13-15 March 2017, CyTOF workshop at UZH

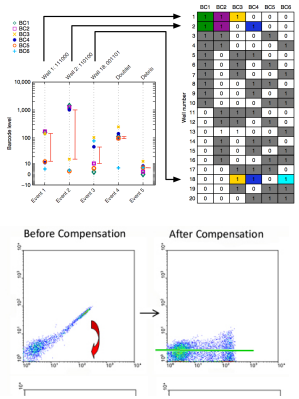
Analysis of high dimensional cytometry (HDCyto) data in R

Mark D. Robinson
Lukas M. Weber
Statistical Bioinformatics Group, DMLS@UZH+SIB

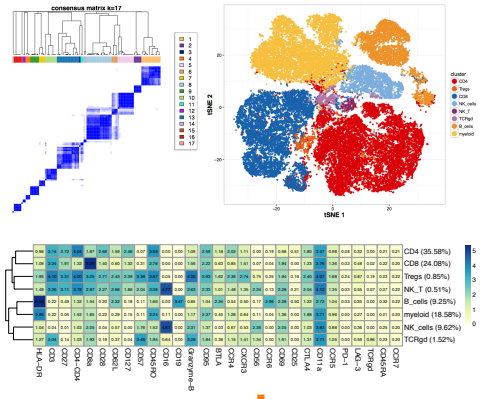
Some slides adapted from Felix Hartmann

Overall view of analysis of **HDCyto** data

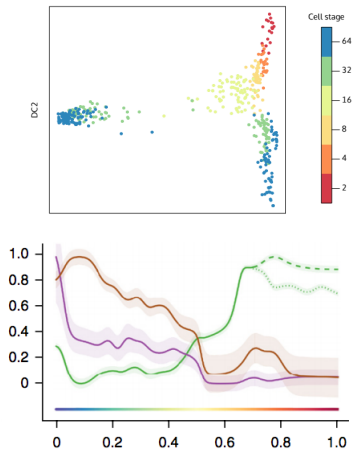
a. normalization, debarcoding, compensation



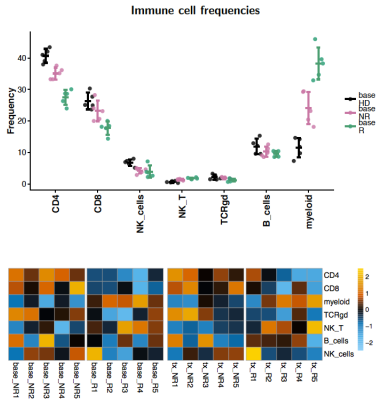
b. clustering, quantification, visualization



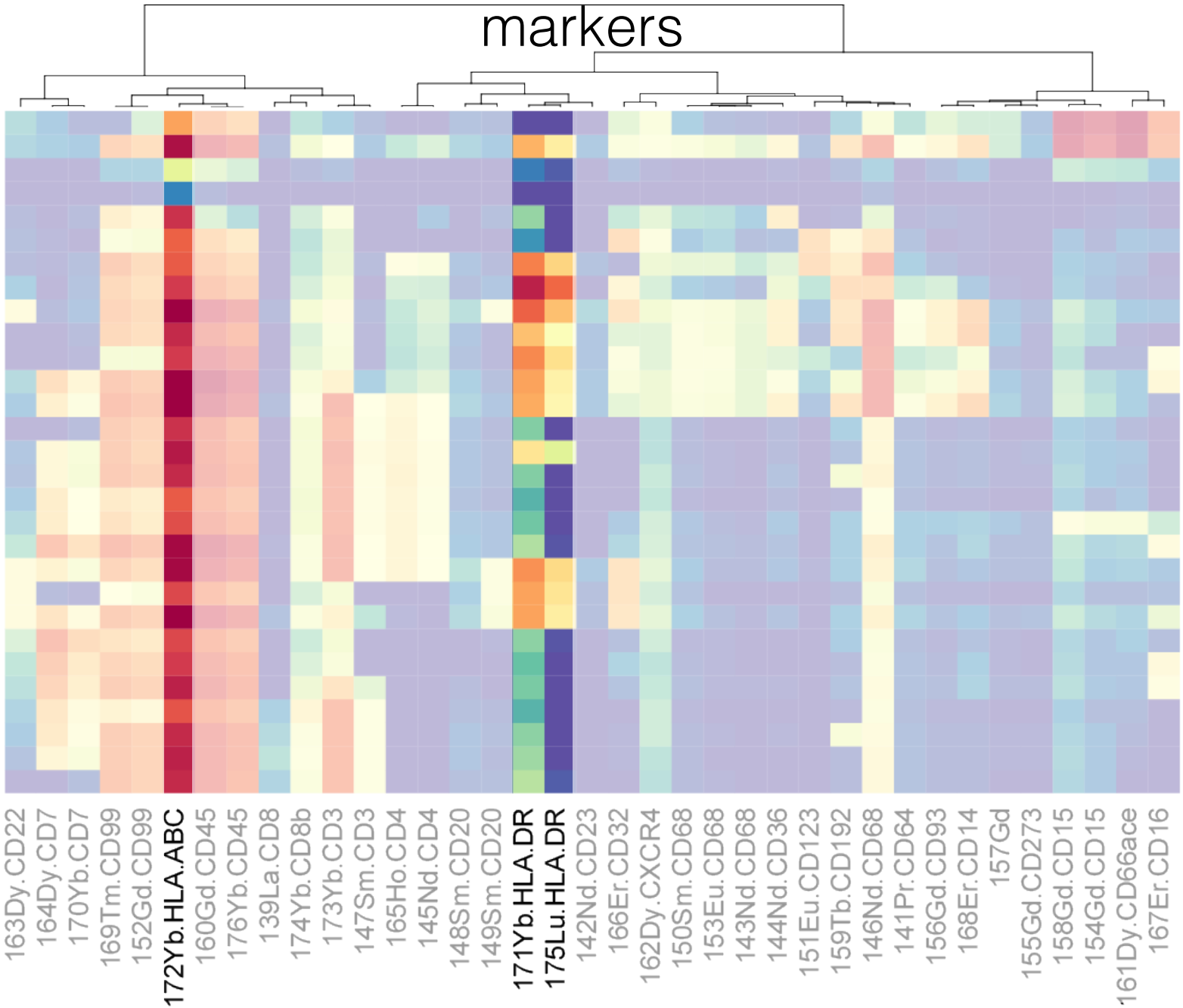
d. trajectory analyses



c. general differential analyses

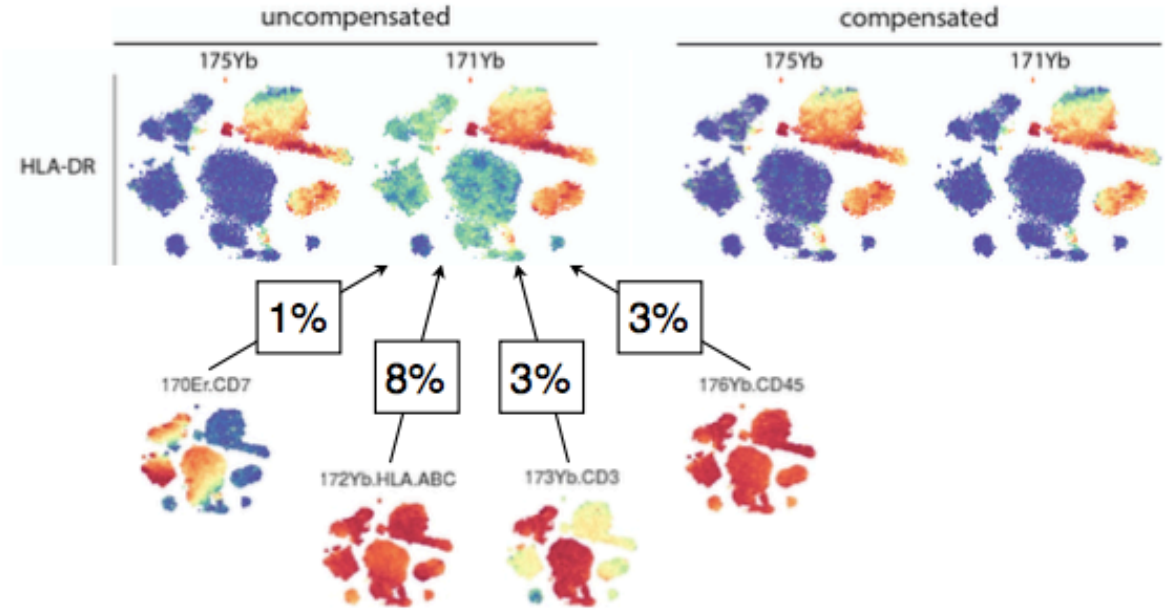


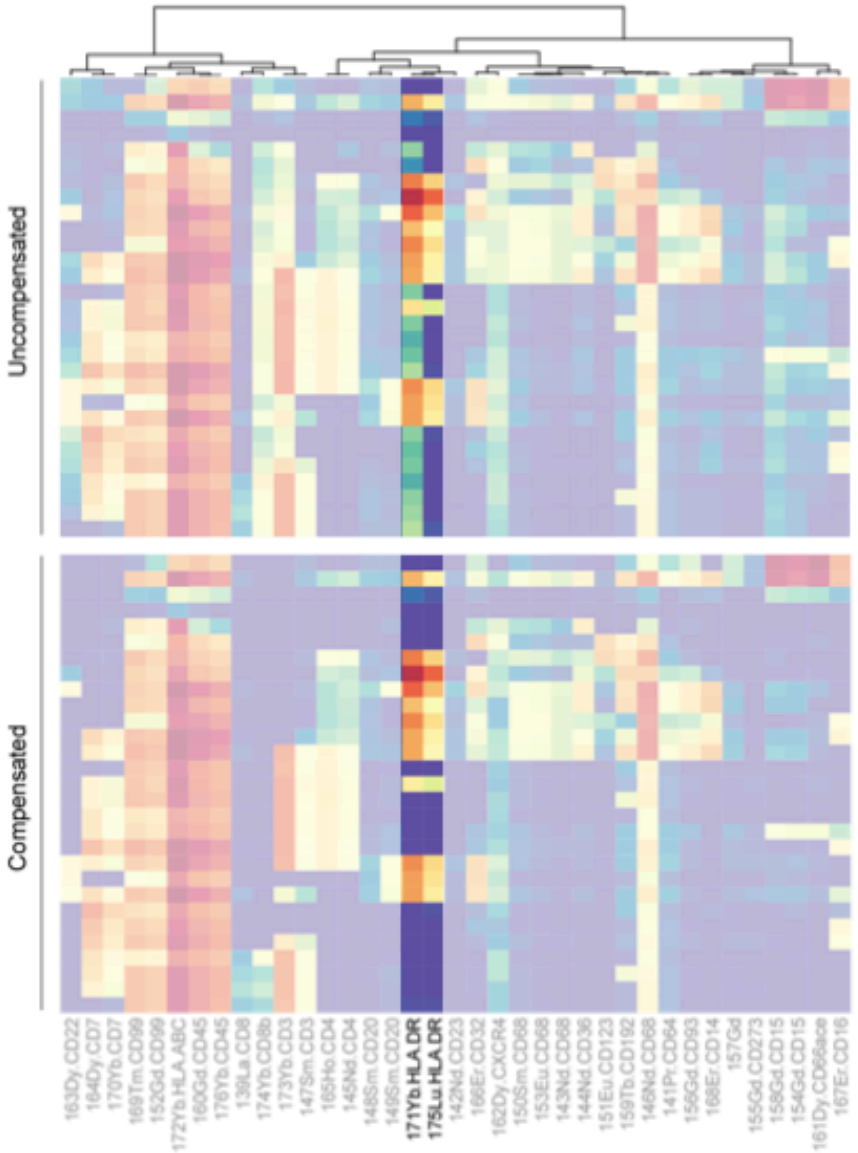
Compensation: is it needed?



