

KSBi-BIML 2021

Bioinformatics & Machine Learning (BIML)
Workshop for Life Scientists

생물정보학 & 머신러닝 워크샵(온라인)

Single-cell RNA-sequencing Analysis of Cancer

이세민

Bioinformatics & Machine Learning for Life Scientists

BIML-2021

안녕하십니까?

한국생명정보학회의 동계 워크샵인 BIML-2021을 2월 15부터 2월 19일까지 개최합니다. 생명정보학 분야의 융합이론 보급과 실무역량 강화를 위해 도입한 전문 교육 프로그램인 BIML 워크샵은 2015년에 시작하였으며 올해로 7차를 맞이하게 되었습니다. 유례가 없는 코로나 대유행으로 인해 올해의 BIML 워크숍은 온라인으로 준비했습니다. 생생한 현장 강의에서만 느낄 수 있는 강의자와 수강생 사이의 상호교감을 가질 수 없다는 단점이 있지만, 온라인 강의의 여러 장점을 살려서 최근 생명정보학에서 주목받고 있는 거의 모든 분야를 망라한 강의를 준비했습니다. 또한 온라인 강의의 한계를 극복하기 위해서 실시간 Q&A 세션 또한 마련했습니다.

BIML 워크샵은 전통적으로 크게 생명정보학과 AI, 두 개의 분야로 구성되어오고 있으며 올해 역시 유사한 방식을 채택했습니다. AI 분야는 Probabilistic Modeling, Dimensionality Reduction, SVM 등과 같은 전통적인 Machine Learning부터 Deep Learning을 이용한 신약개발 및 유전체 연구까지 다양한 내용을 다루고 있습니다. 생명정보학 분야로는, Proteomics, Chemoinformatics, Single Cell Genomics, Cancer Genomics, Network Biology, 3D Epigenomics, RNA Biology, Microbiome 등 거의 모든 분야가 포함되어 있습니다. 연사들은 각 분야 최고의 전문가들이라 자부합니다.

이번 BIML-2021을 준비하기까지 너무나 많은 수고를 해주신 BIML-2021 운영위원회의 김태민 교수님, 류성호 교수님, 남진우 교수님, 백대현 교수님께 커다란 감사를 드립니다. 또한 재정적 도움을 주신, 김선 교수님 (AI-based Drug Discovery), 류성호 교수님, 남진우 교수님께 감사를 표시하고 싶습니다. 마지막으로 부족한 시간에도 불구하고 강의 부탁을 흔쾌히 허락하시고 훌륭한 강의자료를 만드는데 노력하셨을 뿐만 아니라 실시간 온라인 Q&A 세션까지 참여해 수고해 주시는 모든 연사분들께 깊이 감사드립니다.

2021년 2월

한국생명정보학회 회장 김동섭

강의개요

Single-cell RNA-sequencing analysis of cancer

본 강좌에서는 최근 각광 받고 있는 단세포 전사체 데이터 분석 기술에 대한 소개와 실제 데이터에 대한 분석 실습을 병행하고자 한다. 단세포 전사체 분석 기술은 세포의 분화, 암의 진화, 면역세포 프로파일링 및 종양 내 이질성 분석 등에 활용되고 있으며, 관련 기술과 응용 사례에 대한 소개 및 현재 가장 널리 사용되고 있는 10x Genomics사의 Chromium Single Cell Gene Expression Solution을 사용하여 생산된 암샘플 단세포 전사체 데이터를 위주로 다양한 분석 방법에 대한 실습을 진행하고자 한다. 강의는 다음의 내용을 포함한다.

- Single-cell RNA-sequencing(scRNA-seq)의 소개 및 개요
- 암 scRNA-seq 연구 동향
- 암 scRNA-seq 데이터 분석 실습

*참고강의교재:

없음

*교육생준비물:

노트북 (메모리 8GB 이상, 디스크 여유공간 30GB 이상)

* 강의: 이세민 교수 (울산과학기술원 바이오메디컬공학과)

Curriculum Vitae

Speaker Name: Semin Lee, Ph.D.



► Personal Info

Name Semin Lee
Title Associate Professor
Affiliation Ulsan National Institute of Science and Technology

► Contact Information

Address UNIST-gil 50, Bldg #110, Room #301-7, Ulsan, 44919
Email seminlee@unist.ac.kr
Phone Number 010-3166-4751

Research interest : Cancer genomics & single-cell genomics

Educational Experience

2003 B.S. in Biological Sciences, Seoul National University, Korea
2004 M.S. in Bioinformatics, KAIST, Korea
2007 Ph.D. in Bioinformatics, University of Cambridge, UK

Professional Experience

2011-2016 Research Fellow, Department of Biomedical Informatics, Harvard Medical School
USA
2016- Associate Professor, Department of Biomedical Engineering, UNIST, Korea

Selected Publications (5 maximum)

1. Ji Hoon Phi, Ae Kyung Park, Semin Lee, et al. Genomic analysis reveals secondary glioblastoma after radiotherapy in a subset of recurrent medulloblastomas, *Acta Neuropathologica*, 2018, Jun;135(6):939-953.
2. Cancer Genome Atlas Research Network. Integrated genomic and molecular characterization of cervical cancer. *Nature*. 2017 Jan 23.
3. Xi R, Lee S, Xia Y, Kim TM, Park PJ. Copy number analysis of whole-genome data using BIC-seq2 and its application to detection of cancer susceptibility variants. *Nucleic Acids Res*. 2016 Jul 27;44(13):6274-86.
4. Lodato MA, Woodworth MB, Lee S, Evrony GD, Mehta BK, Karger A, Lee S, Chittenden TW, Cai X, Luquette LJ, Lee E, Park PJ, Walsh CA. Mosaic Mutations Trace Developmental and Transcriptional Histories of Single Human Neurons. *Science*. 2015 Oct 2;350(6256):94-8.
5. Cancer Genome Atlas Research Network. Comprehensive molecular characterization of human colon and rectal cancer. *Nature*. 2012 Jul 18;487(7407):330-7.

KSBi-BIML 2021

Single-cell RNA-sequencing analysis of cancer

Semin Lee, PhD

Email : seminlee@unist.ac.kr

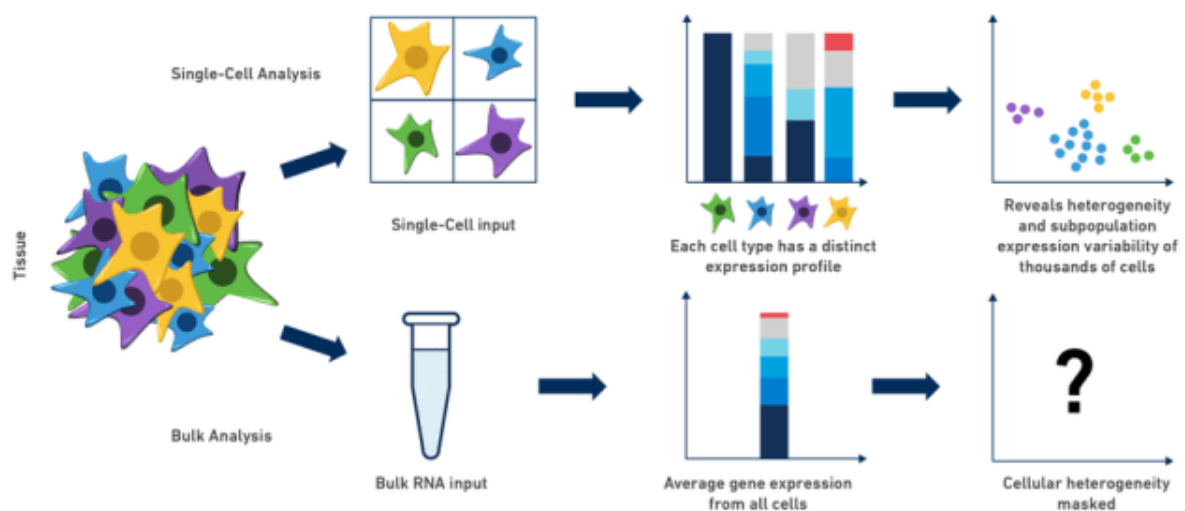
Ulsan National Institute of Science and Technology

본 강의 자료는 한국생명정보학회가 주관하는 KSBi-BIML 2021 워크숍 온라인 수업을 목적으로 제작된것으로 해당 목적 이외의 다른 용도로 사용할 수 없음을 분명하게 알립니다. 수업 목적으로 배포 및 전송 받은 경우에도 이를 다른 사람과 공유하거나 복제, 배포, 전송할 수 없습니다.

만약 이러한 사항을 위반할 경우 발생하는 모든 법적 책임은 전적으로 불법 행위자 본인에게 있음을 경고합니다.

Introduction

Why single-cell RNA sequencing?



<https://www.10xgenomics.com/blog/single-cell-rna-seq-an-introductory-overview-and-tools-for-getting-started>

Limitations of bulk RNA sequencing

- Bulk sequencing methods are limited to reporting an **average signal** from a complex population of cells.
- So, it is difficult to resolve **cell-to-cell variations** in tumor and identify the complex nature of **tumor microenvironment** (TME).
- Computational deconvolution techniques could help infer the cellular composition of tumors, but such analyses are **limited to a few known cell types**.

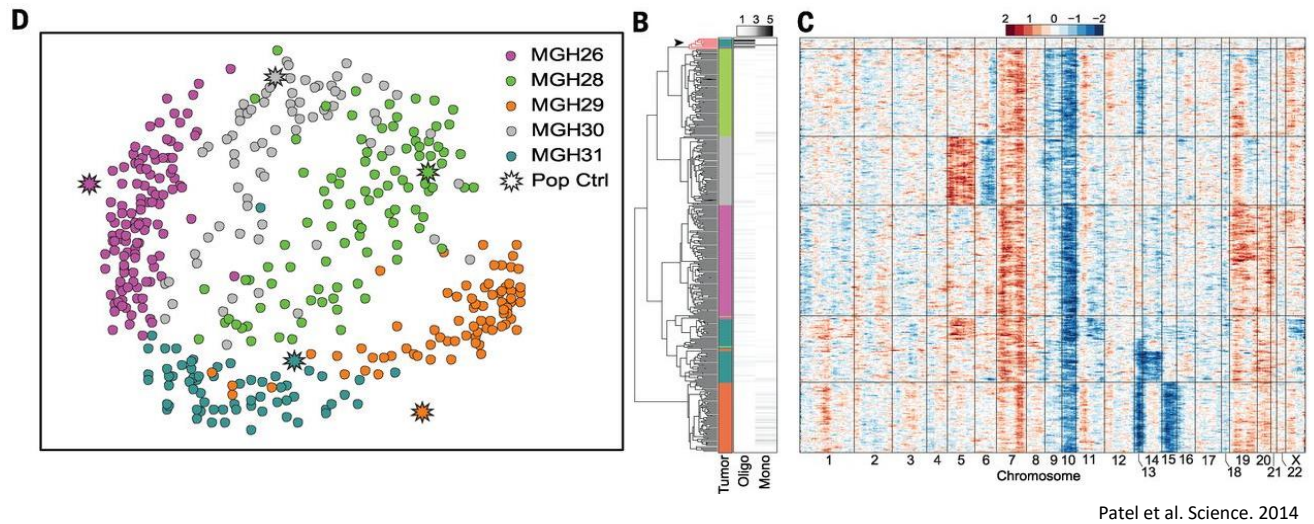
Advantages of single-cell RNA sequencing

- Single-cell **RNA** sequencing
 - not only identifies intratumoral heterogeneity but also reconstruct a **high-resolution map** of the TME.
 - identifies **cell-specific genetic variants** and reconstruct tumor clonality and evolution.
- Single-cell sequencing does not rely on known cell type-specific gene signatures or surface markers.

- Tumor heterogeneity
- Clonal evolution of cancer
- Circulating tumor cells
- Tumor microenvironment



Tumor heterogeneity

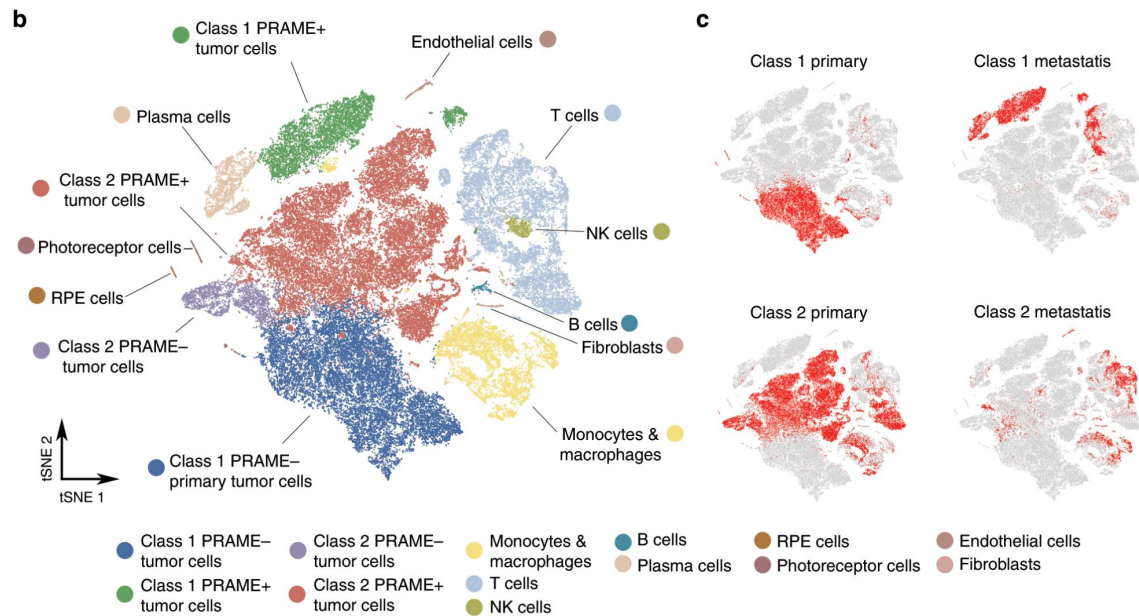


Circulating tumor cells

Cancer type	CTC enrichment	CTC criteria (micromanipulation)	Single-cell profiling	Number of CTCs (number of patients) ^a
Multiple myeloma	FACS with serial dilution	CD45 ⁻ , CD138 ⁺	SMART-seq2	21 (2)
Colon	CellSearch®	CD45 ⁺ , EpCAM ⁺	Multiplex PCR	11 (8)
Ovary	Biocoll separation, Dynabeads® CD45 depletion	DAPI ⁺ , CK/EpCAM ⁺ , CD45 ⁻	Multiplex PCR	15 (3)
Breast	MagSweeper®	EpCAM ⁺	Microfluidic RT-PCR ^b	105 (50)
	Microfluidic ^{neq} CTC-iChip	EpCAM/HER2/CDH11 ⁺ , CD45/CD16/CD14 ⁻	Optimized Tang's method	15 (10)
	Microfluidic CTC-iChip	EpCAM/HER2/EGFR ⁺ , CD45 ⁻	SMART-seq v4 ^c	15 (10)
	Microfluidic ClearCell® FX	CD45/CD31 ⁻ , Calcein ⁺ ^d	Polaris™ IFC	68 (4)
Melanoma	MagSweeper®	CD45 ⁻ , Calcein ⁺	SMART-seq	6 (1)
Prostate	MagSweeper®	CD45 ⁻ , EpCAM ⁺ , DAPI ⁻	SMART-seq, Advantage 2 PCR (Clontech)	20 (4)
	ScreenCell®	CD45 ⁻	Microfluidic RT-PCR ^e	38 (9)
	Microfluidic CTC-iChip	CD45 ⁻ , EpCAM/CDH11 ⁺	Modified Tang's method	77 (13)
Lung	Integrated nanoplatform	EpCAM ⁺	Multiplex PCR	8 (1), 18 (1), 74 (1)
	Microfluidic ClearCell® FX	CD45 ⁻ ^f	Multiplex PCR	61 (20)
Prostate, breast	CellSearch®, Parsortix™	EpCAM/pan-keratins ⁺	Multiplex PCR	13 (1), 8 (1)
Pancreas, breast, prostate	Microfluidic CTC-iChip	CD45 ⁻	Modified Tang's method	7 (-), 29 (-), 77 (-)

