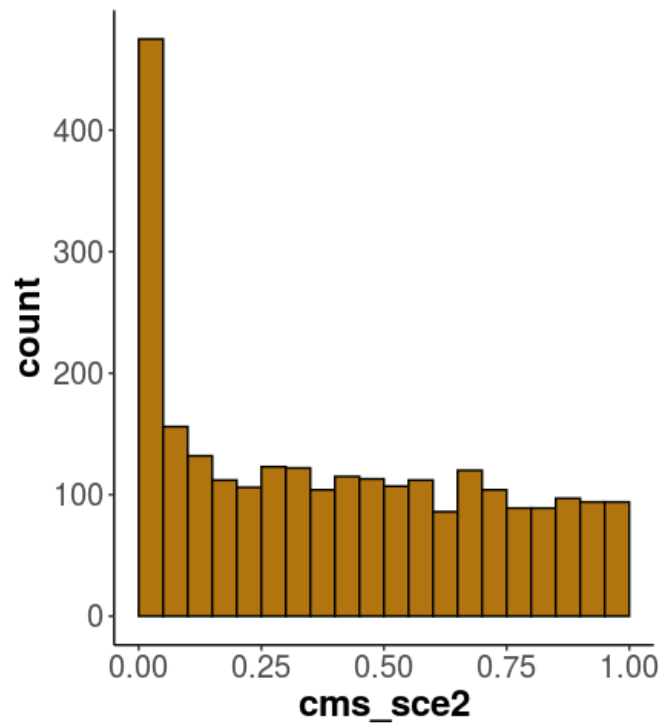
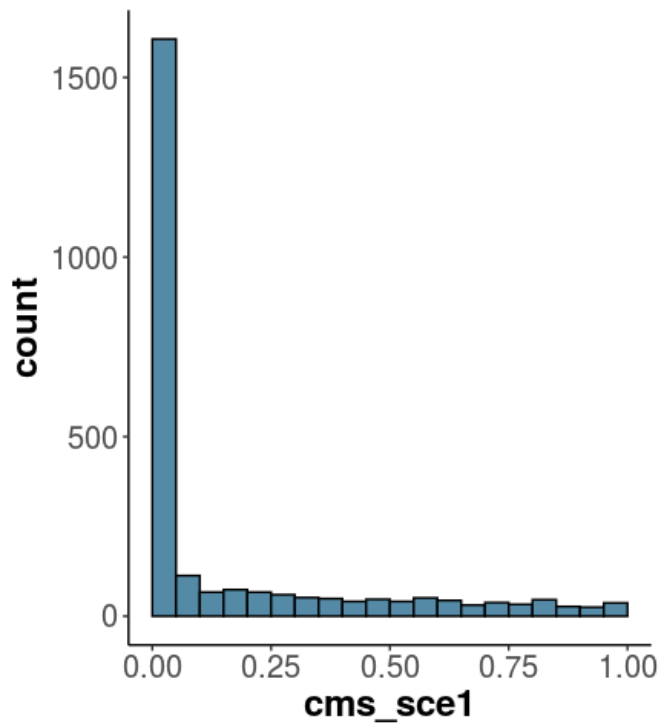
```
# show groups
```

```
visGroup(sce1, "batch_id")
```

```
# show metric
```

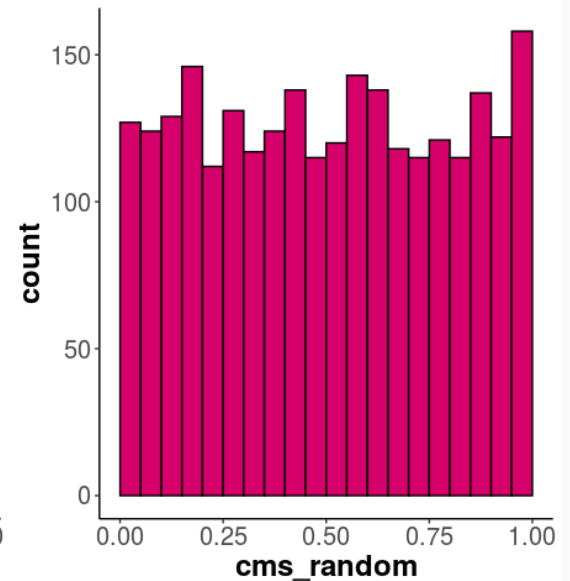
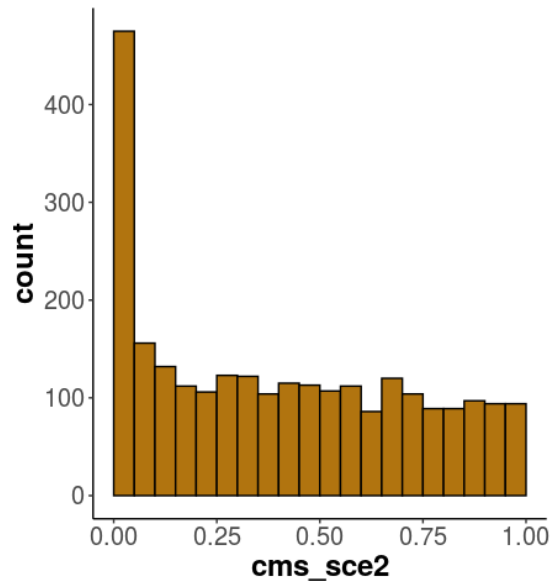
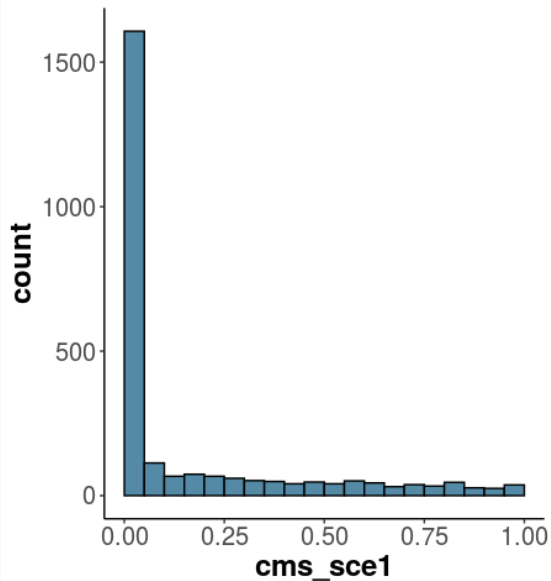
```
visMetric(sce1, "cms_smooth")
```

Interpretation



```
#get cms scores from sce  
cms_res <- data_frame("cms_sce1" = sim_batch2$cms, "cms_sce2" = sim_batch$cms)  
  
# show scores as histogram  
p <- visHist(cms_res, n_col = 2)
```

Random batches



```
#assign random batch label
sce2$random_batch <- sample(sce2$batch_id, ncol(sce), replace = FALSE)

# run cms
sce2 <- evalIntegration("cms", sce = sce2, group = "random_batch", k = 100,
  res_name = "cms_random")

# show scores as histogram
cms_res$cms_random <- sce2$cms_random
p <- visHist(cms_res, n_col = 3)
```


More options

More options

More functionalities

- Adjust neighbourhoods (important for unbalanced batches)
- Group by celltype
- Compare Integration

More metrics

- entropy
- inverse Simpson Index
- mixingMetric (Seurat)
- Local density factor
- LocalStructure (Seurat)

Summary

- Quantify (batch-) mixing
 - Cellspecific
 - Neighbourhood adjustment (Unbalanced batches!)
 - Different metrics
- Quantify structural changes upon integration
 - cellspecific
- Visualize mixing