PBMCs measured, clustered with Phenograph; **several antibodies used twice** with different metals

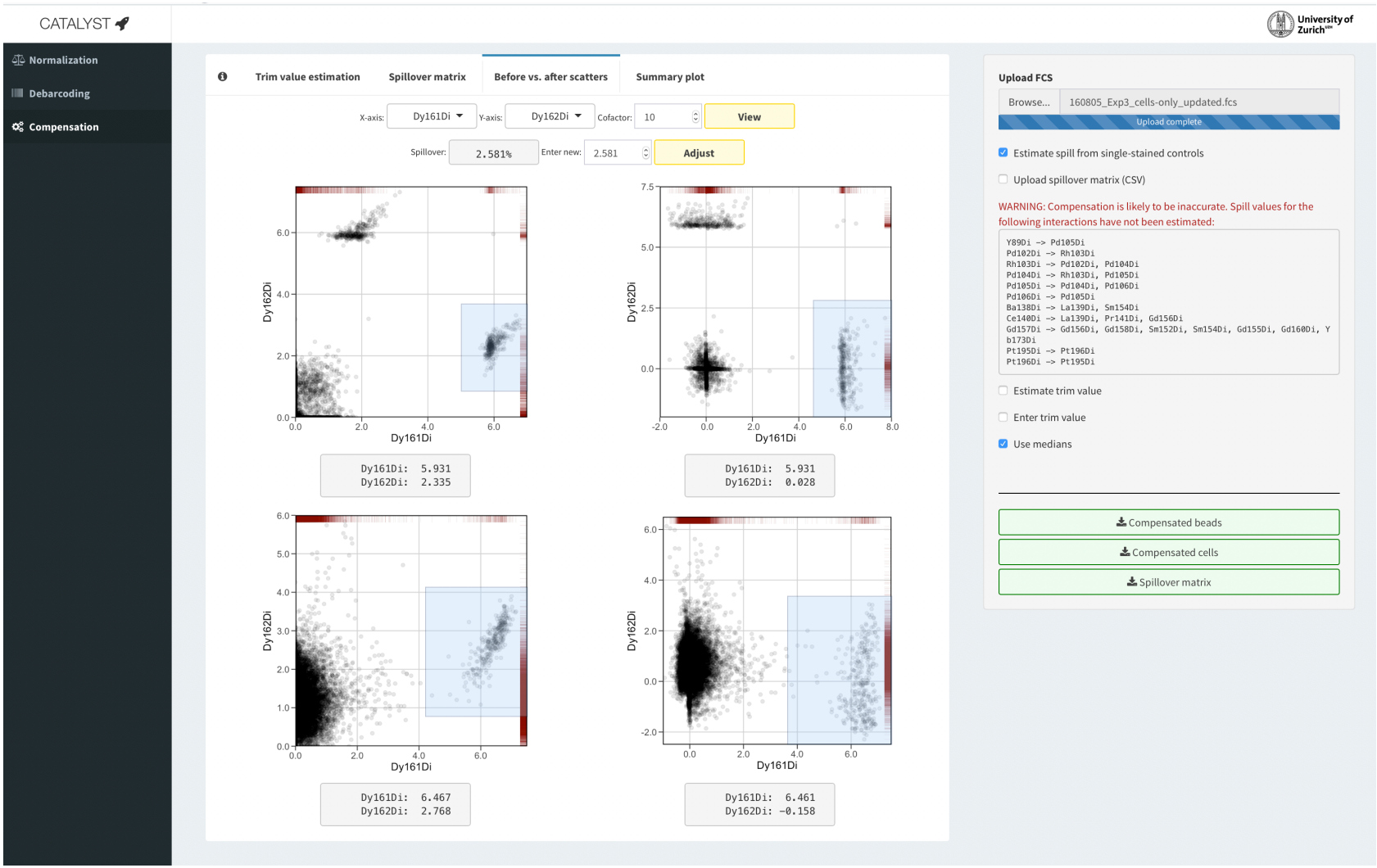
Correction of spillover artefacts on a 36ab panel



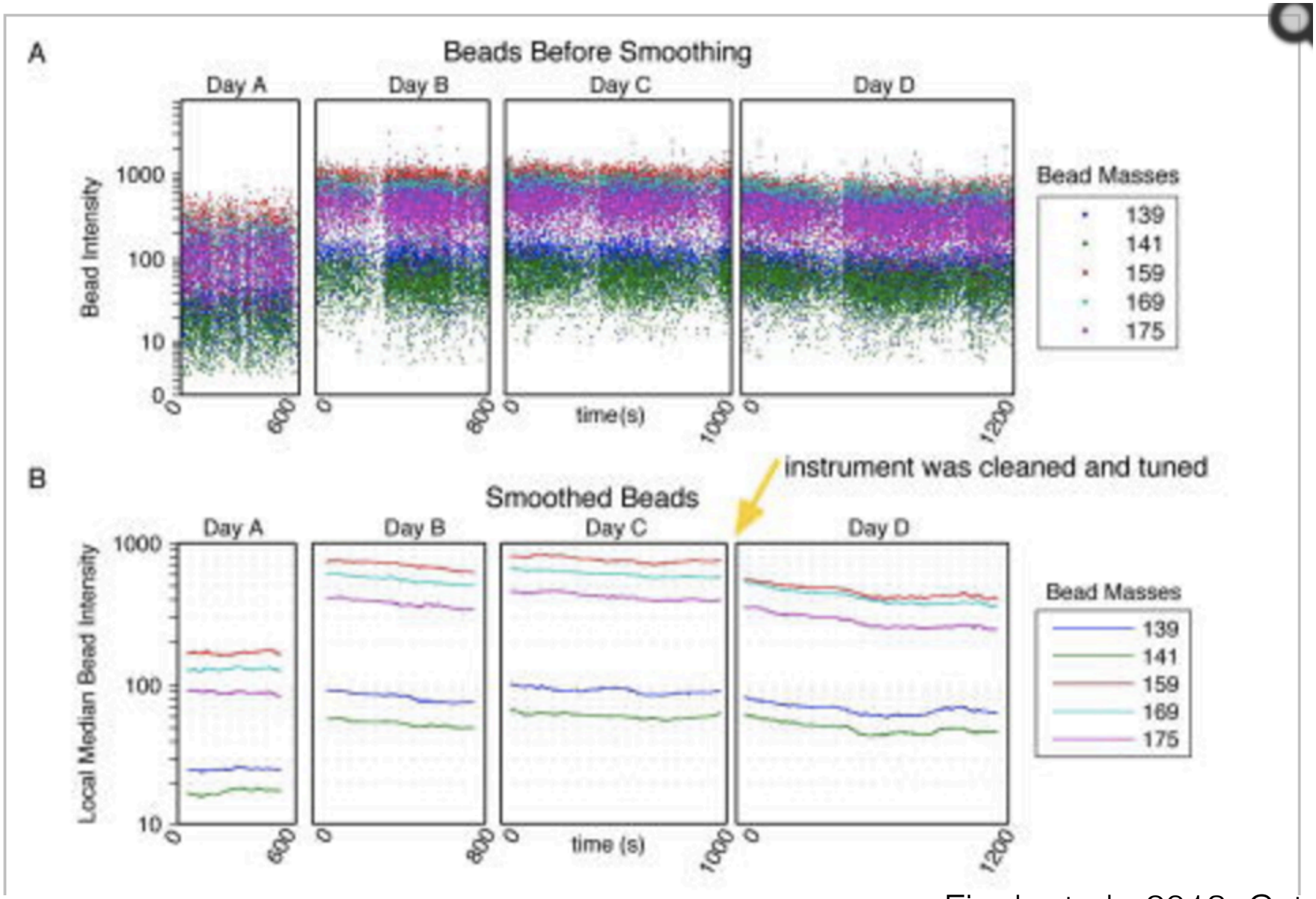


R/Bioconductor CATALYST package + web app **coming soon**: *normalization*, *debarcoding*, *compensation*

Helena Crowell



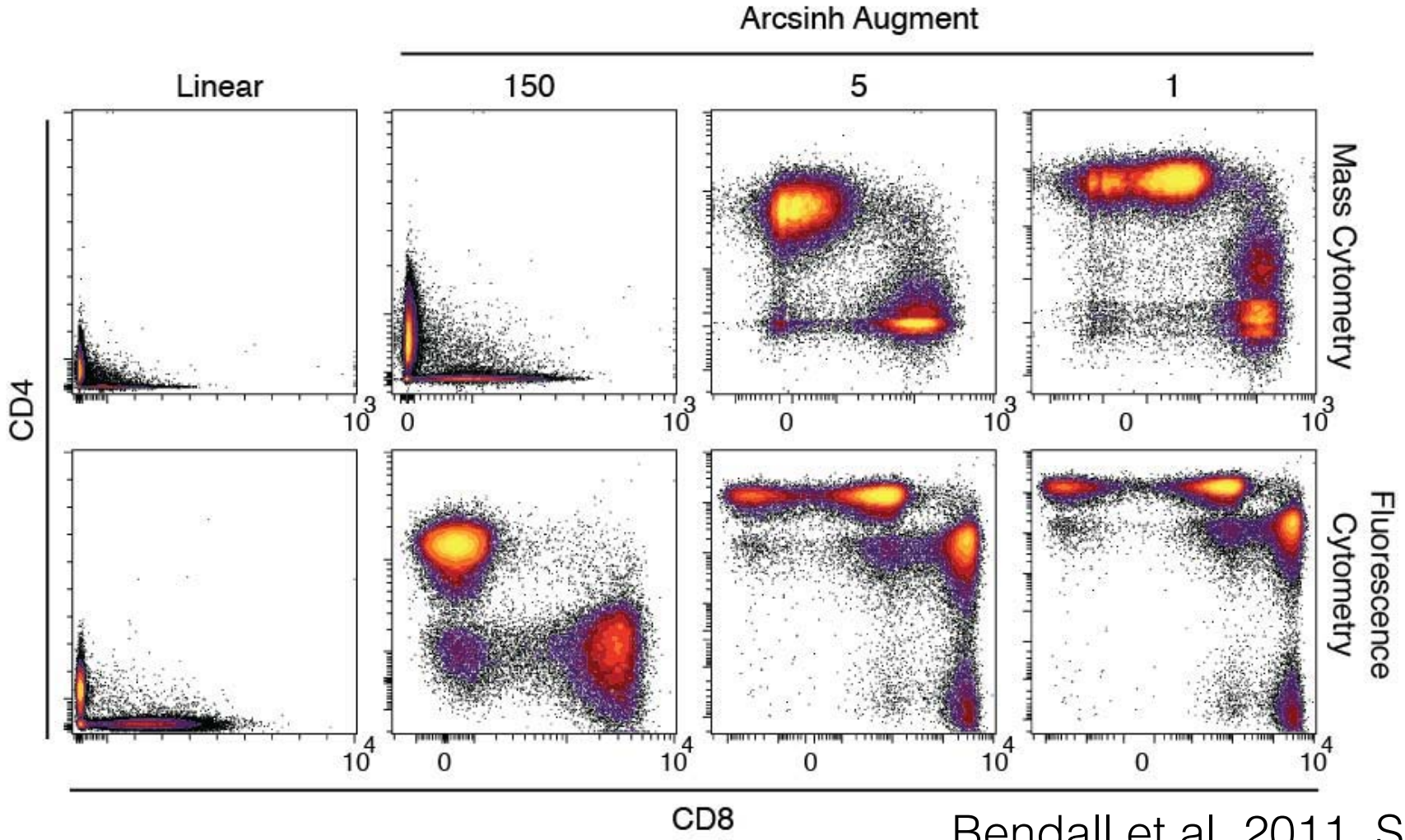
Finck normalization



Finck et al., 2013, Cytometry A

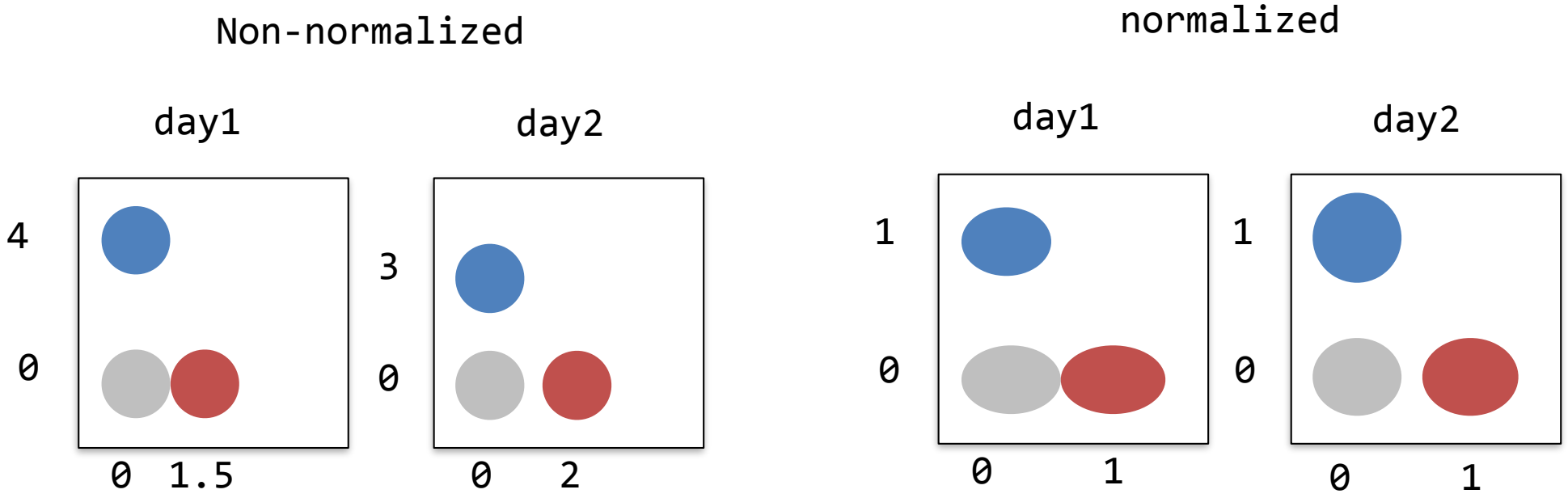
Data transformation: biexponential, logicle, **arcsinh**