Lim et al. NPJ Precis Oncol. 2019

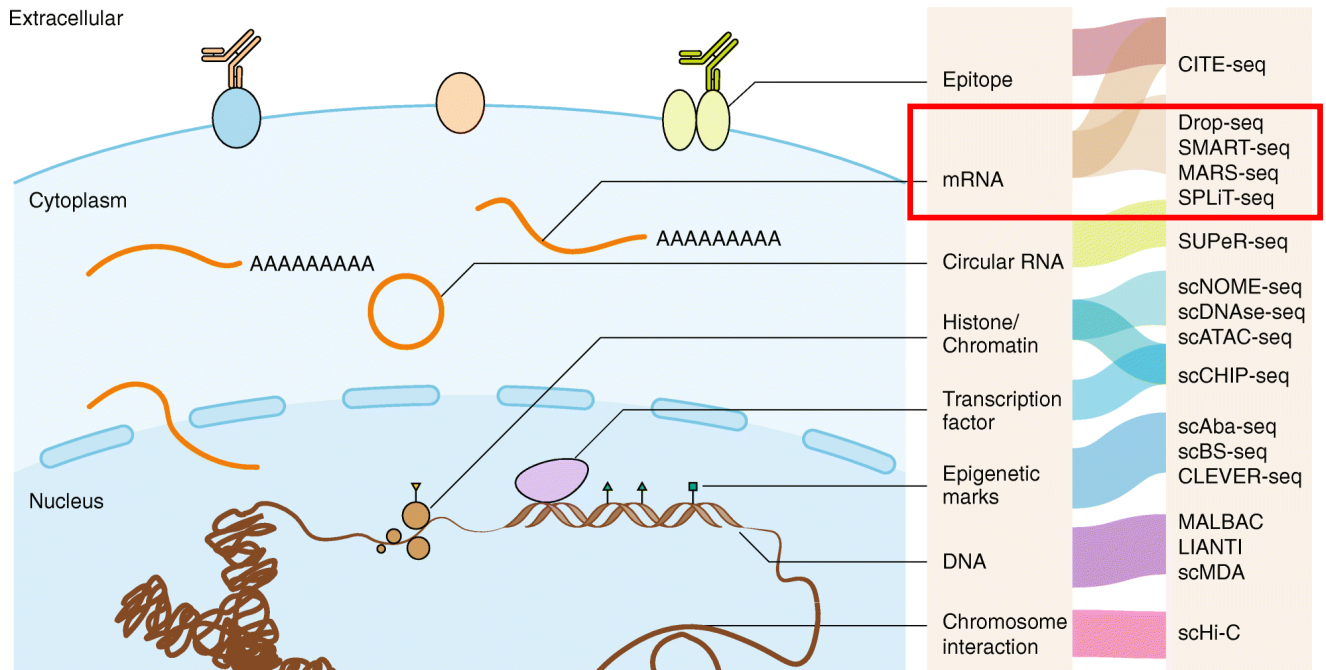
Tumor microenvironment



Durante et al. Nat Commun. 2020

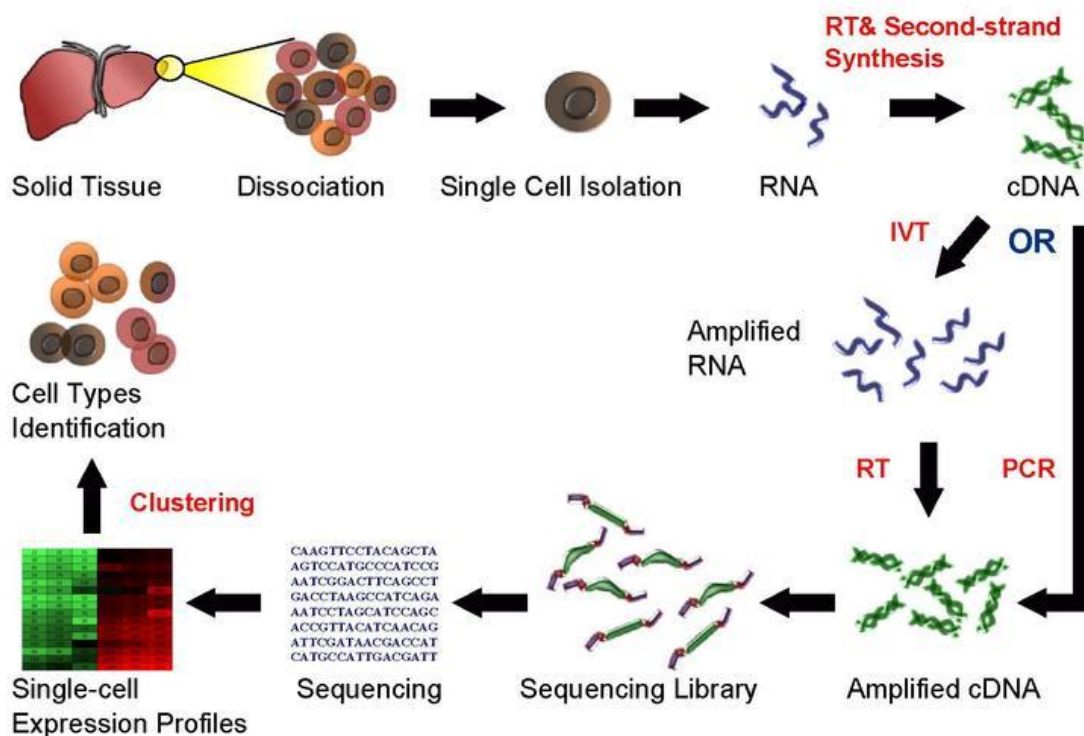
Single-cell sequencing

State of the art of single-cell sequencing technologies

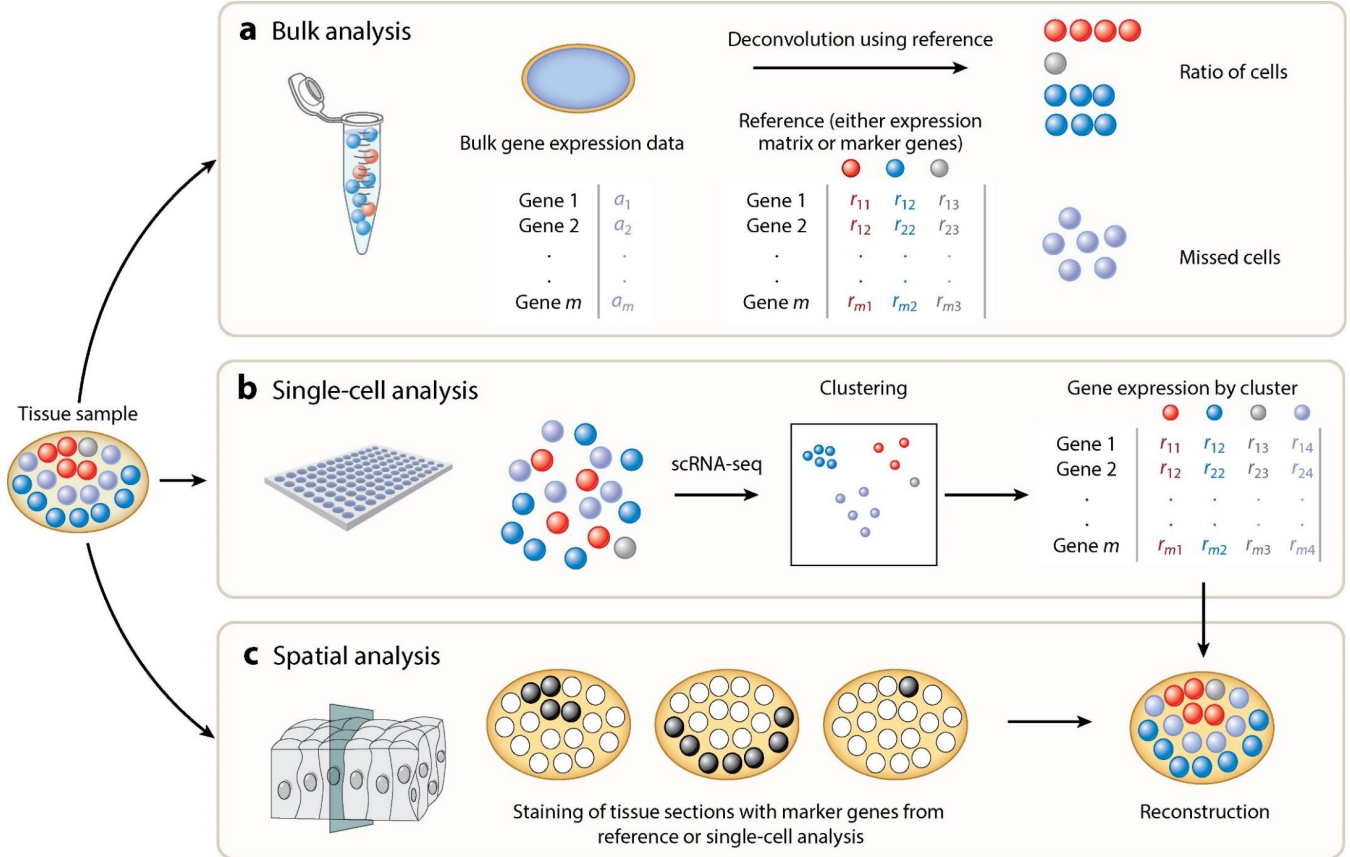


Ren et al. Genome Biol. 2018

Single Cell RNA Sequencing Workflow



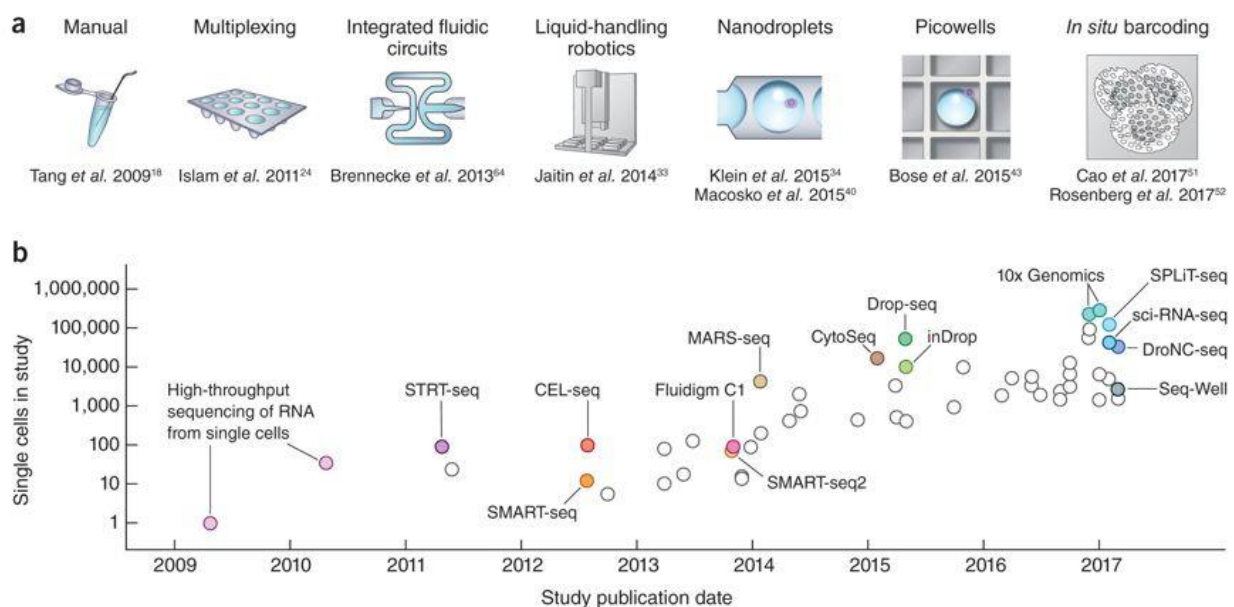
https://en.wikipedia.org/wiki/Single_cell_sequencing



Chen X, et al. 2018.
Annu. Rev. Biomed. Data Sci. 1:29–51

Chen et al. Annual review of biomedical data science. 2018

Scaling of scRNA-seq experiments



Svensson et al. Nature Protocols. 2018

REPORT

Defining the earliest step of cardiovascular lineage segregation by single-cell RNA-seq

Fabienne Lescroart^{1,*}, Xiaonan Wang^{2,3,*}, Xionghui Lin^{1,*}, Benjamin Swedlund¹, Souhir Gargouri¹, Adriana Sánchez-Dânes¹, ...

★ See all authors and affiliations

Science 09 Mar 2018:
Vol. 359, Issue 6380, pp. 1177-1181
DOI: 10.1126/science.aao4174

REPORT

Single-cell multiomics sequencing and analyses of human colorectal cancer

Shuhui Bian^{1,2,3,*}, Yu Hou^{1,2,*}, Xin Zhou^{4,*}, Xianlong Li^{1,2,*}, Jun Yong^{1,5,*}, Yicheng Wang^{1,2,*}, Wendong Wang⁴, Jia Yan^{1,2}, Bo...

★ See all authors and affiliations

Science 30 Nov 2018:
Vol. 362, Issue 6418, pp. 1060-1063
DOI: 10.1126/science.aao3791

RESEARCH ARTICLE

Decoupling genetics, lineages, and microenvironment in IDH-mutant gliomas by single-cell RNA-seq

Andrew S. Venteicher^{1,2,3,*}, Itay Tirosh^{2,*,†}, Christine Hebert^{1,2}, Keren Yizhak^{1,2}, Cyril Neftel^{1,2,4}, Mariella G. Filbin^{1,2,5}, Volk...