Figure S2



Bendall et al. 2011, Science

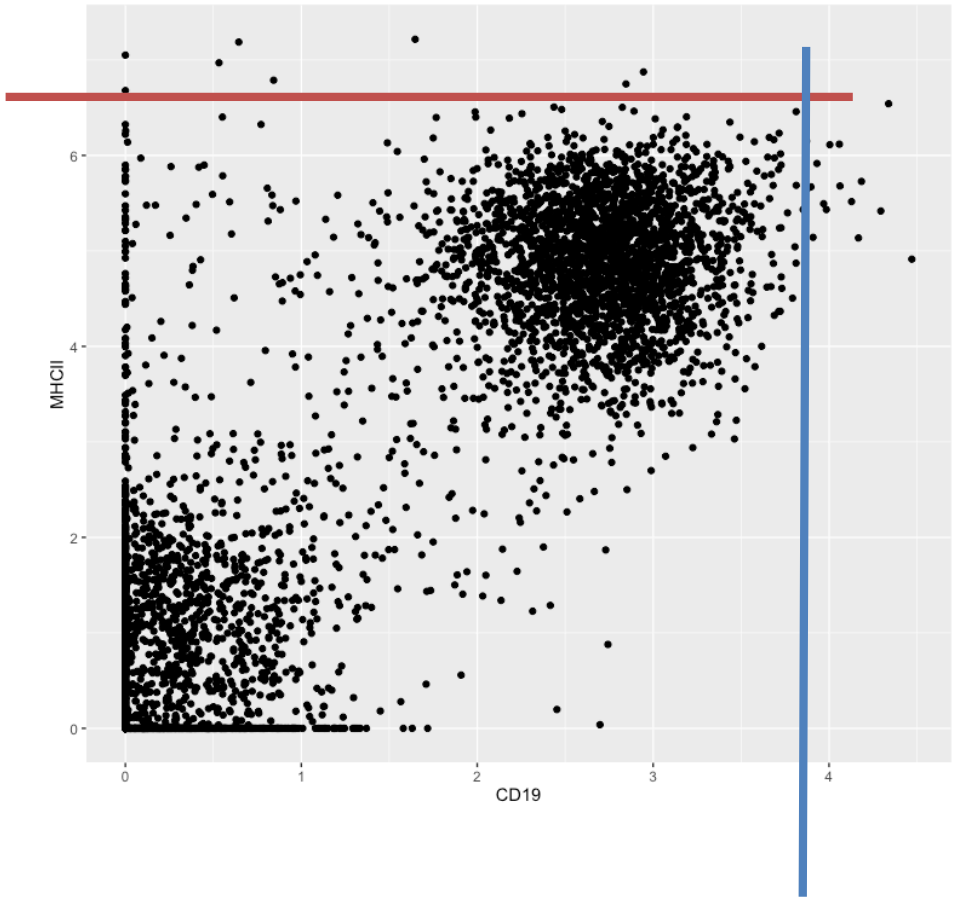
After arcsine transform, optional normalization to 0-1



Often useful for visualisation, unclear if useful for quantitative analyses

Percentile normalization

99th percentile of MHCII (6.7)

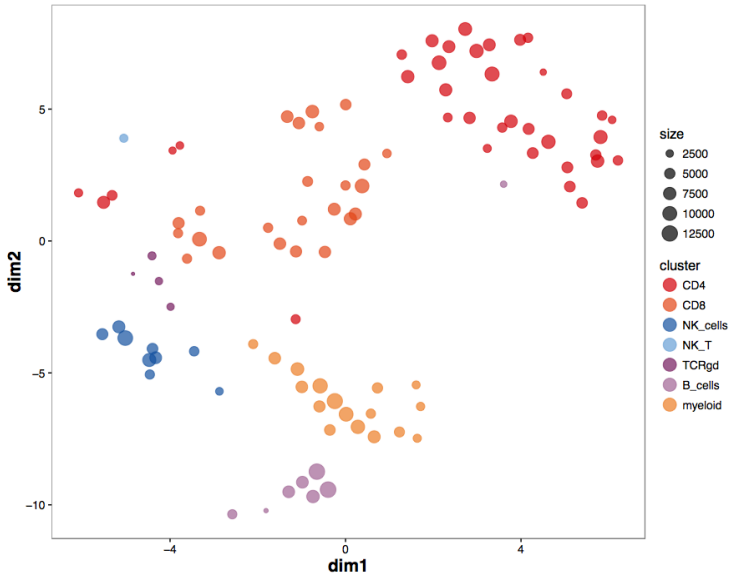


	CD3	CD4	CD8
Cell_1	8	0	12
Cell_2	10	6	1

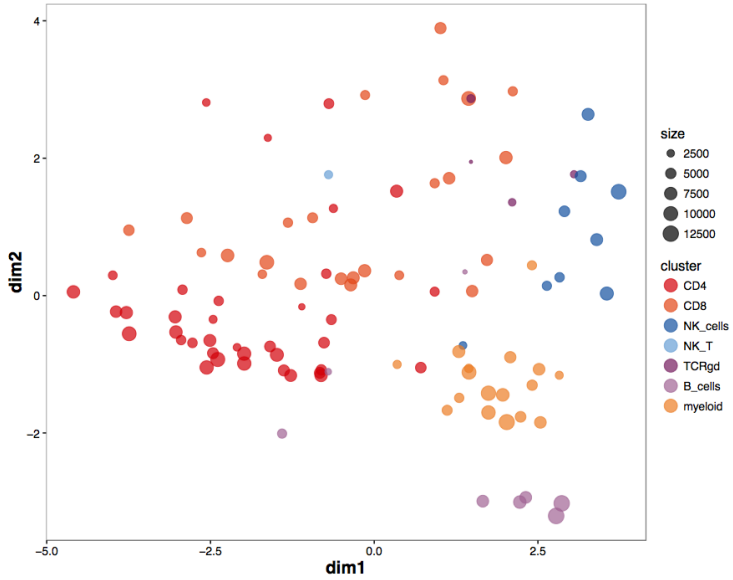
99th percentile of CD19 (3.9)

Visualization -> Dimension reduction

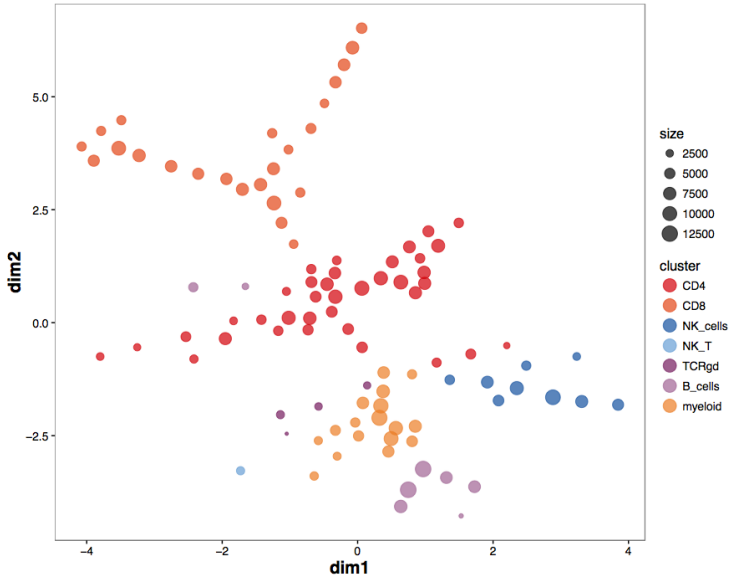
tSNE



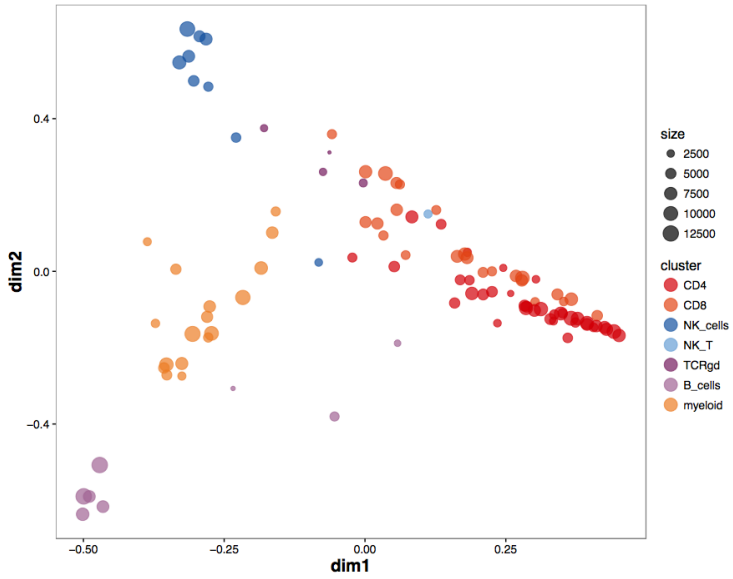
PCA



MST

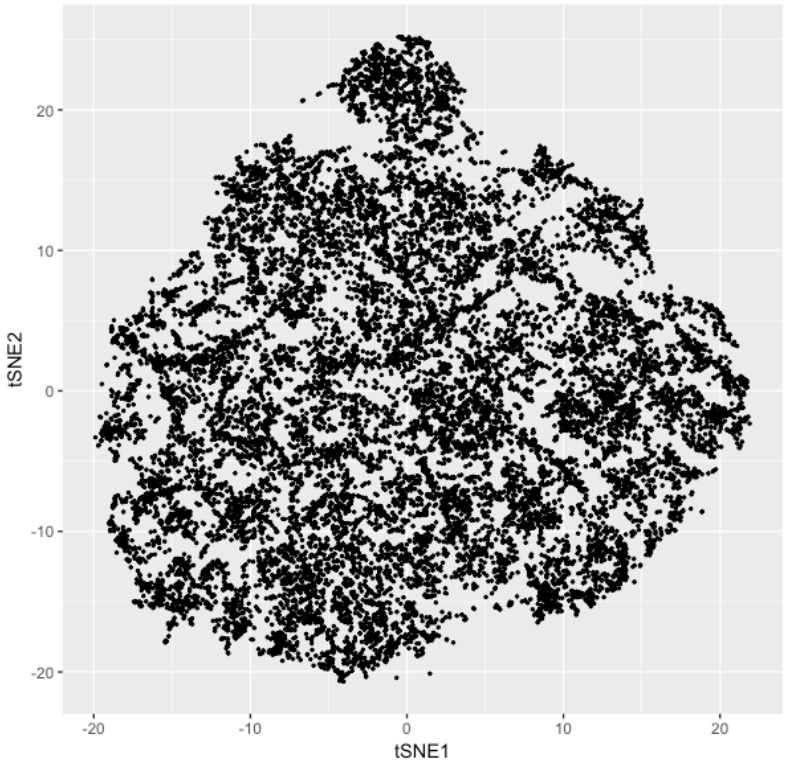


Diffusion Maps



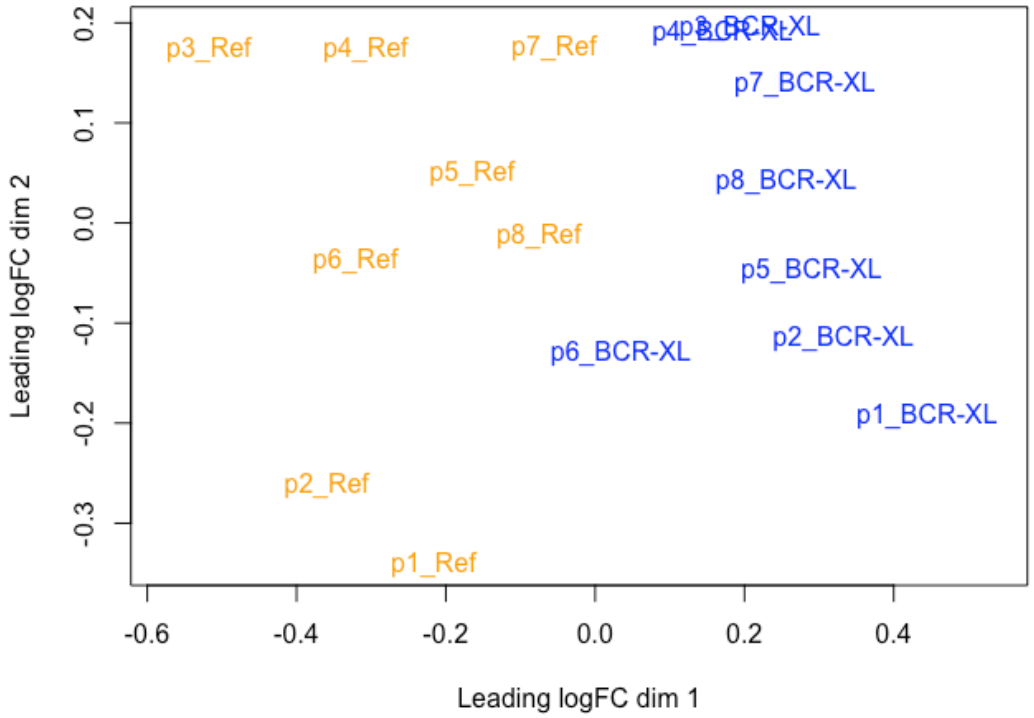
Dimension reduction useful in both directions: cells + samples

N cells x K markers —>
N cells x 2 dimensions



Each point = **cell**

M cluster (frequencies) x P samples
—> P samples x 2 dimensions

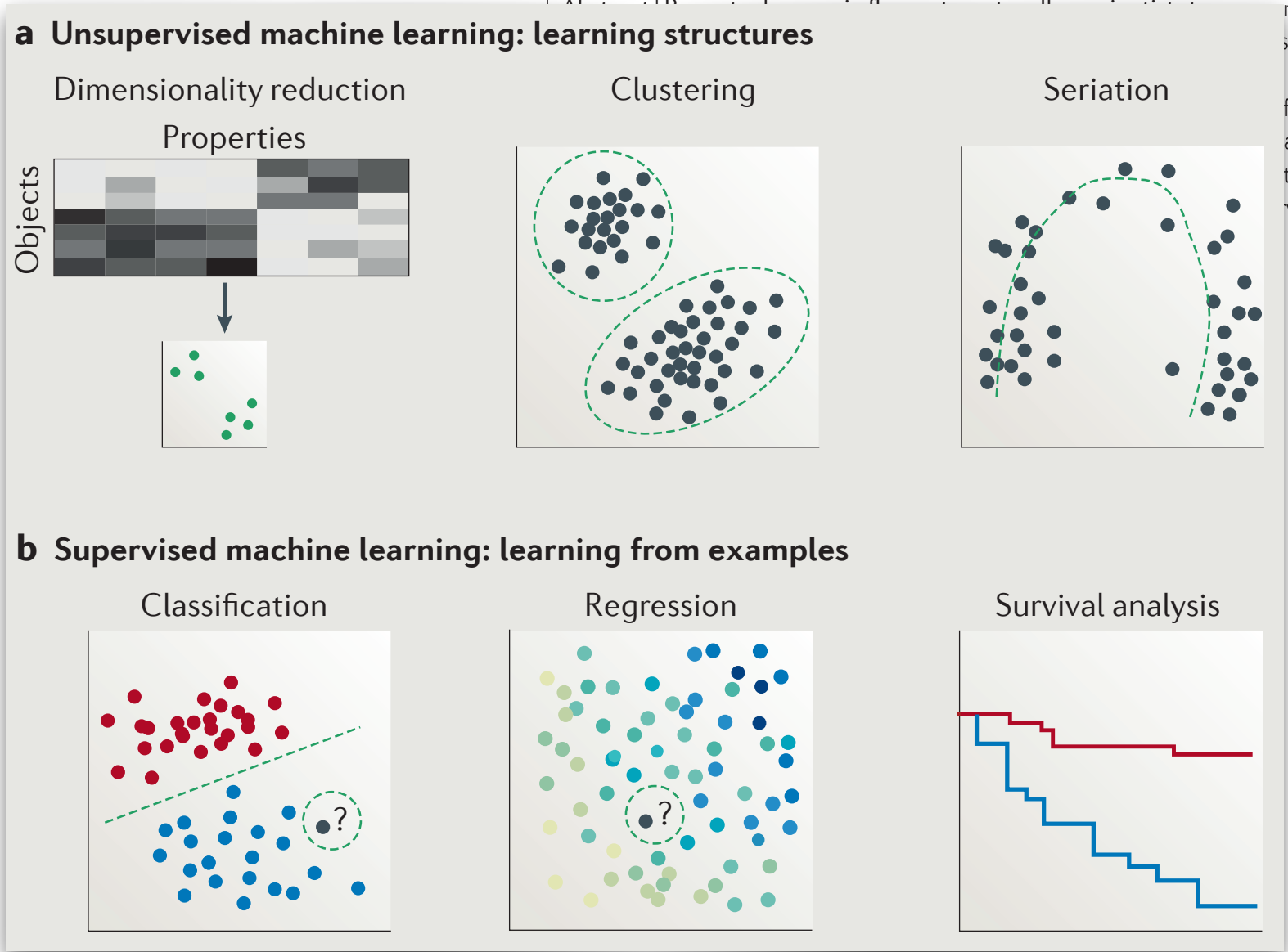


Each point = **sample**

Computational flow cytometry: helping to make sense of high-dimensional immunology data

Yvan Saeys[1,2], Sofie Van Gassen[1,3] and Bart N. Lambrecht[1,2,4]

re an increasing number
s. To analyse, visualize and
be adopted, evaluated
flow cytometry is
and computational
throughput single-cell
view of the many recent



Clustering high-dimensional flow and mass cytometry

Motivation: Many new computational methods, explosion in the number of dimensions (both FACS and CyTOF) — what works “best”?

Lukas Weber

EDITOR'S CHOICE

Cytometry PART A
Journal of the International Society for Advancement of Cytometry

Comparison of Clustering Methods for High-Dimensional Single-Cell Flow and Mass Cytometry Data

Lukas M. Weber,[1,2] Mark D. Robinson[1,2]*

preprint April 2016, published Dec 2016

Table 1. Overview of clustering methods compared in this study

METHOD	ENVIRONMENT AND AVAILABILITY	SHORT DESCRIPTION	REF.
ACCENSE	Standalone application with graphical interface	Nonlinear dimensionality reduction (t-SNE) followed by density-based peak-finding and clustering in two-dimensional projected space.	22
ClusterX	R package (cytofkit) from Bioconductor	Density-based clustering on t-SNE projection map; faster than DensVM.	23
DensVM	R package (cytofkit) from Bioconductor	Density-based clustering on t-SNE projection map; similar to ACCENSE, with additional support vector machine to classify uncertain points.	24
FLOCK	C source code (also available in ImmPort online platform)	Partitioning of each dimension into bins, followed by merging of dense regions, and density-based clustering.	25
flowClust	R package from Bioconductor	Model-based clustering based on multivariate *t* mixture models with Box-Cox transformation.	26
flowMeans	R package from Bioconductor	Based on k-means, with merging of clusters to allow non-spherical clusters.	27
flowMerge	R package from Bioconductor	Extension of flowClust; merges cluster mixture components from flowClust.	28
flowPeaks	R package from Bioconductor	Peak-finding on smoothed density function generated by k-means; using finite mixture model.	29
FlowSOM	R package from Bioconductor	Self-organizing maps, followed by hierarchical consensus meta-clustering to merge clusters.	30
FlowSOM_pre	R package from Bioconductor	Same as FlowSOM, but without the final consensus meta-clustering step.	30
immunoClust	R package from Bioconductor	Iterative clustering based on finite mixture models, using expectation maximization and integrated classification likelihood.	31
k-means	R base packages (stats)	Standard k-means clustering.	
PhenoGraph	Graphical interface (cyt) launched from MATLAB (Python implementation also available)	Construction of nearest-neighbor graph, followed by partitioning of the graph into sets of highly interconnected points ("communities").	18
Rclusterpp	R package from GitHub (older version on CRAN)	Large-scale implementation of standard hierarchical clustering, with improved memory requirements.	32
SamSPECTRAL	R package from Bioconductor	Spectral clustering, with modifications for improved memory requirements.	33
SPADE	R package from GitHub (older version on Bioconductor; also available in Cytobank online platform)	"Spanning-tree progression analysis of density-normalized events"; organizes clusters into a branching hierarchy of related phenotypes.	34
SWIFT	Graphical interface launched from MATLAB	Iterative fitting of Gaussian mixture models by expectation maximization, followed by splitting and merging of clusters using a unimodality criterion.	35
X-shift	Standalone application (VorteX) with graphical interface (command-line version also available)	Weighted k-nearest-neighbor density estimation, detection of local density maxima, connection of points via graph, and cluster merging.	17

Table 1

Manually gated populations

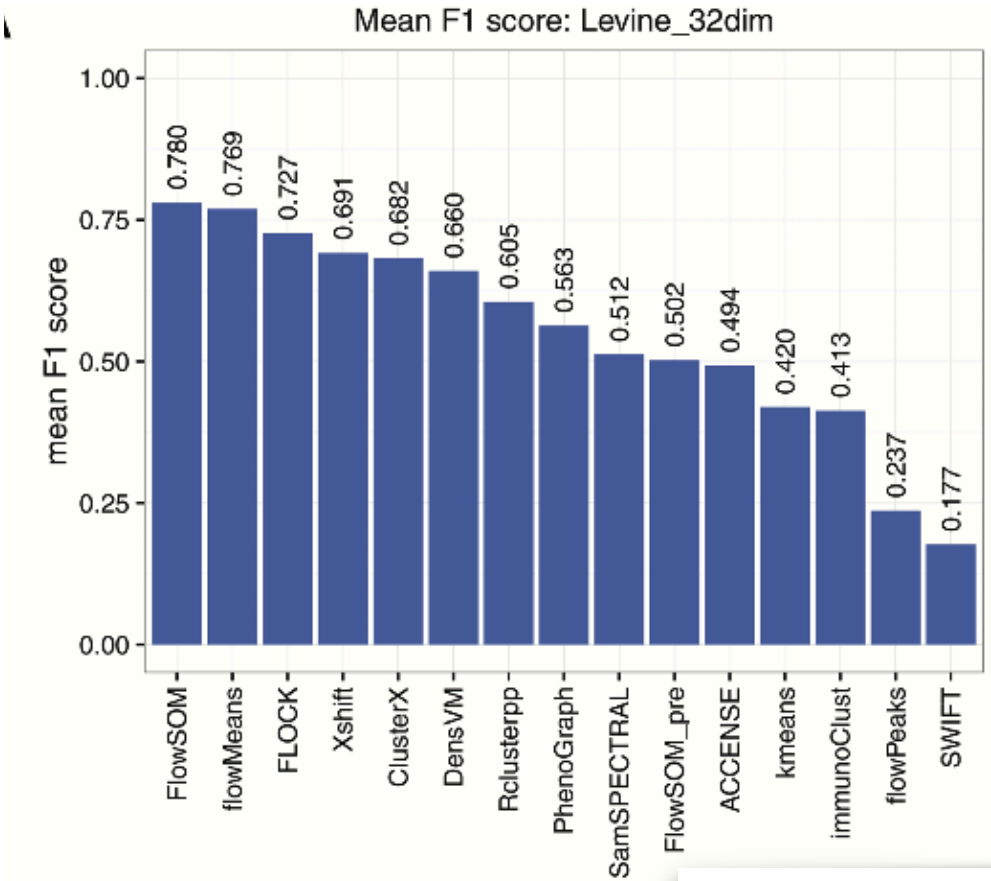
Manual gates = “truth”