★ See all authors and affiliations

Science 31 Mar 2017:
Vol. 355, Issue 6332, eaa18478
DOI: 10.1126/science.aal8478

Cell Stem Cell

Volume 17, Issue 3, 3 September 2015, Pages 360-372

Resource

Single-Cell RNA-Seq with Waterfall Reveals Molecular Cascades underlying Adult Neurogenesis

Jaehoon Shin^{1,2}, Daniel A. Berg^{2,3,7}, Yunhua Zhu^{2,3}, Joseph Y. Shin⁴, Juan Song^{2,3}, Michael A. Bonaguidi^{2,3}, Grigori Enikolopov^{4,9}, David W. Nauen⁵, Kimberly M. Christian^{2,3}, Guo-li Ming^{2,3,4,6}, Hongjun Song^{1,2,3,4}

✉

CellPress

**nature
genetics**

Article | Published: 11 March 2019

Interrogation of human hematopoiesis at single-cell and single-variant resolution

Jacob C. Ullrich, Caleb A. Lareau, Erik L. Bao, Lief S. Ludwig, Michael H. Guo, Christian Benner, Ansuman T. Satpathy, Vinay K. Kartha, Rany M. Salem, Joel N. Hirschhorn, Hilary K. Finucane, Martin J. Aryee, Jason D. Buenrostro & Vijay G. Sankaran

Nature Genetics 51, 683–693 (2019) | Download Citation

nature
International Journal of Science

Letter | Published: 14 March 2018

A single-cell RNA-seq survey of the developmental landscape of the human prefrontal cortex

Suijuan Zhong, Shu Zhang, Xiaoying Fan, Qian Wu, Liying Yan, Ji Dong, Haofeng Zhang, Long Li, Le Sun, Na Pan, Xiaohui Xu, Fuchou Tang, Jun Zhang, Jie Qiao & Xiaoqun Wang

Nature 555, 524–528 (22 March 2018) | Download Citation

**nature
medicine**

Letter | Published: 25 June 2018

Single-cell profiling of breast cancer T cells reveals a tissue-resident memory subset associated with improved prognosis

Peter Savas, Balaji Virassamy, [...] Sherene Loi

**nature
immunology**

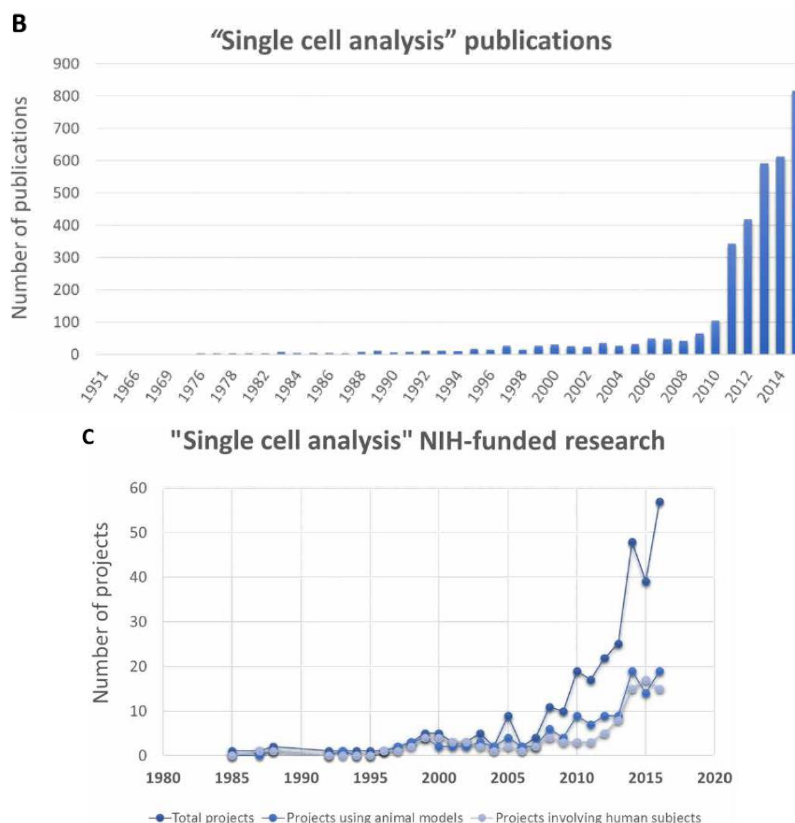
Resource | Published: 15 February 2016

The heterogeneity of human CD127⁺ innate lymphoid cells revealed by single-cell RNA sequencing

Åsa K Björklund, Marianne Forkel, Simone Picelli, Viktoria Konya, Jakob Theorell, Danielle Friberg, Rickard Sandberg & Jenny Mjösberg

Nature Immunology 17, 451–460 (2016) | Download Citation

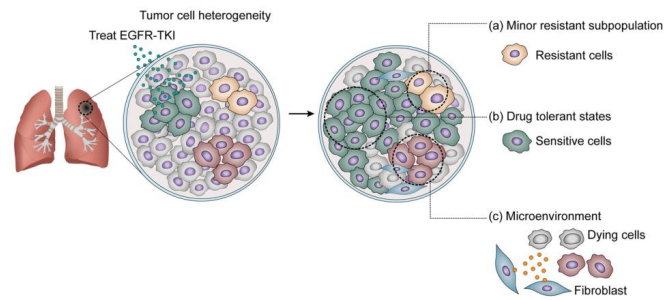
Historical trends of single cell analysis



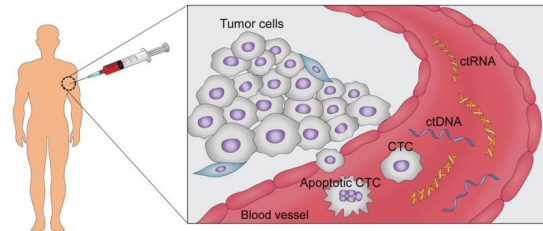
Roy et al., Sci. Adv. 2018

Many facets of scRNA-seq applications

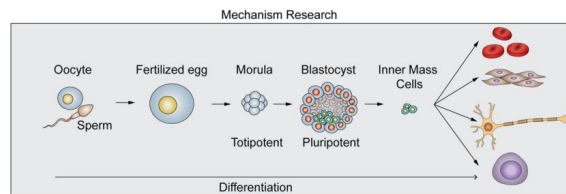
a. Drug resistance clone identification



b. Non-invasive biopsy diagnosis

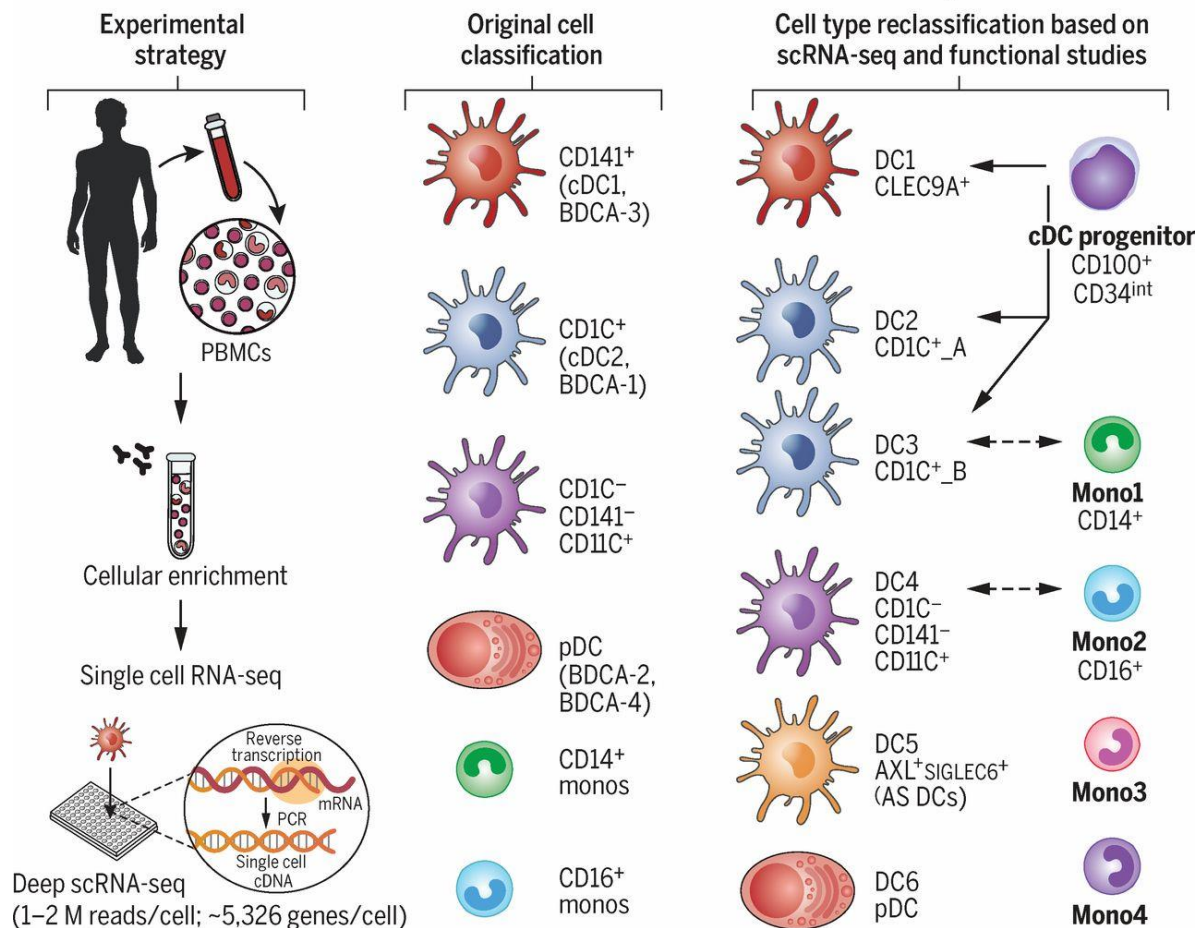


c. Single-cell lineage and stem cell regulatory network



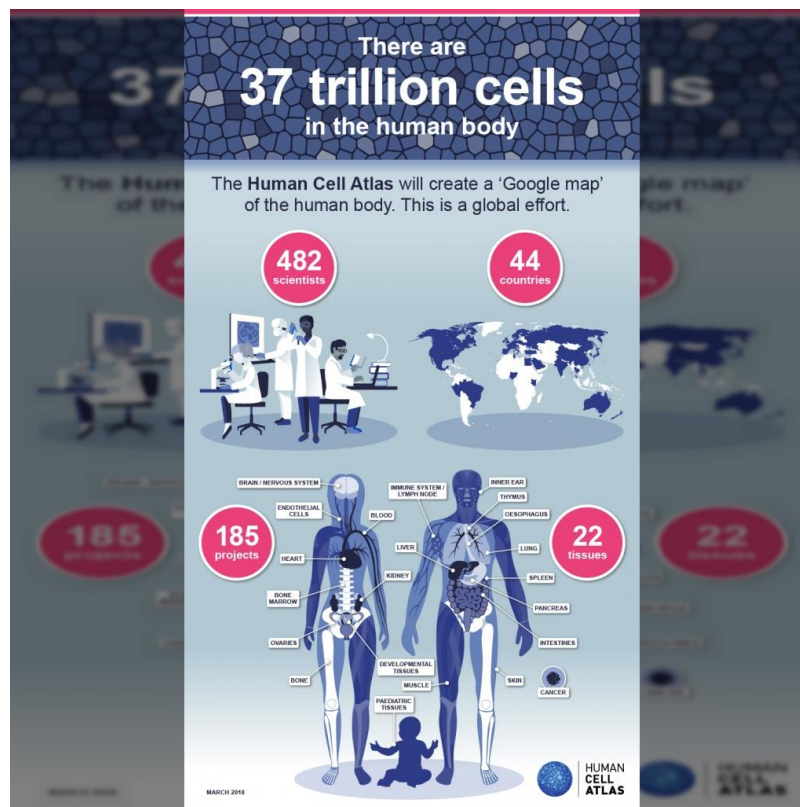
Hwang et al. *Experimental & Molecular Medicine*. 2018

Atlas of human blood dendritic cells and monocytes



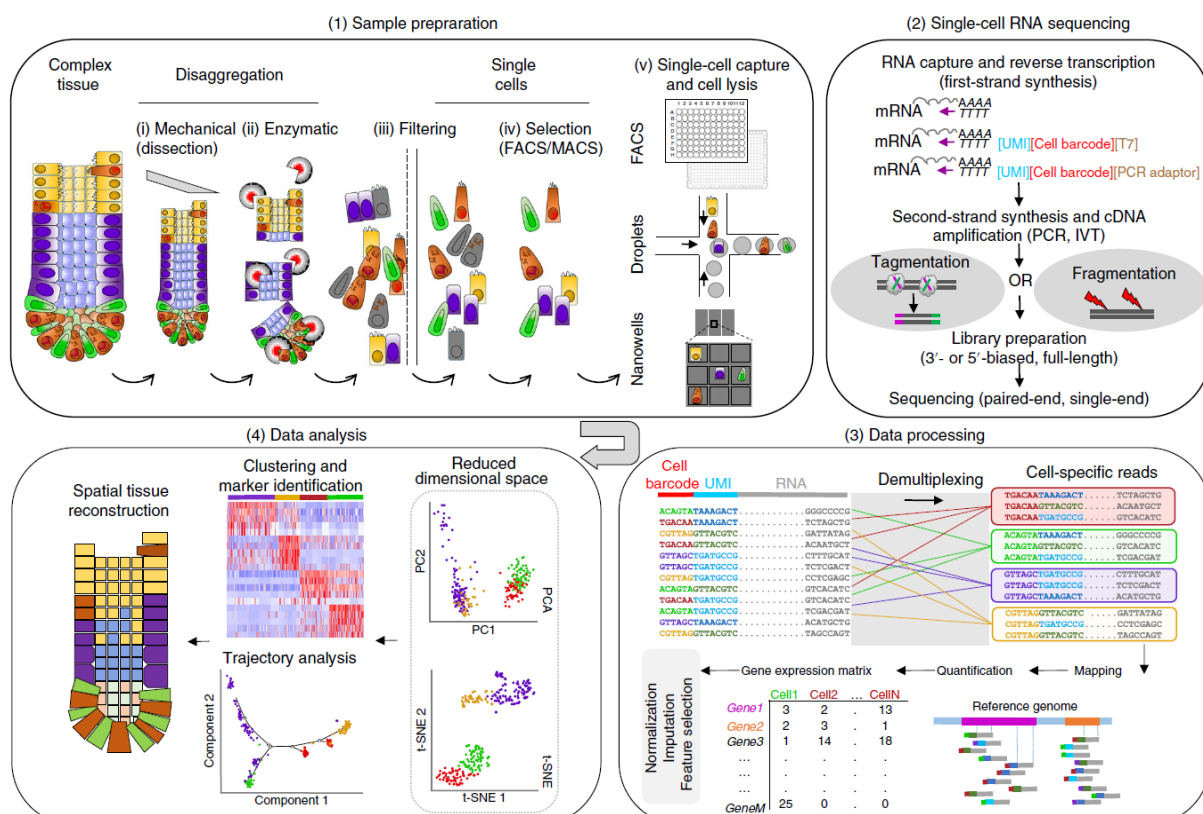
Villani et al. *Science*. 2017

The Human Cell Atlas

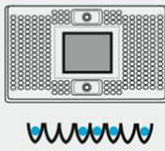
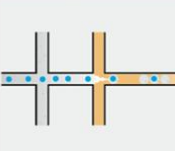