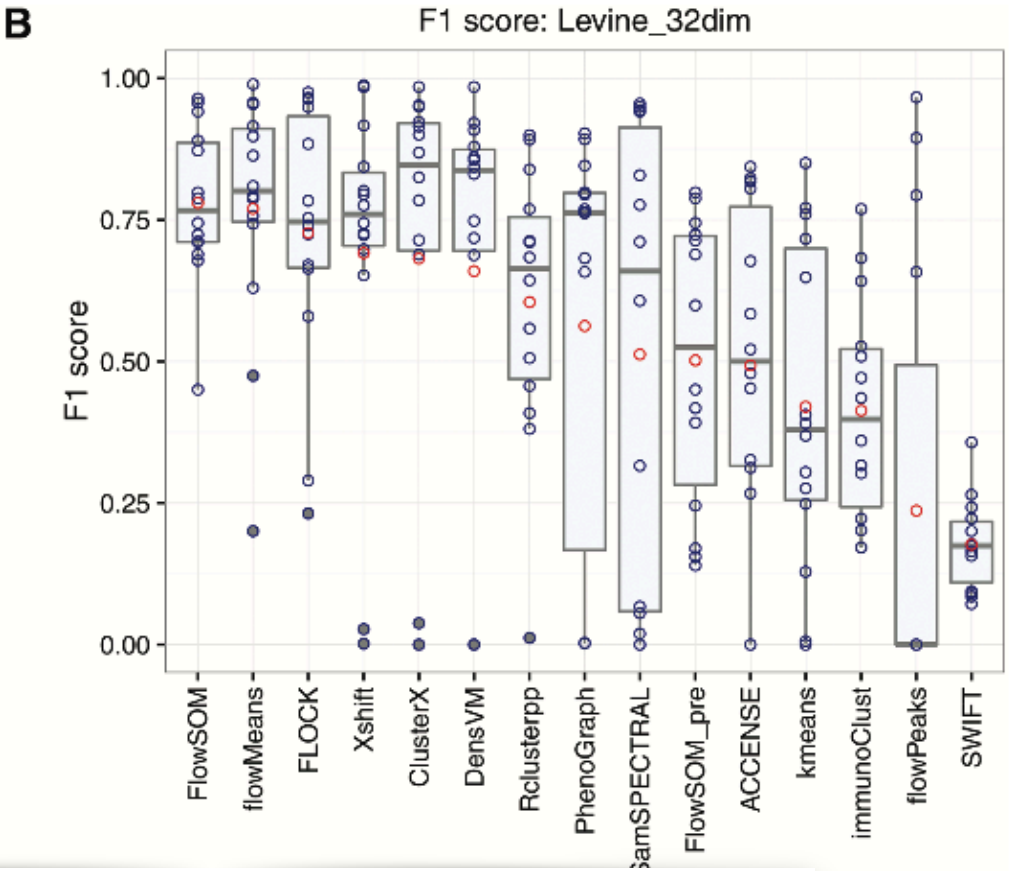
Table 2. Summary of data sets used to evaluate clustering methods

DATA SET	CYTOF OR FLOW CYTOMETRY	CLUSTERING TASK	NO. OF CELLS	NO. OF DIMENSIONS	NO. OF MANUALLY GATED POPULATIONS OF INTEREST	NO. OF MANUALLY GATED CELLS	ORGANISM	NO. OF INDIVIDUALS (PATIENTS, MICE)	SAMPLE DESCRIPTION	REF.
Levine_32dim	CyTOF	Multiple populations	265,627	32 (surface markers)	14	104,184 (39%)	Human	2	Bone marrow cells from healthy donors	(18)
Levine_13dim	CyTOF	Multiple populations	167,044	13 (surface markers)	24	81,747 (49%)	Human	1	Bone marrow cells from healthy donor	(18)
Samusik_01	CyTOF	Multiple populations	86,864	39 (surface markers)	24	53,173 (61%)	Mouse	1	Replicate bone marrow samples from C57BL/6J mice (sample 01 only)	(17)
Samusik_all	CyTOF	Multiple populations	841,644	39 (surface markers)	24	514,386 (61%)	Mouse	10	Replicate bone marrow samples from C57BL/6J mice (all samples)	(17)
Nilsson_rare	Flow cytometry	Rare population	44,140	13 (surface markers)	1 (hematopoietic stem cells)	358 (0.8%)	Human	1	Bone marrow cells from healthy donor	(36)
Mosmann_rare	Flow cytometry	Rare population	396,460	14 (surface and intracellular)	1 (activated memory CD4 T cells)	109 (0.03%)	Human	1	Peripheral blood cells from healthy donor, stimulated with influenza antigens	(35)

Table 2

Comparison of clustering methods





Hungarian algorithm to match clusters to populations

F1 score

From Wikipedia, the free encyclopedia

"F score" redirects here. For the significance test, see F-test.

In statistical analysis of binary classification, the **F_1 score** (also **F-score** or **F-measure**) is a measure of a test's accuracy. It considers both the precision *p* and the recall *r* of the test to compute the score: *p* is the number of correct positive results divided by the number of all positive results, and *r* is the number of correct positive results divided by the number of positive results that should have been returned. The F_1 score can be interpreted as a weighted average of the precision and recall, where an F_1 score reaches its best value at 1 and worst at 0.

The traditional F-measure or balanced F-score (**F_1 score**) is the harmonic mean of precision and recall — multiplying the constant of 2 scales the score to 1 when both recall and precision are 1:

$$F_1 = 2 \cdot \frac{1}{\frac{1}{\text{recall}} + \frac{1}{\text{precision}}} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}.$$

Figure 1

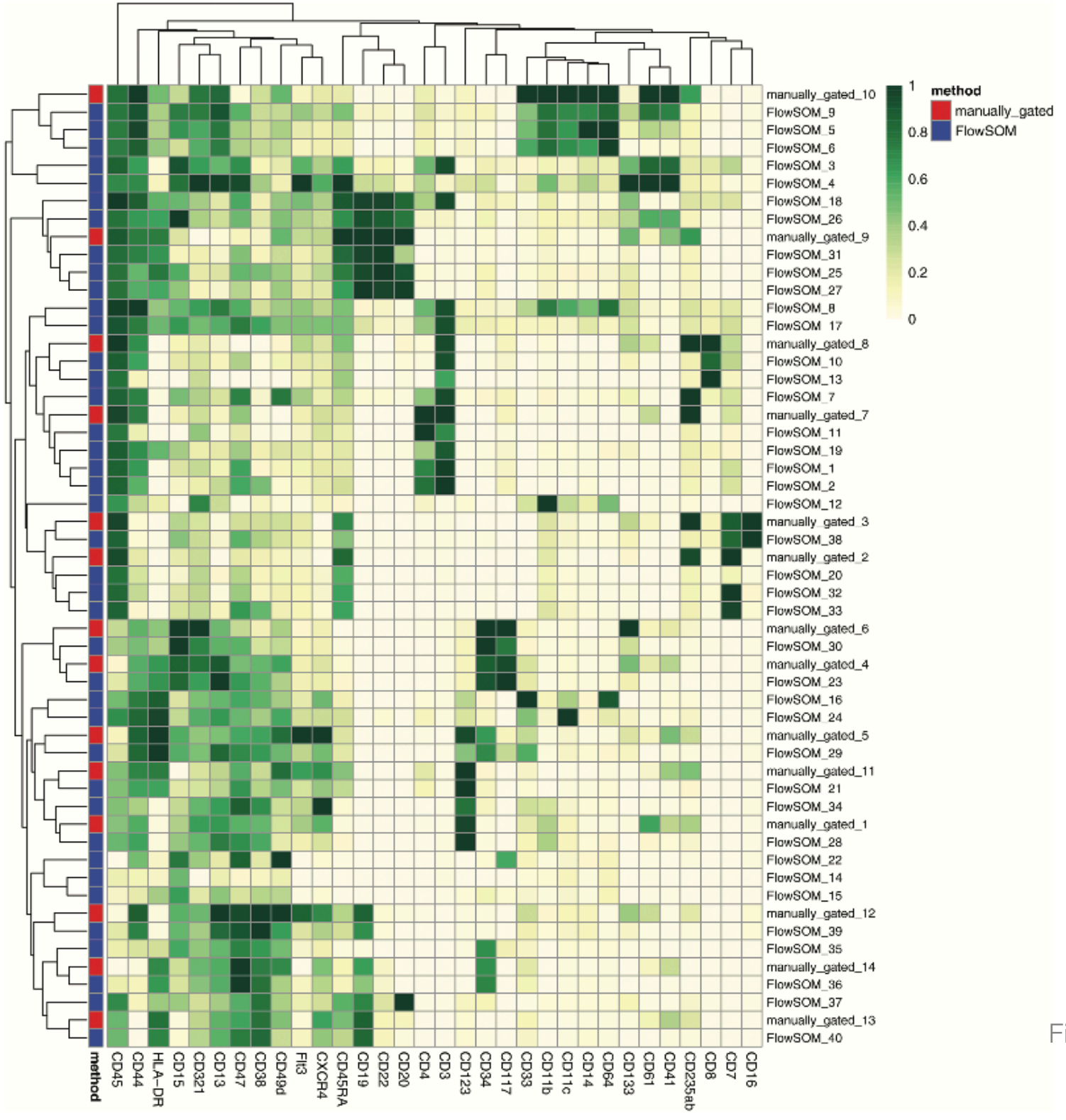
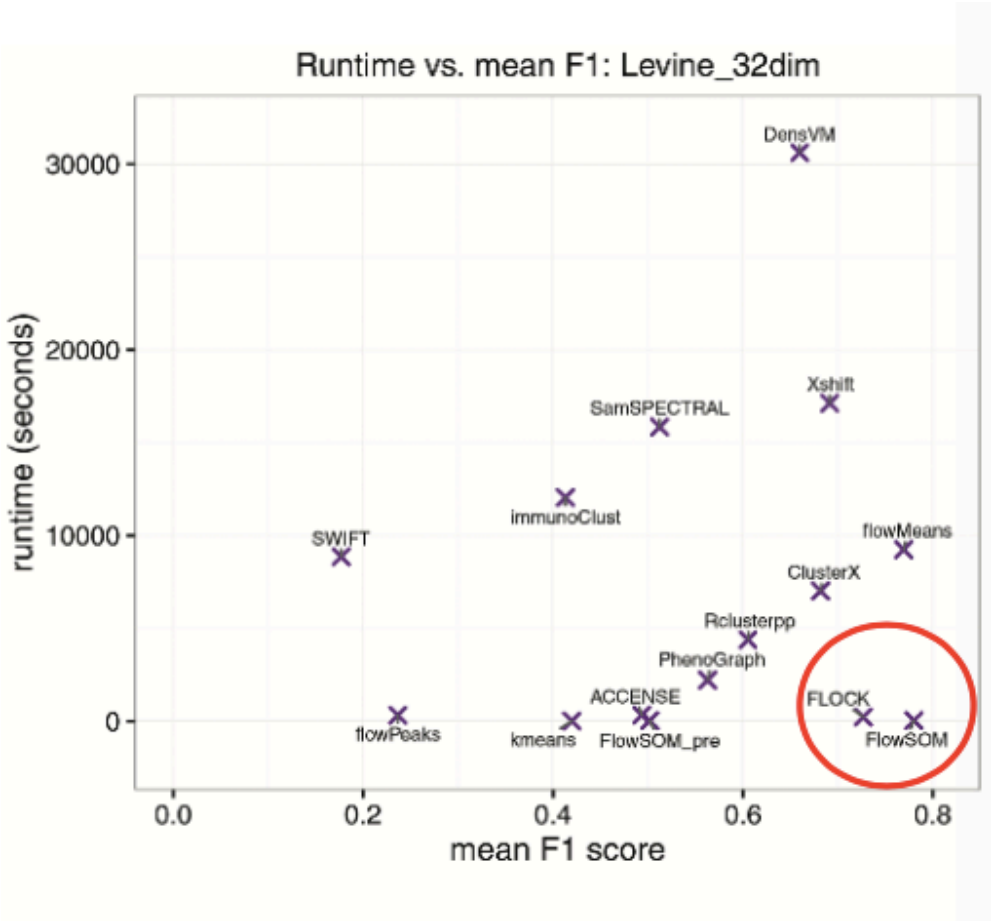