<https://www.humancellatlas.org/>

Single-cell RNA sequencing process

Lafzi et al. *Nat Prodoc.* 2018

Single-cell RNA sequencing platforms

	Micro-manipulation / Automated Pipetting	FACS	Microwell encapsulation	Droplet encapsulation
				
Cell Stress	Low	Moderate	Moderate	Moderate
Selection	Yes	Yes	No* / Yes ⁺⁺	No*
Doublet	Low	Low	Low-High	Moderate
Throughput	Low	Moderate	Moderate	High
Capture efficiency	Low	Moderate	Moderate	Low-Moderate
Academic / Commerical scRNA workflow	- CellenONE (Cellenion)* - Smart-Seq2 (42)	- MARS-Seq (39) - Smart-Seq2 (42)	- C1 (Fluidigm) - ddSeq (Biorad / Illumina) - ICell8 (Clontech) ⁺⁺ - Rhapsody (BD)	- InDrop (1CellBio) - DropSeq (Dolomite-bio) - 10X (Chromium)
Example of use	Fragile rare cells	Rare cells based on phenotype or marking	Large cell numbers	Large cell numbers

	FACS		Microwell encapsulation				Droplet encapsulation		
	Smart-Seq2	MARS-Seq	C1	ddSeq	ICell8	Rhapsody	InDrop	DropSeq	10X
Singlet Capture efficiency	82%	92%	39%	2.6%	37% ⁺⁺	Not reported	7%	Not reported	50%
Doublet rate	Not reported	2.27%	3-30%	5.8%	1.3-4%	0.6%	4%	0.36-11.3	1.6-3%
Reference	42	39	37 FWP	PB	PB	PB	36	37	26

‡Automated pipetting system

*Preselection or enrichment can be performed prior

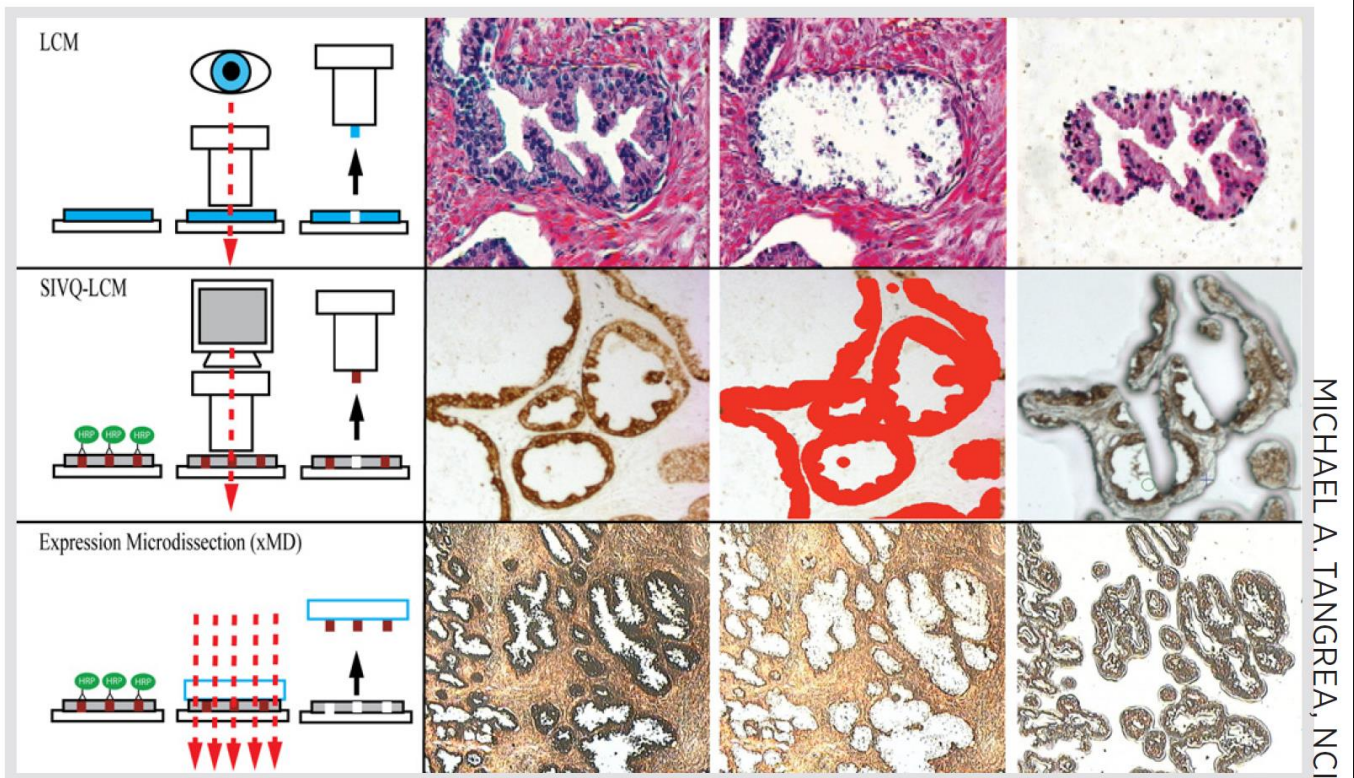
⁺⁺Only reagents added to wells containing singlets, determined by system

FWP: Fluidigm white paper

PB: Product brochure / manual

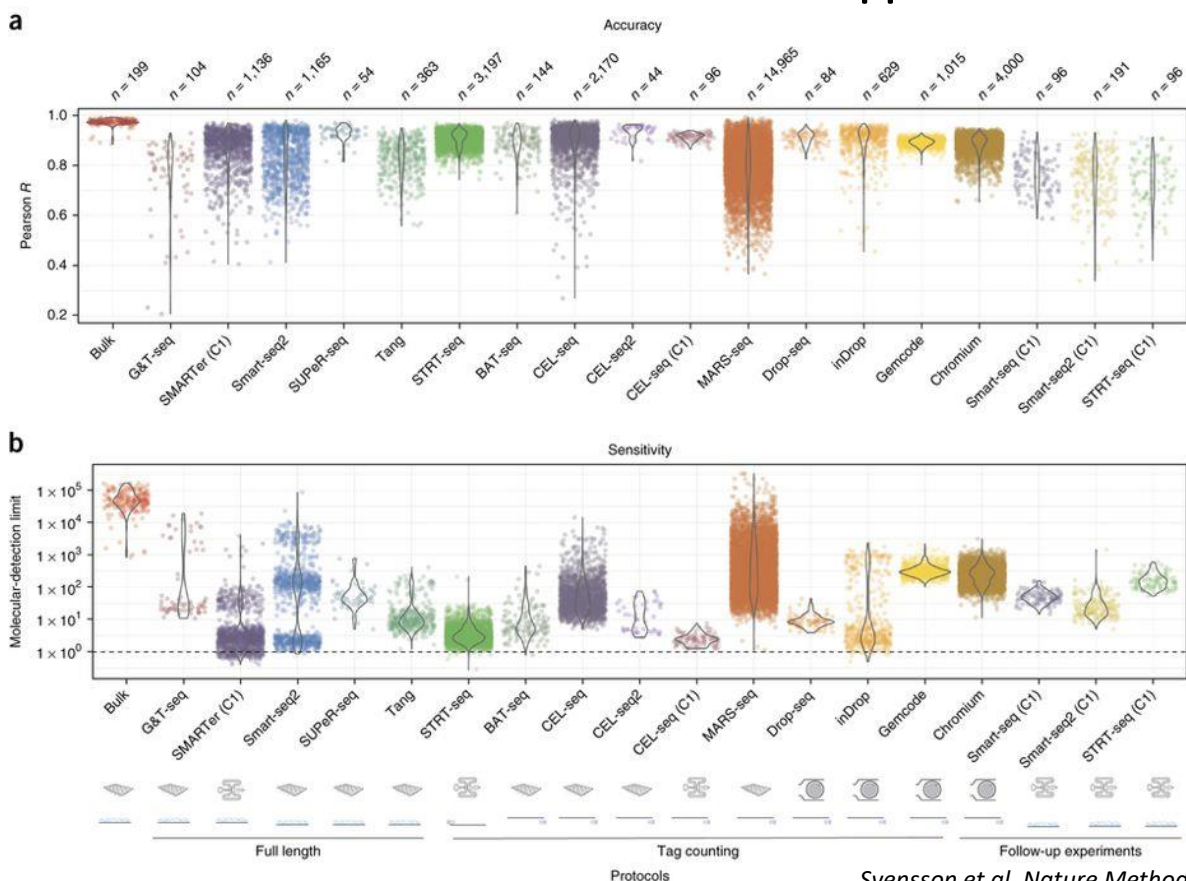
Nguyen et al. *Front. Immunol.* 2018

Laser capture microdissection (LCM)



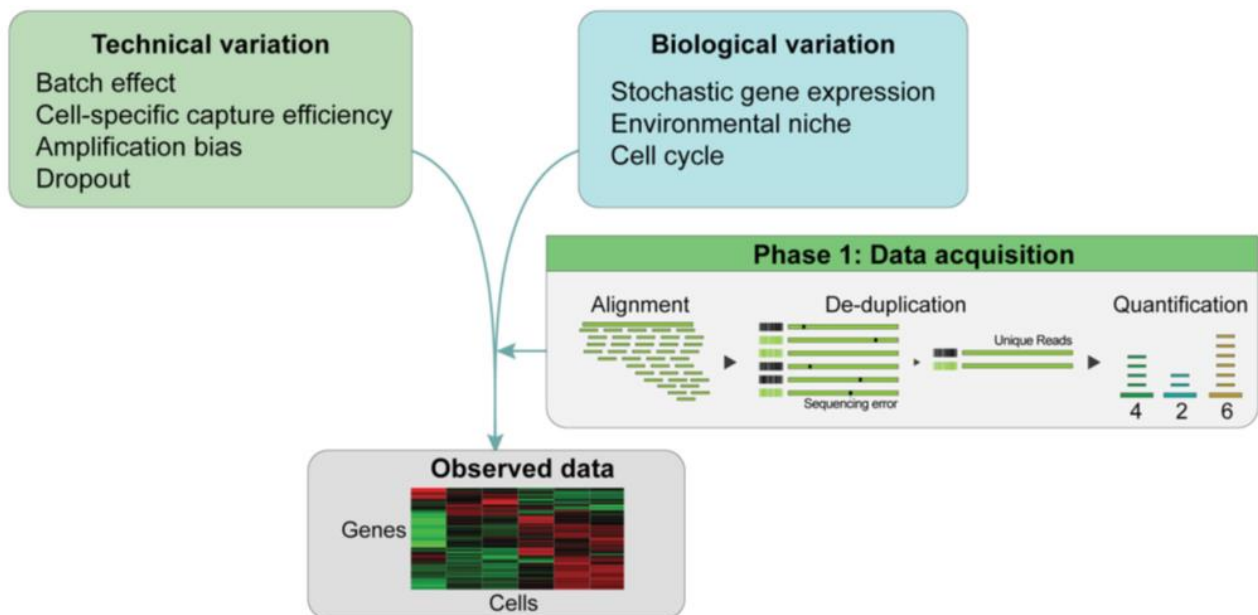
<https://irp.nih.gov/catalyst/v19i6/new-methods>

Performance metrics for scRNA-seq protocols



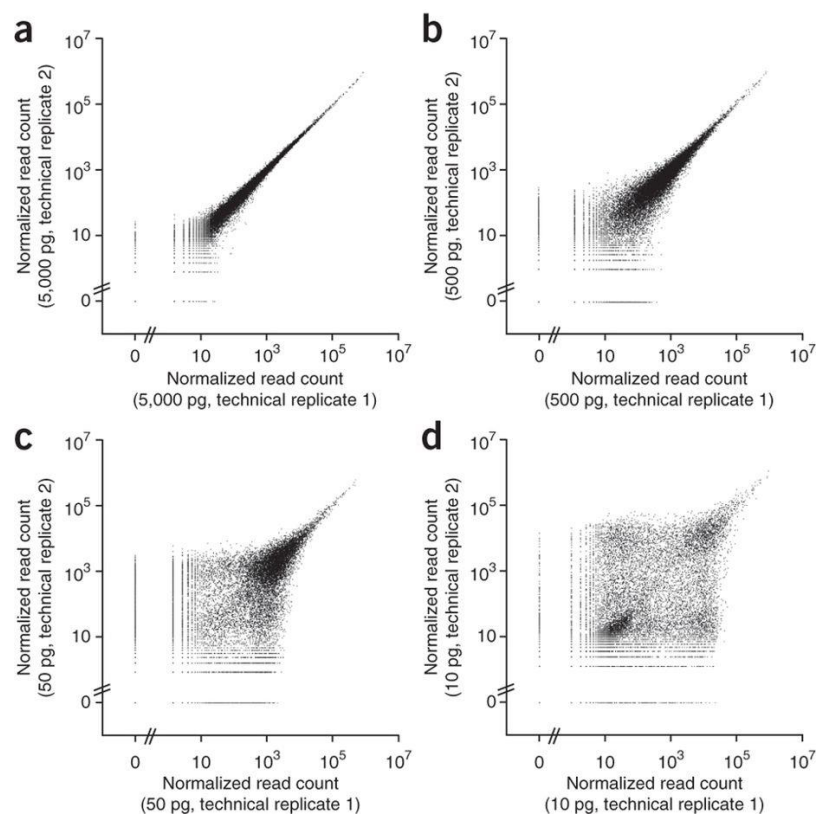
Svensson et al. Nature Methods. 2017

Technical noise in scRNA-seq



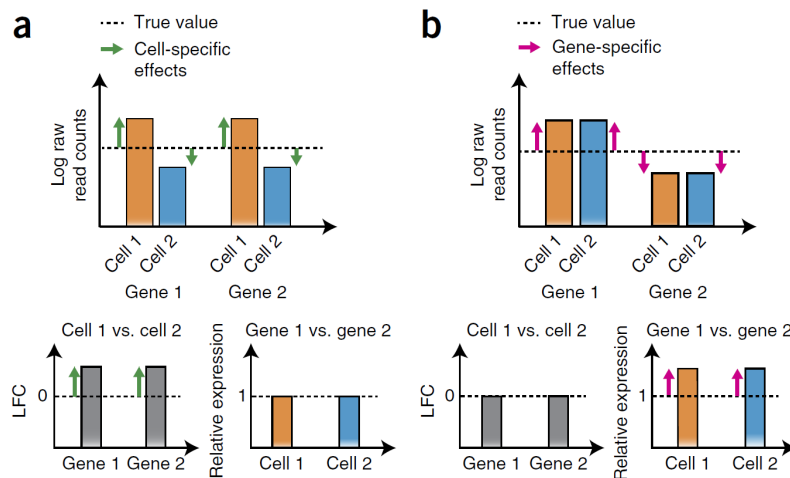
Hwang et al. *Experimental & Molecular Medicine*. 2018