method
manually_gated
FlowSOM
1
0.8
0.6
0.4
0.2
0
manually_gated_10
FlowSOM_9
FlowSOM_5
FlowSOM_6
FlowSOM_3
FlowSOM_4
FlowSOM_18
FlowSOM_26
manually_gated_9
FlowSOM_31
FlowSOM_25
FlowSOM_27
FlowSOM_8
FlowSOM_17
manually_gated_8
FlowSOM_10
FlowSOM_13
FlowSOM_7
manually_gated_7
FlowSOM_11
FlowSOM_19
FlowSOM_1
FlowSOM_2
FlowSOM_12
manually_gated_3
FlowSOM_38
manually_gated_2
FlowSOM_20
FlowSOM_32
FlowSOM_33
manually_gated_6
FlowSOM_30
manually_gated_4
FlowSOM_23
FlowSOM_16
FlowSOM_24
manually_gated_5
FlowSOM_29
manually_gated_11
FlowSOM_21
FlowSOM_34
manually_gated_1
FlowSOM_28
FlowSOM_22
FlowSOM_14
FlowSOM_15
manually_gated_12
FlowSOM_39
FlowSOM_35
manually_gated_14
FlowSOM_36
FlowSOM_37
manually_gated_13
FlowSOM_40
method
CD45
CD44
HLA−DR
CD15
CD321
CD13
CD47
CD38
CD49d
Flt3
CXCR4
CD45RA
CD19
CD22
CD20
CD4
CD3
CD123
CD34
CD117
CD33
CD11b
CD11c
CD14
CD64
CD133
CD61
CD41
CD235ab
CD8
CD7
CD16

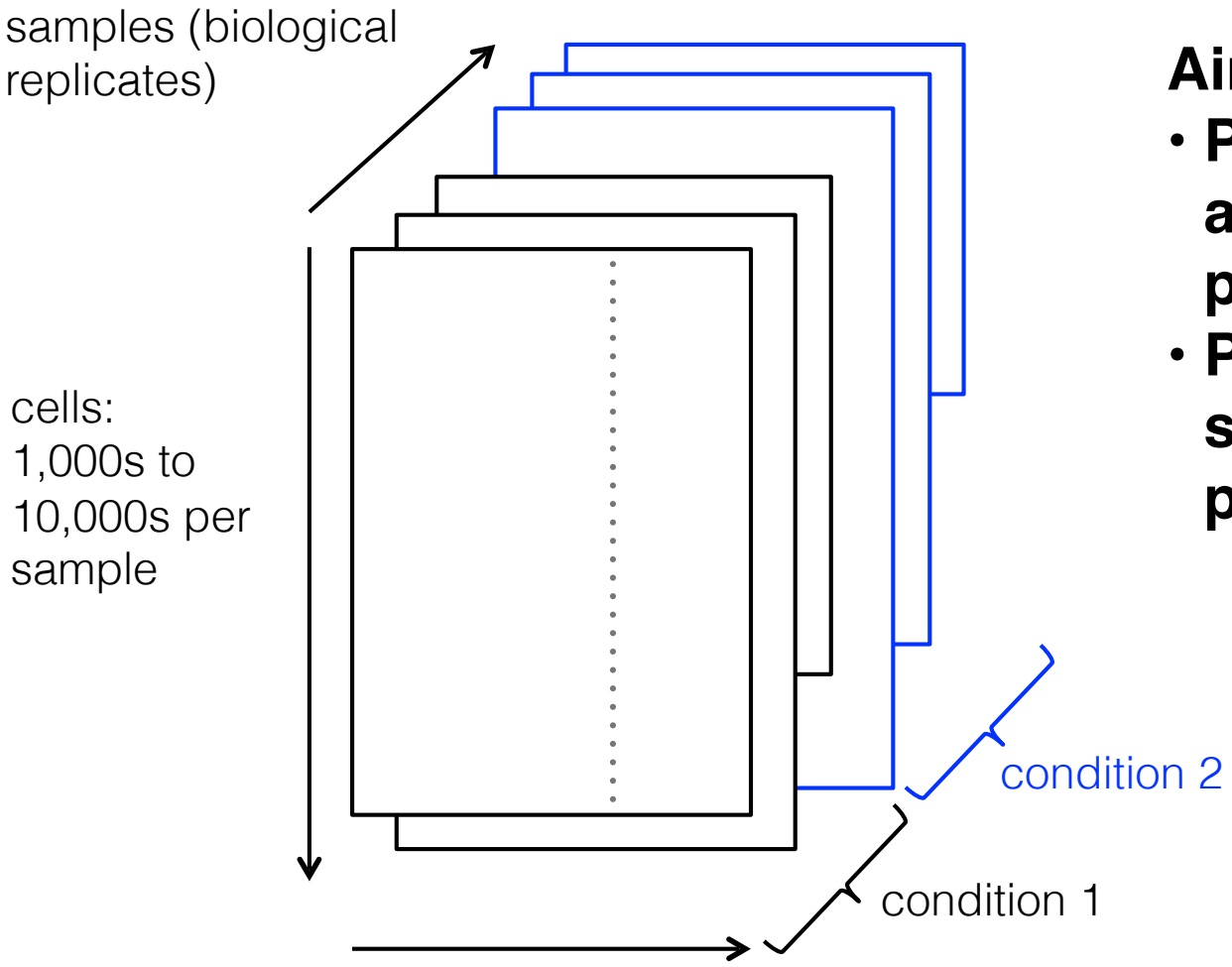
Figure 2

Comparison of clustering methods

- several methods performed well: *FlowSOM*, *X-shift*, *PhenoGraph*, *Rclusterpp*, *flowMeans*
- **FlowSOM** gave best performance (for several data sets) and was fast
- **X-shift** gave best performance for rare cell populations
- several methods sensitive to random starts (rare populations)
- code, data freely available



Differential analyses for HDCyto data



Aims (after clustering):

- **Part 1: differential abundance of populations (clusters)**
- **Part 2: differential marker signal within a population (cluster)**

Pipeline

Gosia Nowicka

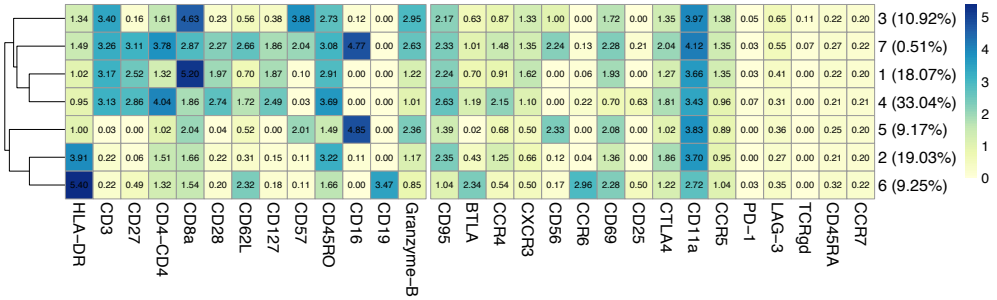
Main workflow:

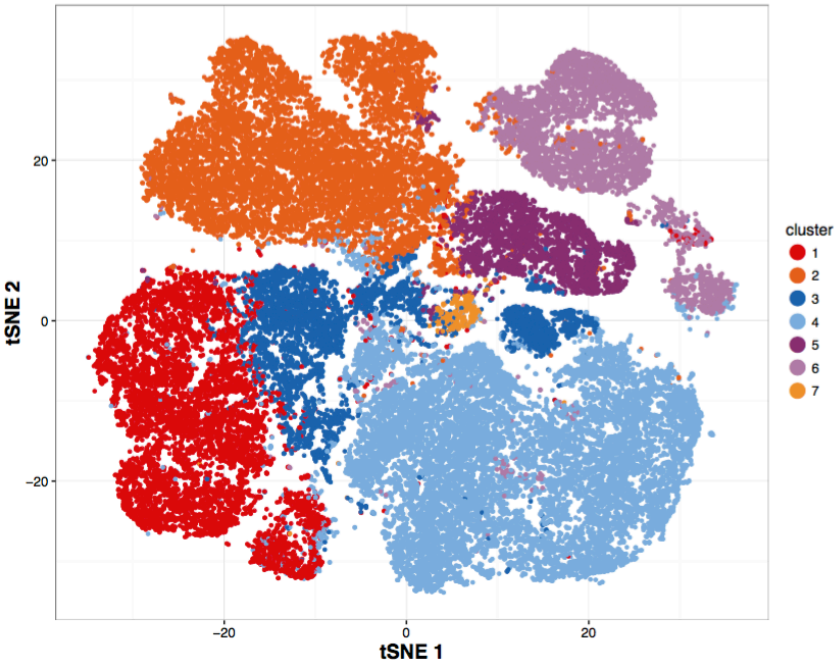
- Data normalization: (i) arcsineh, (ii) arcsineh + 01 on 1% and 99% quantiles shrunk to 0 or 1, respectively
- PCA scores on (i) data (select top varying observables)
- Run tSNE on (i) data
- Clustering into 20 groups on (i) data: FlowSOM + ClusterConsensus
- Heatmaps of 20 clusters (with all the obs.)
- Plot tSNE with 20 cluster heat
- Cluster merging done by the human expert – Carsten; or sometimes automatic
- Heatmaps of merged clusters(with all the obs.)
- Plot tSNE with merged clusters heat
- Frequency analysis
- Expression analysis

F1000/ Bioconductor Workflow article coming soon!