Technical noise in scRNA-seq



Brennecke et al. *Nature Methods*. 2013

Cell and gene-specific effects in RNA-seq experiments



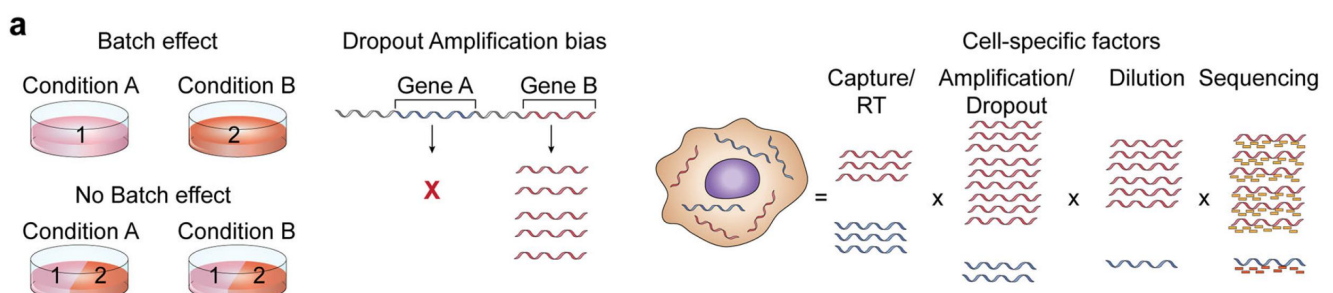
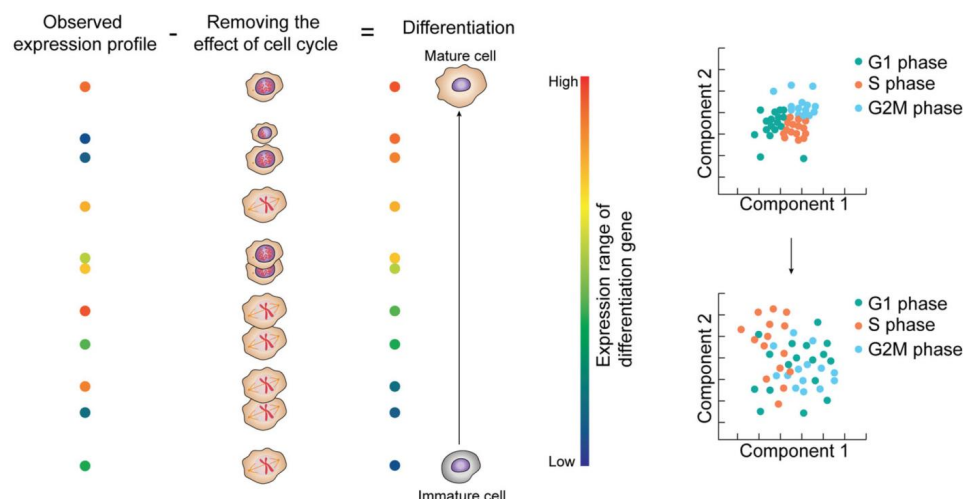
There are several experimental sources of systematic biases that can affect measurements of gene expression, including gene- and cell-specific features

c

	Cell-specific effects	Gene-specific effects	Not removed by UMIs
Sequencing depth	✓		✓
Amplification	✓	✓	
Capture and RT efficiency	✓	✓	✓
Gene length		✓	
GC content	✓	✓	✓
mRNA content	✓		✓

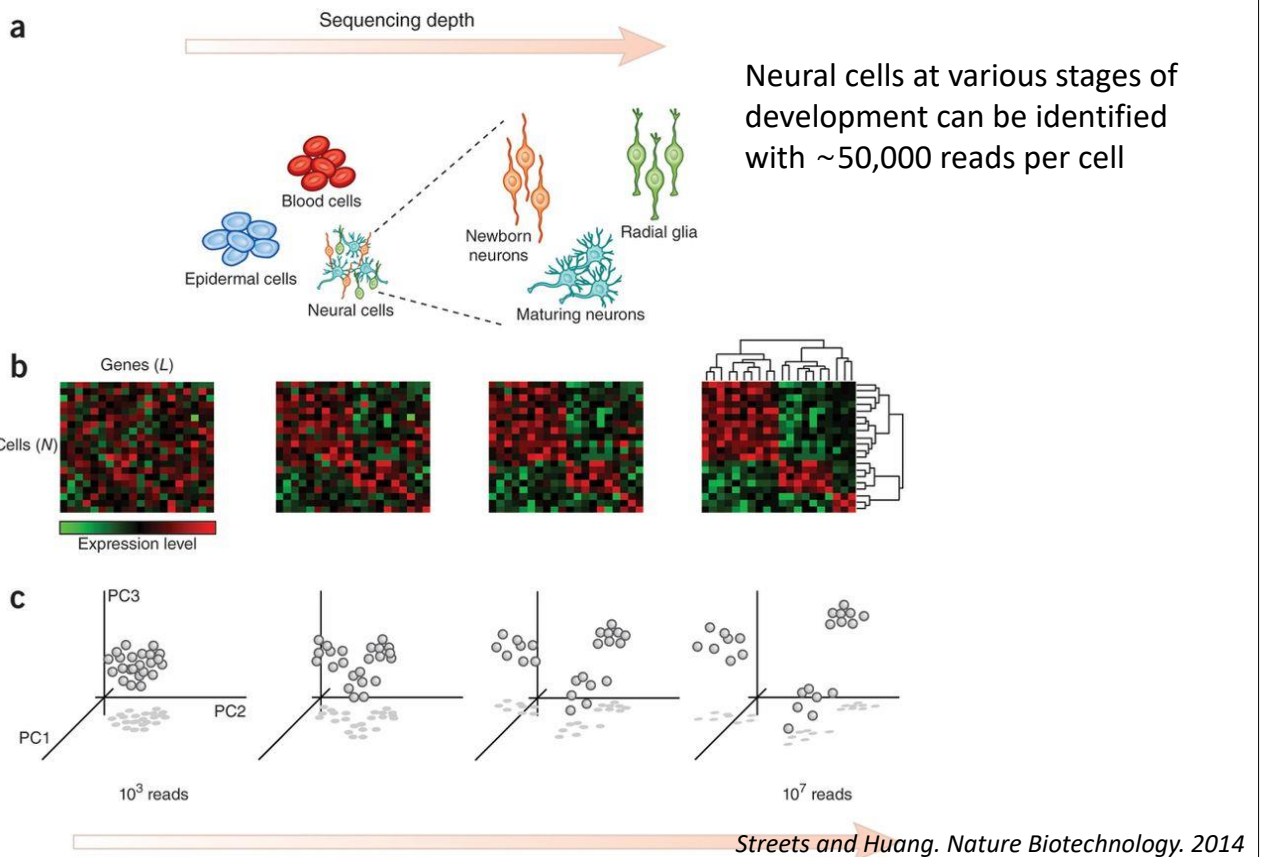
Vallejos et al. Nature Methods. 2017

Technical noise in scRNA-seq

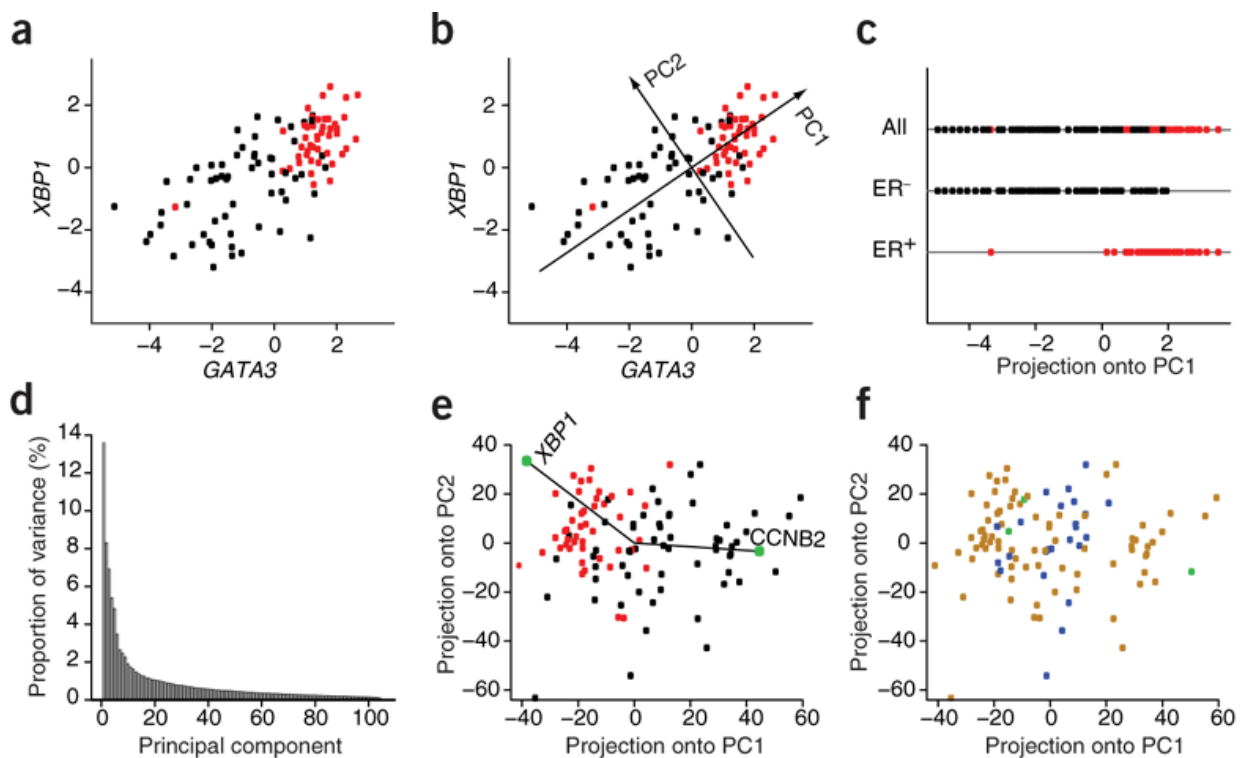


Hwang et al. Experimental & Molecular Medicine. 2018

The effect of sequencing depth



Principal Component Analysis (PCA)



Ringnér et al. Nature Biotechnology. 2008

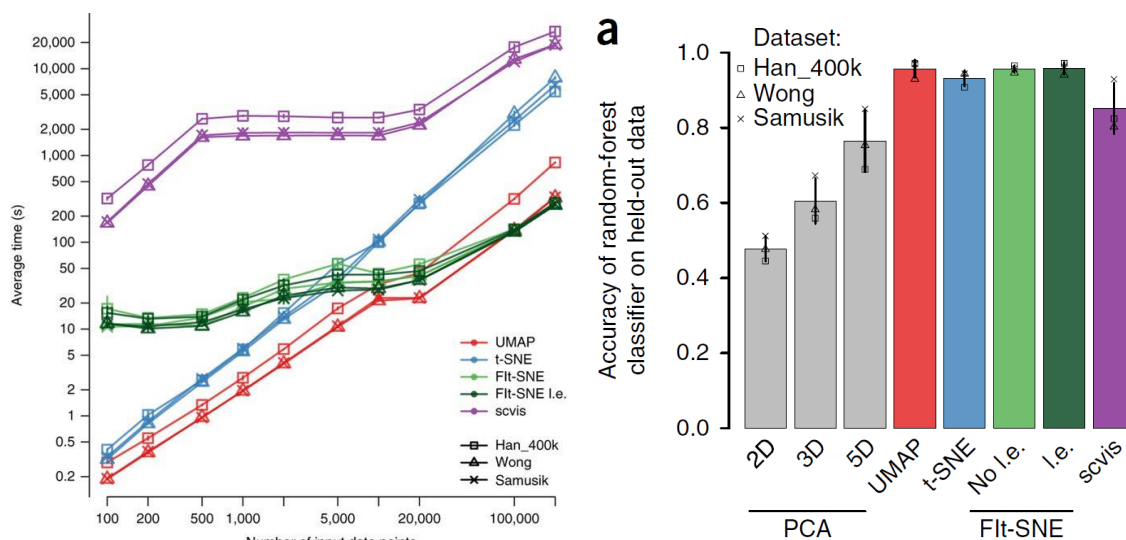
T-distributed stochastic neighborhood embedding (t-SNE)

- PCA has historically been the most commonly used method for dimensionality reduction.
- The importance of nonlinear dimensionality reduction techniques has recently been recognized.
 - able to avoid overcrowding of the representation
 - Isomap, Diffusion Map and t-SNE
- t-SNE is currently the most commonly used technique in single-cell analysis
 - t-SNE suffers from limitations such as loss of large-scale information
 - slow computation time
 - inability to meaningfully represent very large datasets

Becht et al. Nature Biotechnology. 2019

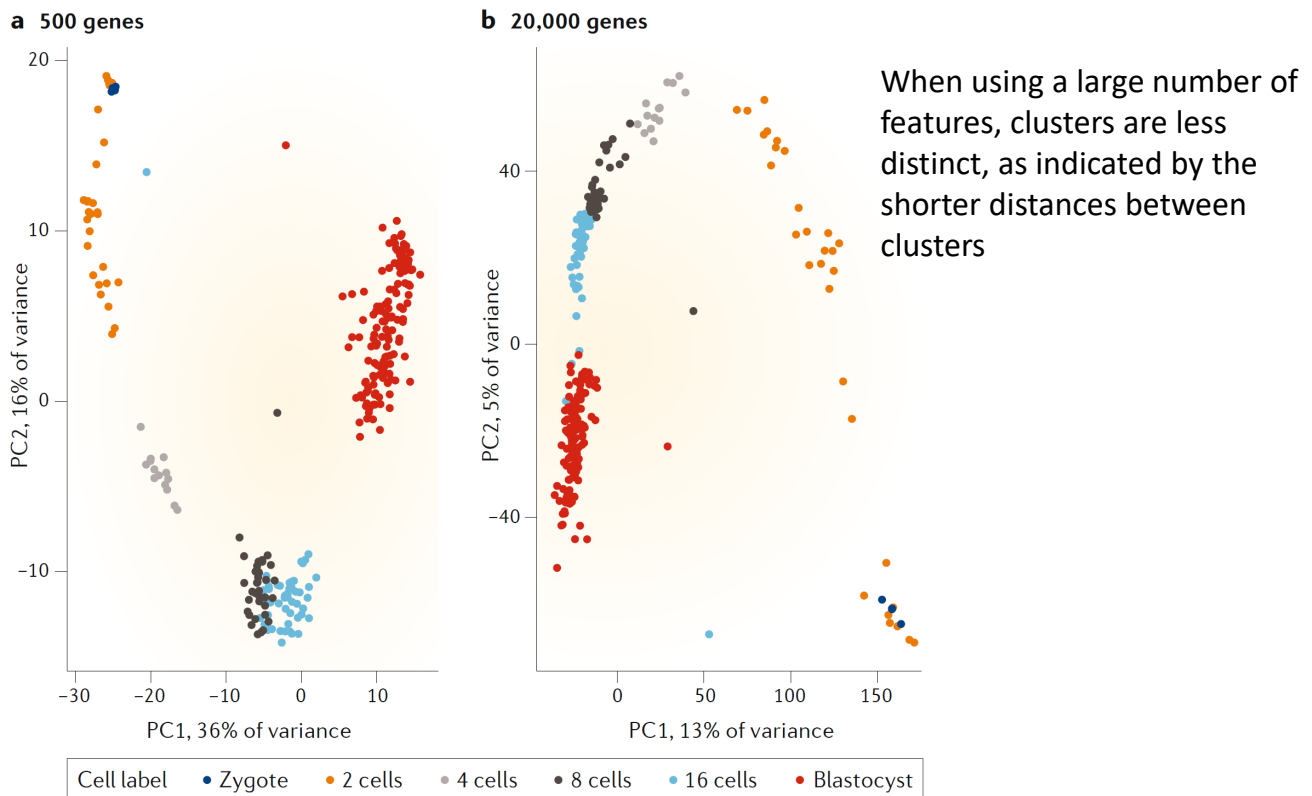
Uniform manifold approximation and projection (UMAP)

- Preserve as much of the local and more of the global data structure than t-SNE, with a shorter run time



Becht et al. Nature Biotechnology. 2019

The curse of dimensionality



Kiselev et al. Nature Reviews. 2019

Clustering analysis

Table 1 | Clustering methods for scRNA-seq

Name	Year	Method type	Strengths	Limitations
scanpy ⁴	2018	PCA + graph-based	Very scalable	May not be accurate for small data sets
Seurat (latest) ³	2016			
PhenoGraph ³²	2015			
SC3 (REF. ²²)	2017	PCA + k-means	High accuracy through consensus, provides estimation of k	High complexity, not scalable
SIMLR ²⁴	2017	Data-driven dimensionality reduction + k-means	Concurrent training of the distance metric improves sensitivity in noisy data sets	Adjusting the distance metric to make cells fit the clusters may artificially inflate quality measures
CIDR ²⁵	2017	PCA + hierarchical	Implicitly imputes dropouts when calculating distances	
GiniClust ²⁵	2016	DBSCAN	Sensitive to rare cell types	Not effective for the detection of large clusters
pcaReduce ²⁷	2016	PCA + k-means + hierarchical	Provides hierarchy of solutions	Very stochastic, does not provide a stable result
Tasic et al. ²⁸	2016	PCA + hierarchical	Cross validation used to perform fuzzy clustering	High complexity, no software package available
TSCAN ⁴¹	2016	PCA + Gaussian mixture model	Combines clustering and pseudotime analysis	Assumes clusters follow multivariate normal distribution
mpath ⁴⁵	2016	Hierarchical	Combines clustering and pseudotime analysis	Uses empirically defined thresholds and a priori knowledge
BackSPIN ²⁶	2015	Biclustering (hierarchical)	Multiple rounds of feature selection improve clustering resolution	Tends to over-partition the data
RaceID ²³ , RaceID2 (REF. ¹¹⁵), RaceID3	2015	k-Means	Detects rare cell types, provides estimation of k	Performs poorly when there are no rare cell types
SINCERA ⁵	2015	Hierarchical	Method is intuitively easy to understand	Simple hierarchical clustering is used, may not be appropriate for very noisy data
SNN-Cliq ⁶⁰	2015	Graph-based	Provides estimation of k	High complexity, not scalable

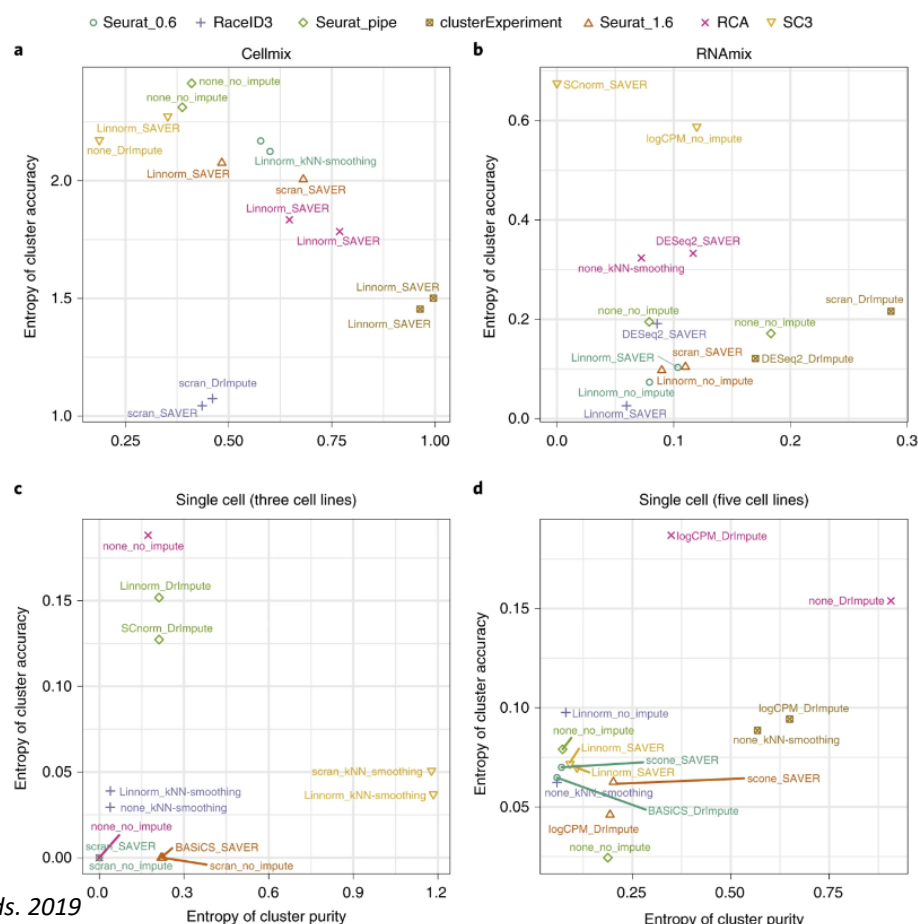
DBSCAN, density-based spatial clustering of applications with noise; PCA, principal component analysis; scRNA-seq, single-cell RNA sequencing.

Kiselev et al. Nature Reviews. 2019

Computational challenges

- A large data set ensures that analyses will have high power and improves the ability to detect rare cell types.
- Although it is possible to cluster such large data sets in a time span of hours, visualizing and interpreting the clustering results is difficult.
- Linear transformations, such as PCA, are unable to accurately capture relationships between cells because of the high levels of dropout and noise.
- Nonlinear techniques, such as tSNE and UMAP, can provide outcomes that are often more aesthetically pleasing and easier to interpret by visual inspection.
- They contain parameters that are required to be manually defined by the user and can strongly affect the visualization.

Comparisons of scRNA-seq clustering methods



Tian et al. Nature Methods. 2019

Databases

The image displays two screenshots of the Single Cell Portal. The top screenshot is the homepage, featuring the portal's logo, a tagline, and statistics on featured studies and cells. The bottom screenshot shows the 'Study Overview' page for a specific study, including a t-SNE plot of cell clusters and various interactive tools for data exploration.

Single Cell Portal BETA

Help Sign In

Single Cell PORTAL
Reducing barriers and accelerating single-cell research

Featuring
95 studies
3,436,482 cells

Search Studies Search Genes

Search Studies... Help Most Recent Most Popular Reset Filters

Single Cell PORTAL > Study Overview

Help Sign In

Summary Explore Download

Search genes Clusters View Options

Y X

Major cell types coordinates (top-level clusters)

Load cluster
Coordinates_Major_cell_types

Select annotation
CLUSTER

Subsampling threshold
All Cells

> Distribution
> Scatter
> Heatmap

Data availability. All raw sequencing data are available in [ArrayExpress](#) under accessions [E-MTAB-6149](#) and [E-MTAB-6653](#). Also, Rds files were uploaded. These can be imported in CellView to visualise clusters, scroll through tSNE projections and explore gene expression. Moreover, scRNA-seq source data were formatted as .loom files, which can be visualized in an interactive manner through SCoPe (<https://gbiomed.kuleuven.be/scRNAseq-NSCLC>)⁵⁴. Finally, gene expression data for all 52 clusters are available in Supplementary Table 4, and cluster-specific gene expression data for tumor-derived and non-malignant lung-tissue-derived cells are available in Supplementary Table 5 (only for clusters having >100 cells from both sources).

Lambrechts et al. Nat Med. 2018

EMBL-EBI

Services

Research

Training

About us

EMBL-EBI

Hinxton



ArrayExpress

Examples: E-MEXP-31, cancer, p53, Geuvadis

advanced search

Home

Browse

Submit

Help

About ArrayExpress

Contact Us

Login

ArrayExpress – functional genomics data

ArrayExpress Archive of Functional Genomics Data stores data from high-throughput functional genomics experiments, and provides these data for reuse to the research community.

 [Browse ArrayExpress](#)

Data Content

Updated today at 03:00

- 72158 experiments
- 2373008 assays
- 54.54 TB of archived data

<https://www.ebi.ac.uk/arrayexpress/>

E-MTAB-6149 - Single cell sequencing of lung carcinoma

Processed data	E-MTAB-6149.processed.1.zip	6.7 MB	14 March 2018, 13:36
	E-MTAB-6149.processed.2.zip	13.2 MB	14 March 2018, 13:36
	E-MTAB-6149.processed.3.zip	157.6 MB	14 March 2018, 13:36
	E-MTAB-6149.processed.4.zip	57.4 MB	14 March 2018, 13:36
	E-MTAB-6149.processed.5.zip	40.4 MB	14 March 2018, 13:36
	E-MTAB-6149.processed.6.zip	5.8 MB	14 March 2018, 13:36
	E-MTAB-6149.processed.7.zip	7.6 MB	14 March 2018, 13:36
Investigation description	E-MTAB-6149.idf.txt	6 KB	9 July 2018, 16:39
	E-MTAB-6149.sdrf.txt	66 KB	15 March 2018, 08:18
Sample and data relationship	1247.R1.fastq.gz	7.94 GB	14 March 2018, 13:45
	1247.R2.fastq.gz	1.72 GB	14 March 2018, 13:46
	1247.R3.fastq.gz	5.35 GB	14 March 2018, 13:46
	BT1249.R1.fastq.gz	5.47 GB	14 March 2018, 13:47
	BT1249.R2.fastq.gz	1.14 GB	14 March 2018, 13:47
	BT1249.R3.fastq.gz	3.69 GB	14 March 2018, 13:47
	BT1290_R1.fastq.gz	1.48 GB	14 March 2018, 13:47
	BT1290_R2.fastq.gz	2.94 GB	14 March 2018, 13:47
	BT1291_R1.fastq.gz	6.92 GB	14 March 2018, 13:48
	BT1291_R2.fastq.gz	14.61 GB	14 March 2018, 13:49
Raw data			

PanglaoDB is a database for the scientific community interested in exploration of single cell RNA sequencing experiments from mouse and human. We collect and integrate data from multiple studies and present them through a unified framework.

Usage examples

- Run a gene search for [SOX2](#), [PECAM1](#) or [ACE2](#)
- Browse the full list of [samples](#)
- Explore the list of cell type markers for [Schwann cells](#)
- Browse cell types of the mouse [retina](#)
- Look at the expression of [CRX](#) in photoreceptor cells
- Find cell clusters where [both](#) [PECAM1](#) and [VCAM1](#) are expressed using a [boolean search](#) with the 'and' operator
- Find [quiescent neural stem cells](#) using AND+NOT

How to cite

Oscar Franzén, Li-Ming Gan, Johan L M Björkregren, *PanglaoDB: a web server for exploration of mouse and human single-cell RNA sequencing data*, **Database**, Volume 2019, 2019, baz046, doi:10.1093/database/baz046

What is single cell RNA sequencing?

Adapted from the [Wikipedia](#) article on the topic: *Single cell RNA sequencing examines the transcriptomes from individual cells with optimized next generation sequencing technologies, providing a higher resolution of gene expression and a better understanding of the function of an individual cell in the context of its microenvironment.*

Database statistics		
	<i>Mus musculus</i>	<i>Homo sapiens</i>
Samples	1063	305
Tissues	184	74
Cells	4,459,768	1,126,580
Clusters	8,651	1,748

Dataset of the day

Take a closer look at the cellular composition of [Rib](#), using a dataset which consists of 1195 cells. Clustering of this dataset resulted in 6 cell clusters, containing among others, [Chondrocytes](#).

News

- 21-05-2020** Ongoing work to move to new hosting.
 - 30-01-2020** A corrupted MySQL table caused dysfunction in the search function, the problem has now been fixed.
 - 28-11-2019** We are looking for sponsors to host PanglaoDB. We have modest requirements (VPS with Ubuntu, etc). Please get in touch with us if you can provide help (contact@panglaoDB.se).
 - 01-07-2019** Updated the 2d view for data sets (now colors by cell type and not by cluster and colors are consistent across data sets). For example, see [this data set](#).
 - 16-05-2019** Added more markers for [Tannocytes](#).
 - 07-05-2019** Added markers for [Chromaffin cells](#).
 - 01-05-2019** Markers for an additional cell types added: meet the [sebocyte](#).
 - 30-04-2019** Added sensitivity and specificity to the [marker](#) list (shown separately for mouse and human).
- [Show older news](#)

[admin login](#)

PanglaoDB - A gateway to mouse & human single cell exploration.
Feedback: contact@panglaoDB.se or using [this form](#).