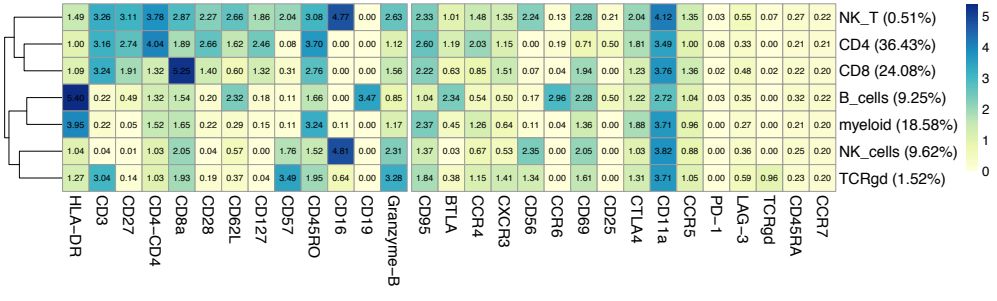
Merging clusters from 20 to 7

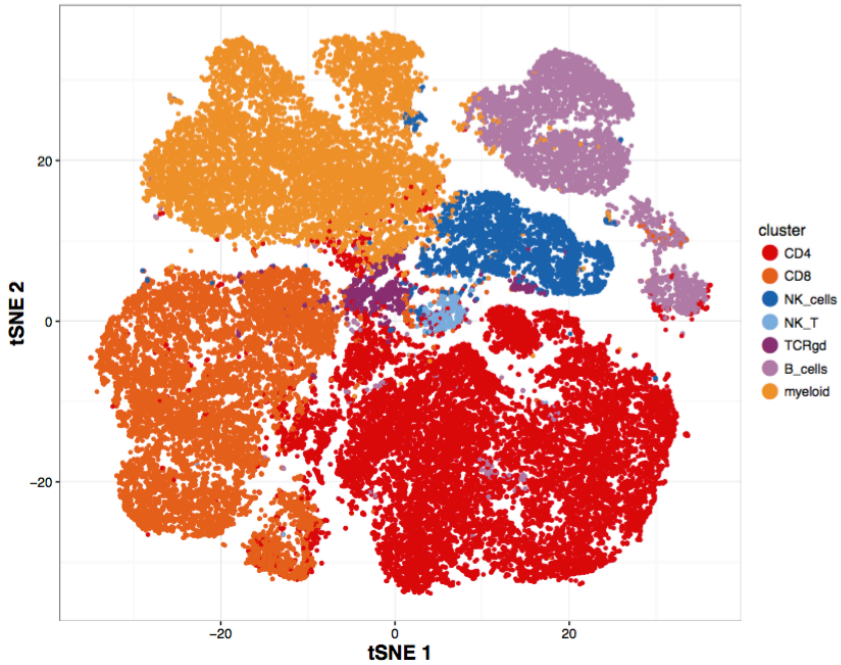
7 clusters





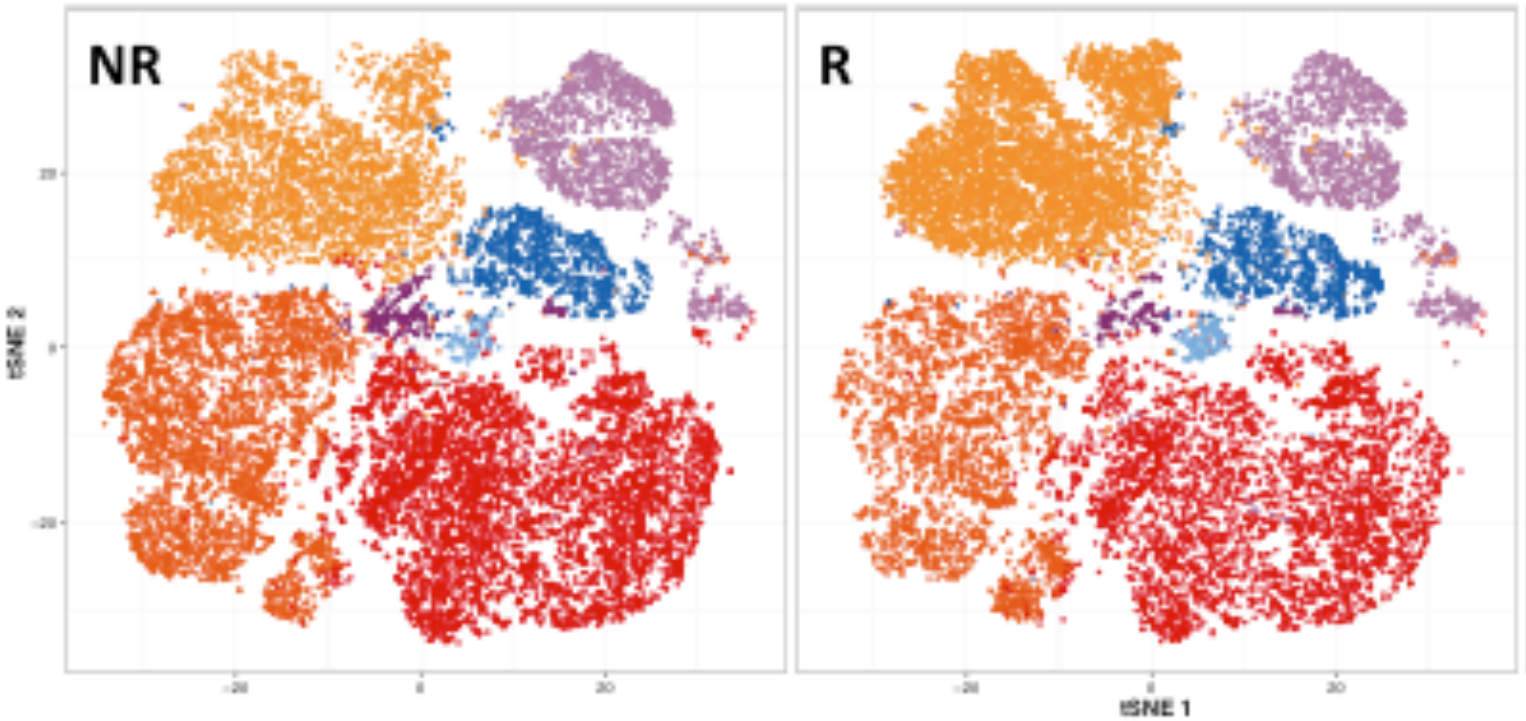
7 clusters by expert





data from Carsten Krieg, UZH
(PBMCs from metastatic melanoma patients, comparing **responders** to **non-responders**)

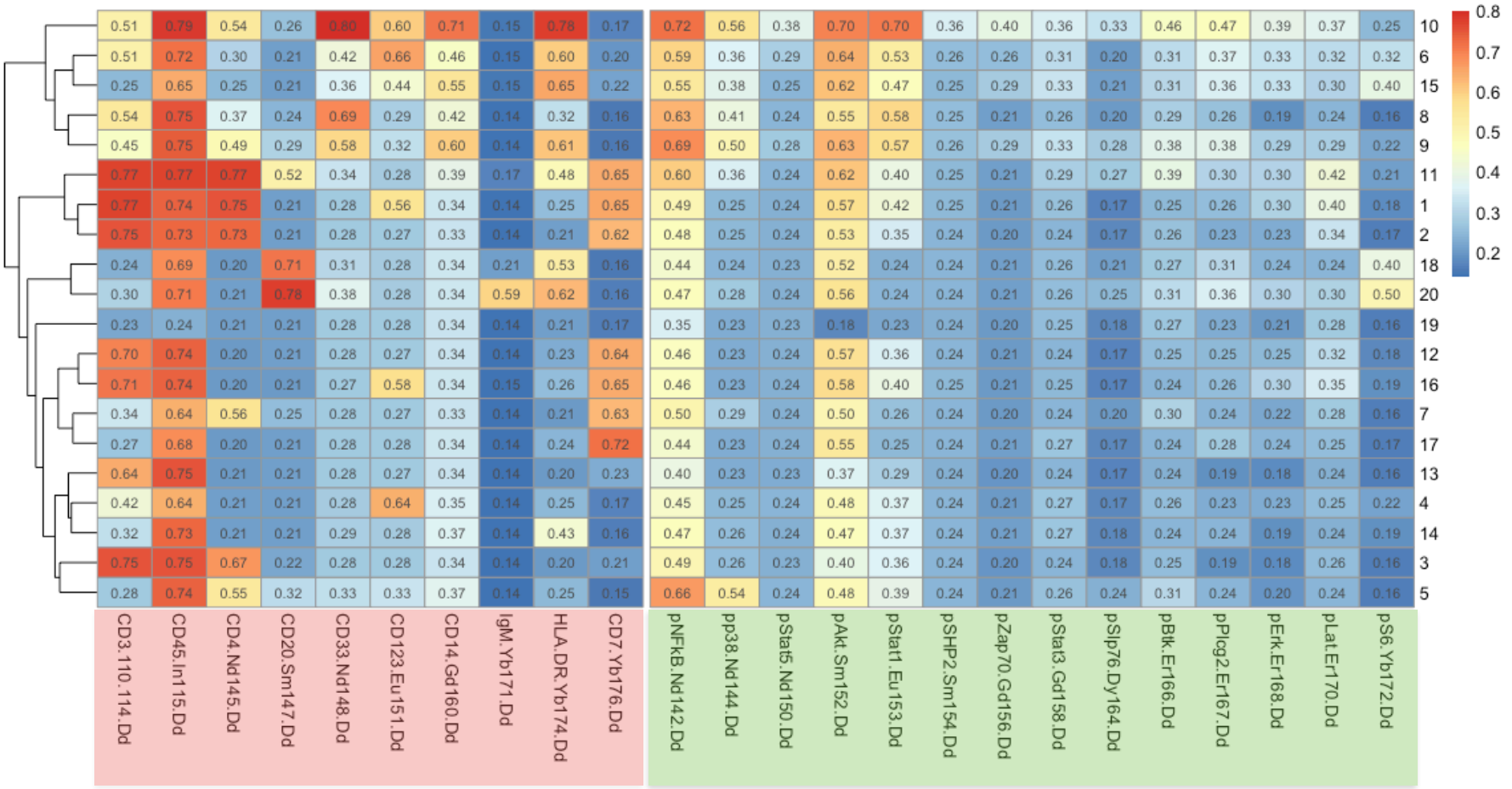
Part 1: Differential abundance of cell populations



NR: non-responders
R: responders

data from Carsten Krieg, UZH

Part 2:
Differential signalling for a given population

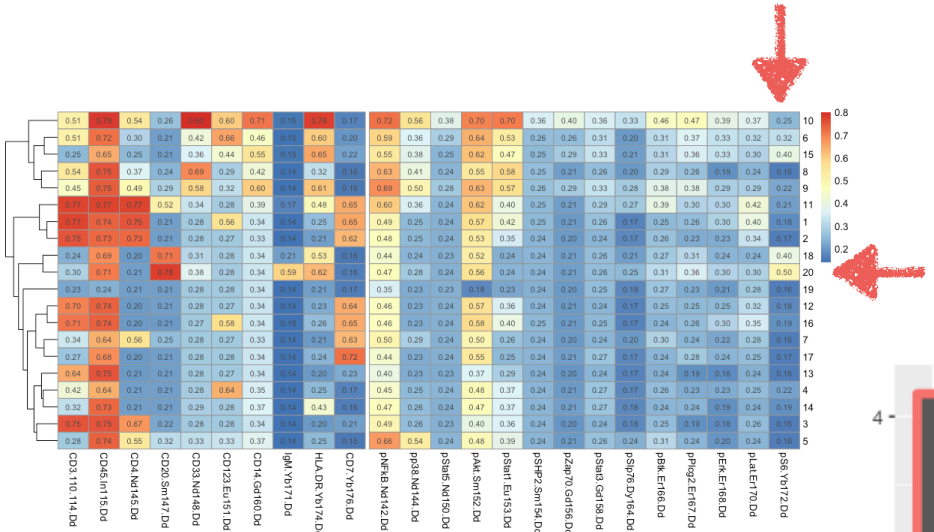


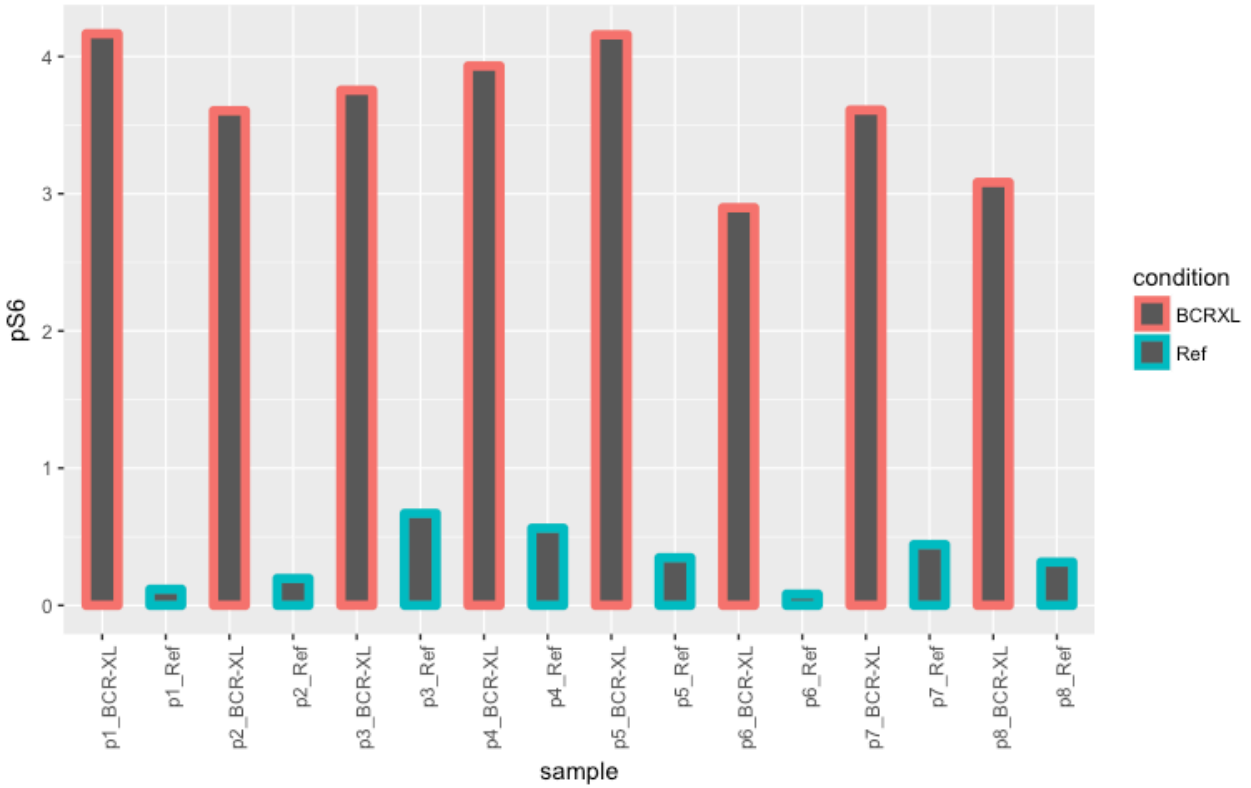
Used for clustering

Not used for clustering

data from Bodenmiller et al. 2012

Part 2:
Differential signalling for a given population (medians)