[Page bottom](#)

Samples

This page shows the samples included in PanglaoDB. The database currently has **1368** scRNA-seq dataset samples. The controls below can be used to filter by species and sequencing protocol.

Filter by species Human	Filter by protocol all protocols	Sort on Most recent
Refresh		

Status	SRA	SRS	Tissue/Site	Protocol	Species	No. Cells	Action
👍	SRA553822	SRS2119548	Cultured embryonic stem cells	10x chromium	Homo sapiens	6501	view
👍	SRA598936	SRS2428405	Kidney cortex	10x chromium	Homo sapiens	3759	view
👍	SRA608611	SRS2517316	Lung progenitors	10x chromium	Homo sapiens	1077	view
👍	SRA608611	SRS2517317	Lung progenitors	10x chromium	Homo sapiens	906	view
👍	SRA711898	SRS3349280	Tonsil	10x chromium	Homo sapiens	3503	view
👍	SRA605557	SRS2493625	Cd14+ monocytes	10x chromium	Homo sapiens	411	view
👍	SRA605557	SRS2493626	Cd14+ monocytes	10x chromium	Homo sapiens	409	view
👍	SRA678017	SRS3112005	Embryonic kidney cortex	10x chromium	Homo sapiens	4231	view
👍	SRA678017	SRS3112006	Embryonic kidney cortex	10x chromium	Homo sapiens	3800	view
👍	SRA695134	SRS3218229	Alveolar rhabdomyosarcoma	10x chromium	Homo sapiens	8992	view
👍	SRA550660	SRS2089635	Peripheral blood mononuclear cells	10x chromium	Homo sapiens	1860	view
👍	SRA550660	SRS2089636	Peripheral blood mononuclear cells	10x chromium	Homo sapiens	1580	view
👍	SRA550660	SRS2089637	Peripheral blood mononuclear cells	10x chromium	Homo sapiens	10,292	view
👍	SRA550660	SRS2089638	Peripheral blood mononuclear cells	10x chromium	Homo sapiens	1818	view
👍	SRA550660	SRS2089639	Peripheral blood mononuclear cells	10x chromium	Homo sapiens	10,940	view
👍	SRA646572	SRS2833946	Human embryo forebrain	10x chromium	Homo sapiens	6943	view
👍	SRA646572	SRS2833947	Human embryo forebrain	10x chromium	Homo sapiens	4127	view
👍	SRA650215	SRS2853842	Fetal kidney	drop-seq	Homo sapiens	1372	view
👍	SRA650215	SRS2853843	Fetal kidney	drop-seq	Homo sapiens	789	view
👍	SRA650215	SRS2857943	Fetal kidney	drop-seq	Homo sapiens	362	view
👍	SRA628554	SRS2664364	Peripheral blood mononuclear cell	10x chromium	Homo sapiens	3352	view
👍	SRA628554	SRS2664365	Peripheral blood mononuclear cell	10x chromium	Homo sapiens	3081	view

[Home](#)
[Search](#)
[Help](#)
[About](#)

scRNASeqDB

scRNASeqDB

a database for gene expression profiling in human single cell by RNA-seq

Welcome to scRNASeqDB!

Single-cell RNA-Seq (scRNA-seq) are an emerging method which facilitates to explore the comprehensive transcriptome in a single cell. To provide a useful and unique reference resource for biology and medicine, we developed the scRNASeqDB database, which contains 36 human single cell gene expression data sets collected from Gene Expression Omnibus (GEO), involving 8910 cells from 174 cell groups. We also provides detailed information for gene expression of cells in different status, as well as some features, including heatmap and boxplot of gene expression, gene correlation matrix, GO and pathway annotations.

You can also [submit](#) scRNASeq data sets to our database. Feel free to [contact](#) us if you have any questions!

Search scRNASeqDB

By Gene

By Cell

☒ Gene symbol
 ☐ Gene Ensembl ID

Please input gene symbol of Ensembl ID

Gene Cloud

Current curation

Number of GSE datasets:	38
Number of GSM entries:	13440
Number of cell groups:	200

New datasets

GSE86982	REGION-SPECIFIC NEURAL STEM CELL LINEAGES REVEALED BY SINGLE-CELL RNA-SEQ FROM HUMAN EMBRYONIC STEM CELLS [Smart-seq]
GSE86977	REGION-SPECIFIC NEURAL STEM CELL LINEAGES REVEALED BY SINGLE-CELL RNA-SEQ FROM HUMAN EMBRYONIC STEM CELLS [Cel-seq]
GSE77564	Coupled electrophysiological recording and single-cell transcriptome analyses revealed molecular mechanisms underlying neuronal maturation

Publication

Yuan Cao, Junjie Zhu, Guangchun Han, Peilin Jia, Zhongming Zhao.
scRNASeqDB: a database for gene expression profiling in human single cell by RNA-seq (in review). [bioRxiv](#)

News

GSE86982 has been added to our database.	2017/03/31
GSE86977 has been added to our database.	2017/03/29
CIDR has been used to cluster single cells in each dataset.	2017/03/12
Rankprod has been used to rank gene expression across all datasets.	2017/03/03
scRNASeqDB has been launched.	2016/09/15

Journal papers on scRNA-seq analysis of cancer

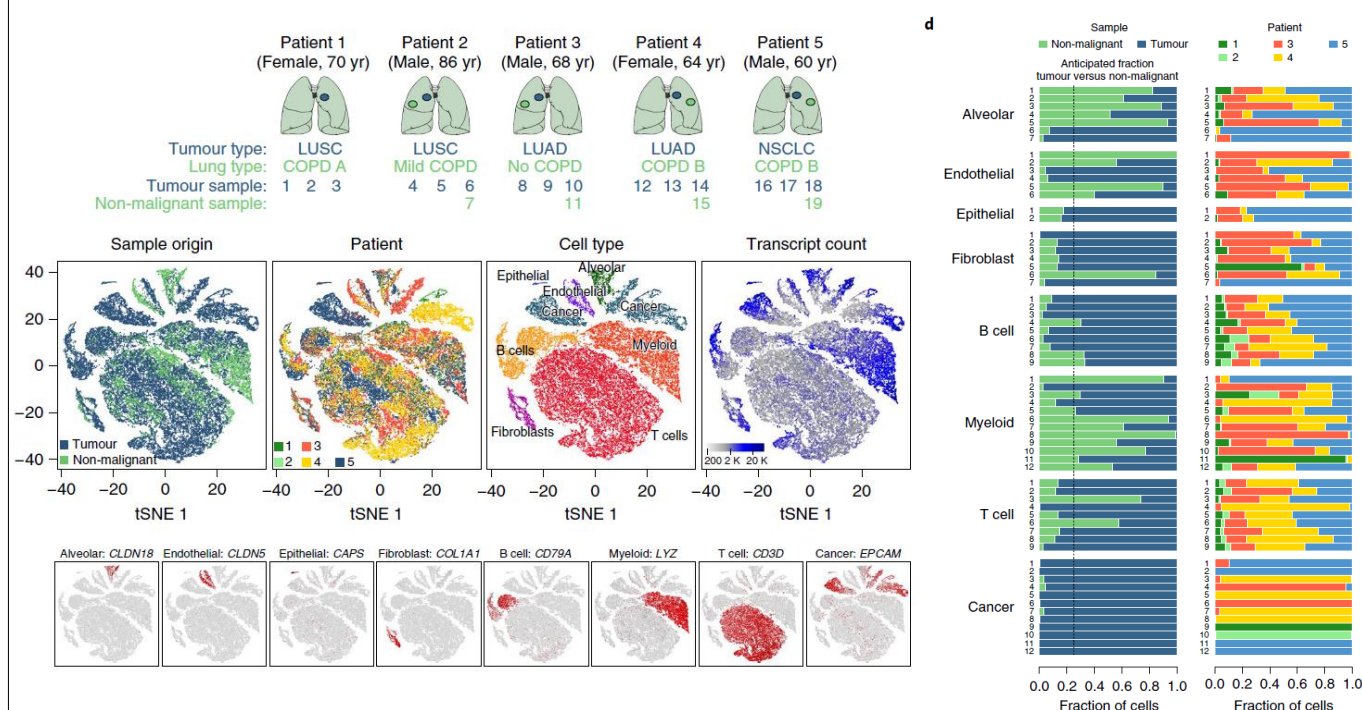
Phenotype molding of stromal cells in the lung tumor microenvironment

Diether Lambrechts^{1,2*}, Els Wauters^{3,4}, Bram Boeckx^{1,2}, Sara Aibar^{5,6}, David Nittner^{7,8}, Oliver Burton^{6,9}, Ayse Bassez^{1,2}, Herbert Decaluwé^{10,11}, Andreas Pircher^{1,12}, Kathleen Van den Eynde¹³, Birgit Weynand¹³, Erik Verbeken¹³, Paul De Leyn¹¹, Adrian Liston^{6,9}, Johan Vansteenkiste^{13,4}, Peter Carmeliet^{1,12,14}, Stein Aerts^{5,6} and Bernard Thienpont^{1,15*}

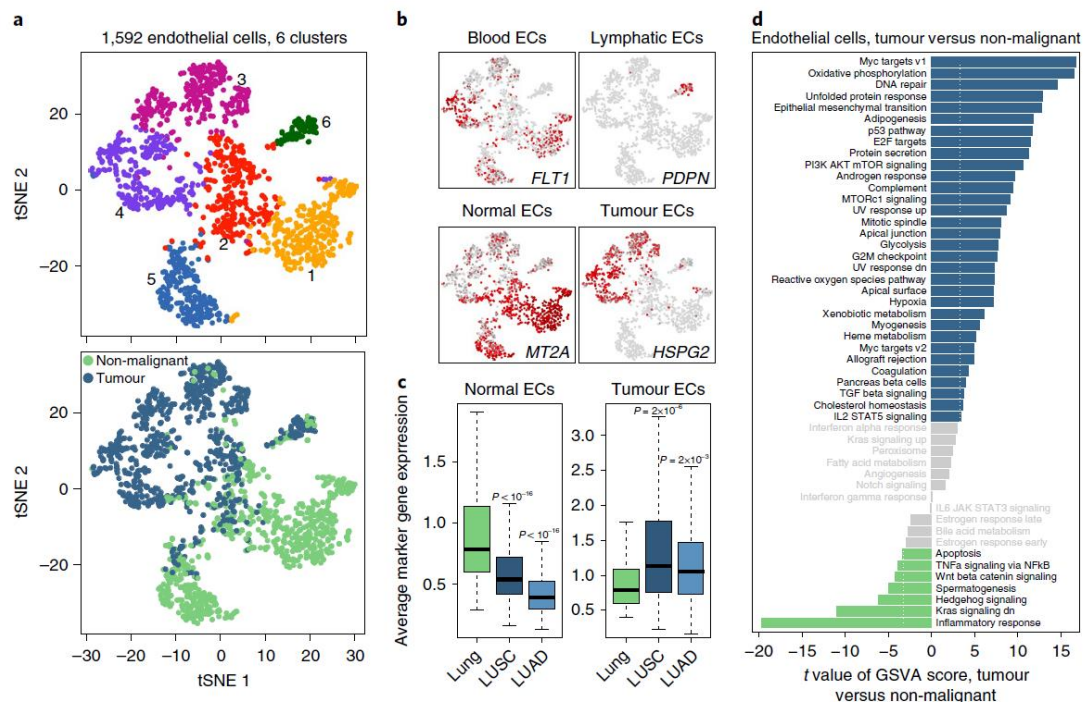
Cancer cells are embedded in the tumor microenvironment (TME), a complex ecosystem of stromal cells. Here, we present a 52,698-cell catalog of the TME transcriptome in human lung tumors at single-cell resolution, validated in independent samples where 40,250 additional cells were sequenced. By comparing with matching non-malignant lung samples, we reveal a highly complex TME that profoundly molds stromal cells. We identify 52 stromal cell subtypes, including novel subpopulations in cell types hitherto considered to be homogeneous, as well as transcription factors underlying their heterogeneity. For instance, we discover fibroblasts expressing different collagen sets, endothelial cells downregulating immune cell homing and genes coregulated with established immune checkpoint transcripts and correlating with T-cell activity. By assessing marker genes for these cell subtypes in bulk RNA-sequencing data from 1,572 patients, we illustrate how these correlate with survival, while immunohistochemistry for selected markers validates them as separate cellular entities in an independent series of lung tumors. Hence, in providing a comprehensive catalog of stromal cells types and by characterizing their phenotype and co-optive behavior, this resource provides deeper insights into lung cancer biology that will be helpful in advancing lung cancer diagnosis and therapy.

NATURE MEDICINE | VOL 24 | AUGUST 2018 | 1277-1289 | www.nature.com/naturemedicine

Many stromal cell subclusters were enriched for either tumor-derived or lung tissue-derived cells



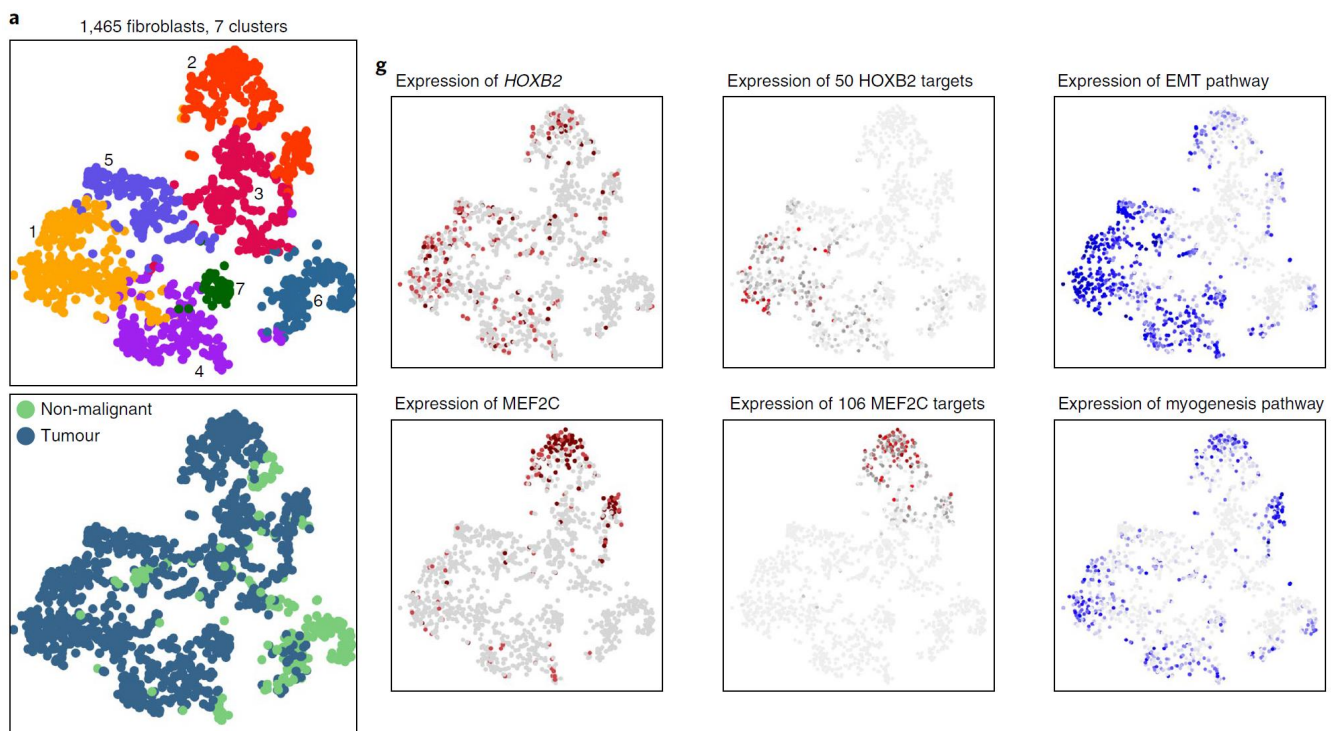
Myc targets as the top enriched signature in tumor endothelial cells



Most significantly downregulated pathway was involved in inflammatory responses.