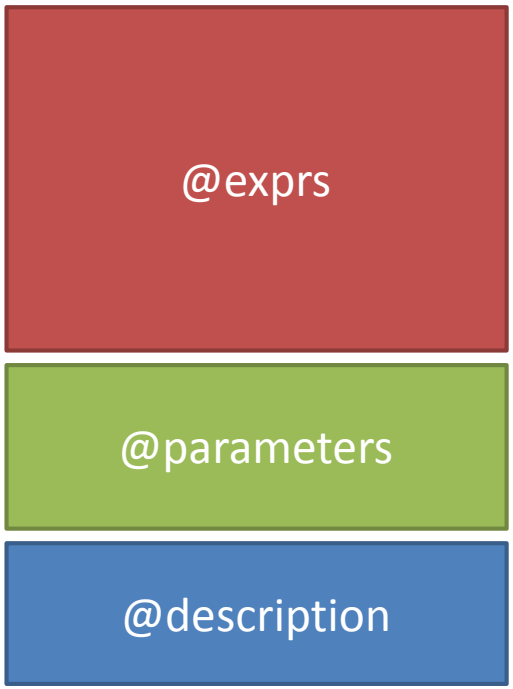




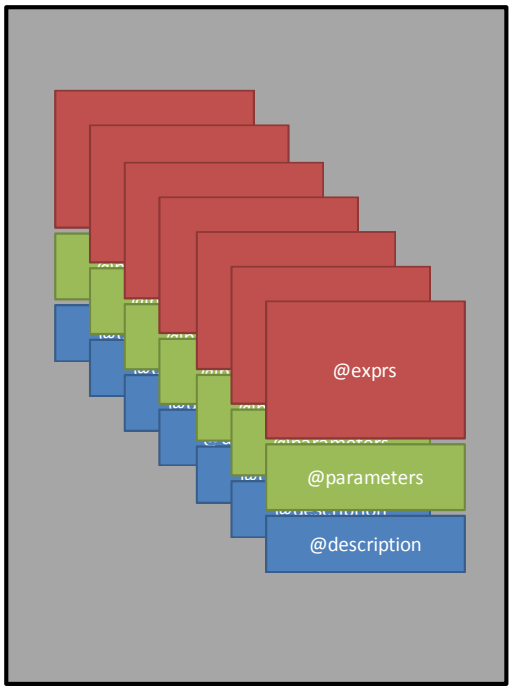
data from Bodenmiller et al. 2012

Cytometry data in R

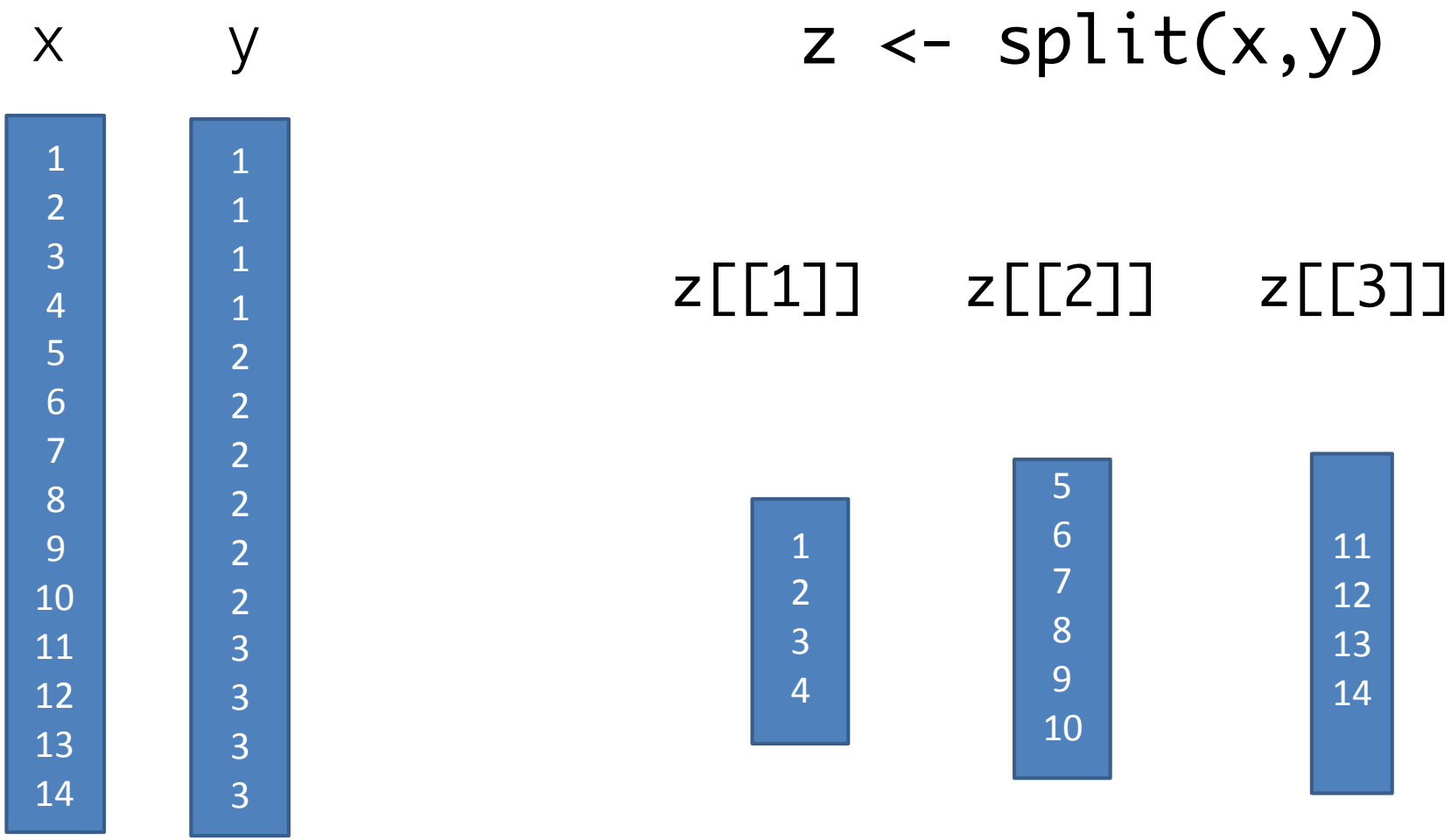
one flowframe = one fcs file



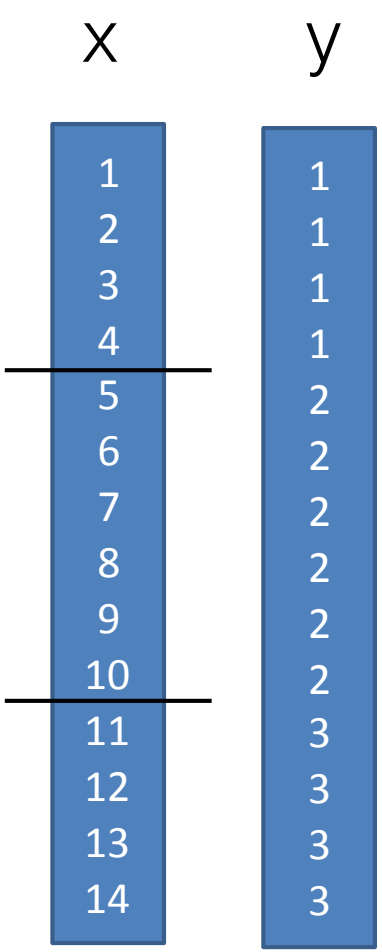
one flowSet = multiple fcs files



Lists, splitting



Lists, split

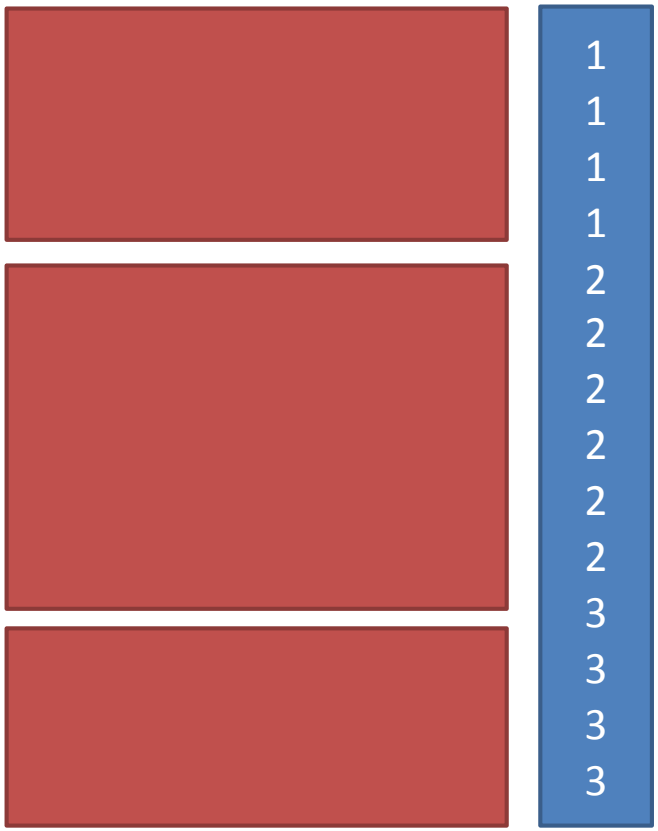



```
> x
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14
> y
 [1] 1 1 1 1 2 2 2 2 2 2 3 3 3 3
> names(s)
[1] "1" "2" "3"
> s <- split(x,y)
> s
$`1`
[1] 1 2 3 4

$`2`
[1]  5  6  7  8  9 10

$`3`
[1] 11 12 13 14

> names(s)
[1] "1" "2" "3"
```

`lapply`, `sapply`, etc.: apply a function to every element of list

```
> s
$`1`
[1] 1 2 3 4

$`2`
[1]  5  6  7  8  9 10

$`3`
[1] 11 12 13 14

> sapply(s, length)
1 2 3
4 6 4
> lapply(s, sum)
$`1`
[1] 10

$`2`
[1] 45

$`3`
[1] 50
```

Aggregation operations (`summarize_each`)



Apply a function (e.g., calculate average) to each column of a big matrix according to a vector specifying the "group"

Concept of "tidy" data

acast, dcast

```
> cell_counts
Source: local data frame [320 x 3]
Groups: samp [?]

         samp clust     n
       <fctr> <int> <int>
1  p1_BCR-XL      1    31
2  p1_BCR-XL      2   624
3  p1_BCR-XL      3    85
4  p1_BCR-XL      4    14
5  p1_BCR-XL      5     2
6  p1_BCR-XL      6    21
7  p1_BCR-XL      7    38
8  p1_BCR-XL      8     5
9  p1_BCR-XL      9    42
10 p1_BCR-XL     10    11
# ... with 310 more rows
```

```
> head(cell_counts_a)
  p1_BCR-XL p1_Ref p2_BCR-XL p2_Ref p3_BCR-XL p3_Ref p4_BCR-XL
1        31     19       290    140       180     75       182
2       624    754      4309   4434      2642   1679      1935
3        85    326      1133   2575       967   1275       602
4        14     11       312    139       321    119       251
5         2     75        33    635        80    635        77
6        21      9       292    162       283    142       289
..
```

melt