Lambrechts et al. Nat Med. 2018

Lung tumors are enriched with fibroblasts with expression of EMT pathway



Preparation of single-cell suspensions. Following resection in the operating room, samples from the tumor and adjacent non-malignant lung tissue from the same resection specimen at maximal distance (>5 cm) from the tumor were isolated and transported rapidly to the research facility. On arrival, samples were rinsed with PBS and the tumor sample macroscopically examined for tumor positioning. The tumor sample was subsequently divided into three pieces, with one piece containing mainly tissue derived from the tumor core, one piece containing tissue mainly derived from the tumor edge and a third piece originating from the position intermediate to the other two samples. Each sample was subsequently minced on ice to smaller pieces of less than 1 mm³ and transferred to 10 ml digestion medium containing 0.2% collagenase I/II (ThermoFisher Scientific), DNase I (Sigma) and 25 units dispase (Invitrogen) in DMEM (ThermoFisher Scientific). Samples were incubated for 15 min at 37°C, with manual shaking every 5 min. Samples were then vortexed for 10 s and pipetted up and down for 1 min using pipettes of descending sizes (25 ml, 10 ml and 5 ml). Next, 30 ml ice-cold PBS, pH 7.4, (ThermoFisher Scientific) containing 2% fetal bovine serum (ThermoFisher Scientific) was added and samples were filtered using a 40-µm nylon mesh (ThermoFisher Scientific). Following centrifugation at 120×g and 4°C for 5 min, the supernatant was decanted and discarded, and the cell pellet was resuspended in 2 ml red blood cell lysis buffer and transferred to a 2-ml DNA low bind tube. Following a 5-min incubation at room temperature, samples were centrifuged (120×g, 4°C, 5 min) using a swing-out rotor. Samples were next resuspended in 1 ml PBS containing 8 µl UltraPure BSA (50 mg ml⁻¹; AM2616, ThermoFisher Scientific) and filtered over Scienceware Flowmi 40-µm cell strainers (VWR) using wide-bore 1 ml low-retention filter tips (Mettler-Toledo). Next, 10 µl of this cell suspension was counted using an automated cell counter (Luna) to determine the concentration of live cells. Throughout the dissociation procedure, cells were maintained on ice whenever possible, and the entire procedure was completed in less than 1 h (typically ~45 min) to avoid dissociation-associated artefacts recently described¹³. By using a dissociation signature¹³ to detect dissociation-associated changes in gene expression, a positive signal for less than 2% of cells was detected (Supplementary Fig. 2a).

Droplet-based scRNA-seq. Single-cell suspensions were converted to barcoded scRNA-seq libraries by using the Chromium Single Cell 3' Library, Gel Bead & Multiplex Kit and Chip Kit (10x Genomics), aiming for an estimated 4,000 cells per library and following the manufacturer's instructions. Samples were processed using kits pertaining to either the V1 or V2 barcoding chemistry of 10x Genomics (Supplementary Table 2). Single samples are always processed in a single well of a PCR plate, allowing all cells from a sample to be treated with the same master mix and in the same reaction vessel. For each patient, all samples (non-malignant and tumor) were processed in parallel in the same thermal cycler. Libraries were sequenced on an Illumina HiSeq4000, and mapped to the human genome (build hg19) using Cell Ranger (10x Genomics). Gene positions were annotated as per Ensembl build 85 and filtered for biotype (only protein-coding, long intergenic non-coding RNA, antisense, immunoglobulin or T-cell receptor).

Single-cell gene expression quantification and determination of the major cell types. Raw gene expression matrices generated per sample using CellRanger (version 2.0.0) were combined in R (version 3.3.2—*Sincere Pumpkin Patch*), and converted to a Seurat object using the Seurat R package (version 1.4.0.7)¹¹. From this, all cells were removed that had either fewer than 201 UMIs, over 6,000 or below 101 expressed genes, or over 10% UMIs derived from mitochondrial genome. From the remaining 52,698 cells, gene expression matrices were normalized to total cellular read count and to mitochondrial read count using linear regression as implemented in Seurat's *RegressOut* function. As a result, none of the principle components subsequently identified were correlated with transcript count (data not shown). From the remaining 52,698 cells, variably expressed genes were selected as having a normalized expression between 0.125 and 3, and a quantile-normalized variance exceeding 0.5. To reduce dimensionality of this dataset, the resulting 2,192 variably expressed genes were summarized by principle component analysis, and the first 8 principle components further summarized using tSNE dimensionality reduction using the default settings of the *RunTSNE* function. Cell clusters in the resulting two-dimensional representation were annotated to known biological cell types using canonical marker genes (Supplementary Fig. 1). Of note, very few stromal cells (~2%) were positive for cell proliferation markers (Supplementary Fig. 4). We therefore opted not to correct our gene expression matrices for effects of cell cycle.

Subclustering of the major cell types. To identify subclusters within these eight cell types, we reanalyzed cells belonging to each of these eight cell types separately. Specifically, we applied dimensionality reduction using principle component analysis in each cell type on variably expressed genes as described above. To identify which principle components were informative, we applied Horn's parallel analysis for principle component analysis⁴⁴ as implemented in the R *paran* package (version 1.5.1.), selecting those principle components having eigenvalues that exceed the eigenvalues generated using ten random permutations by >50%. Using the graph-based clustering approach implemented in the *FindClusters* function of the Seurat package, with a conservative resolution of 0.5 and otherwise default parameters, each cell type was reclustered by its principle components. Notably, subclustering was robust to alterations in the number of principle components, in the resolution or in the *K* parameter (Supplementary Fig. 3a–c). Moreover, few of the subclusters identified contained many cells wherein less than 300 genes were detected, indicating that increasing the threshold of 100 genes will not affect our results (Supplementary Fig. 20). This yielded 64 subclusters (52 stromal subclusters) in total, as listed in Supplementary Table 3. For visualization purposes, these informative principle components were converted into tSNE plots as above.

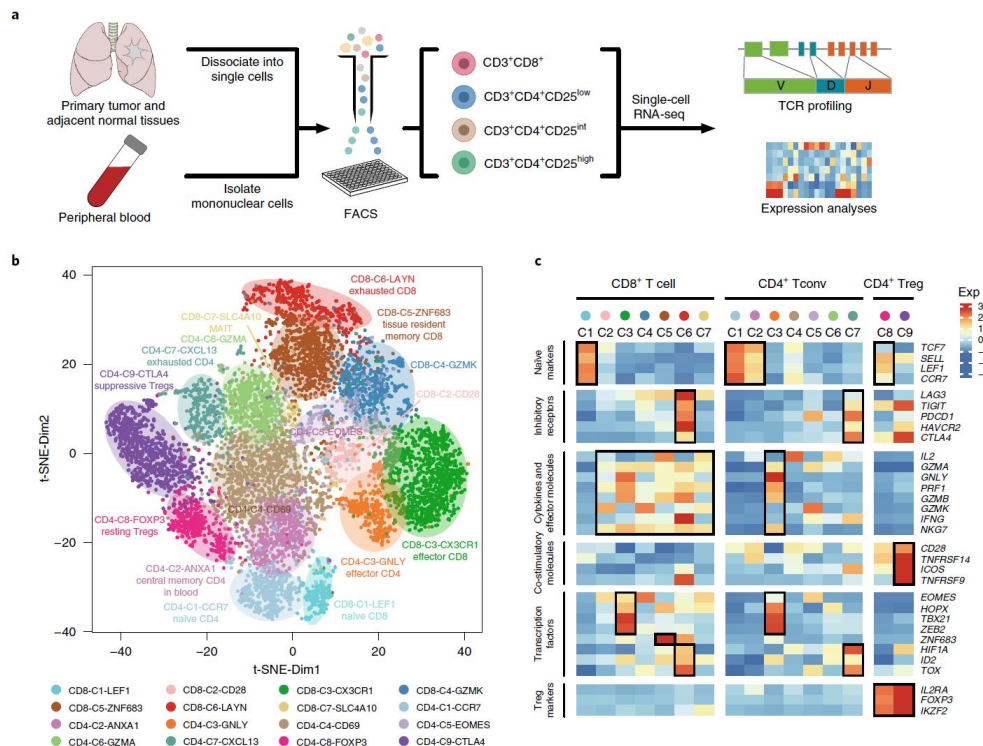
Identification of marker genes. To identify marker genes for each of these 64 subclusters within these 8 cell types, we contrasted cells from that subcluster to all other cells of that subcluster using the Seurat *FindMarkers* function. Marker genes were required to have an average expression in that subcluster that was >2.5-fold higher than the average expression in the other subclusters from that cell type, and a detectable expression in >15% of all cells from that subcluster. Additionally, marker genes were required to have the highest mean expression in that subcluster, out of all 64 subclusters. This yielded a list of in total 402 marker genes (Supplementary Table 3) for 51 subclusters (42 stromal cell subclusters), whereas for 13 subclusters we failed to identify marker genes. When analyzing marker genes for several subclusters in aggregate, such as for tumor endothelial cells (endothelial cell clusters 3 and 4) or for macrophages (myeloid clusters 1–4, 6–8, 10 and 11), we simply combined the marker genes for all associated subclusters.

Global characterization of T cells in non-small-cell lung cancer by single-cell sequencing

Xinyi Guo^{1,6}, Yuanyuan Zhang^{1,6}, Liangtao Zheng^{2,6}, Chunhong Zheng^{1,6}, Jintao Song^{3,6}, Qiming Zhang¹, Boxi Kang¹, Zhouzuerui Liu¹, Liang Jin³, Rui Xing⁴, Ranran Gao¹, Lei Zhang², Minghui Dong¹, Xueda Hu¹, Xianwen Ren¹, Dennis Kirchhoff⁵, Helge Gottfried Roeder⁵, Tiansheng Yan^{3*} and Zemin Zhang^{1,2*}

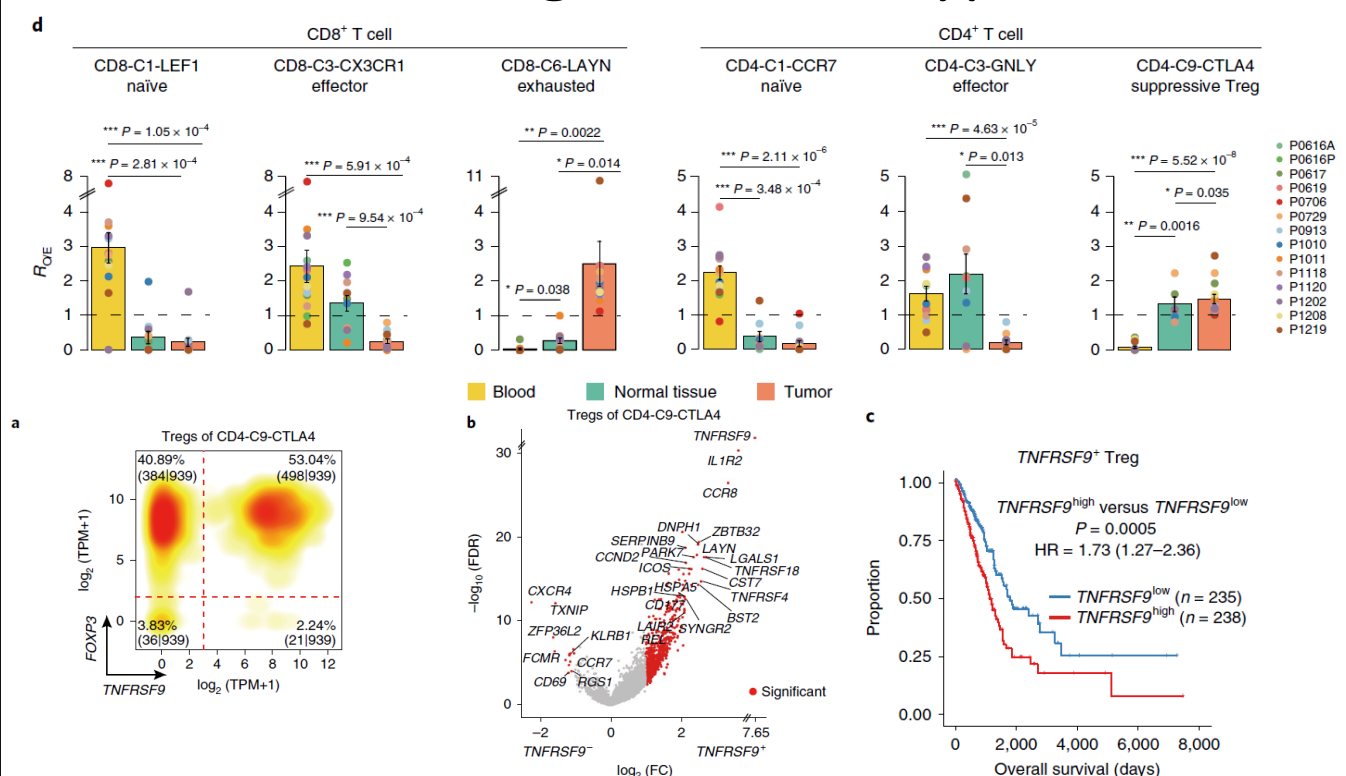
Cancer immunotherapies have shown sustained clinical responses in treating non-small-cell lung cancer^{1–3}, but efficacy varies and depends in part on the amount and properties of tumor infiltrating lymphocytes^{4–6}. To depict the baseline landscape of the composition, lineage and functional states of tumor infiltrating lymphocytes, here we performed deep single-cell RNA sequencing for 12,346 T cells from 14 treatment-naïve non-small-cell lung cancer patients. Combined expression and T cell antigen receptor based lineage tracking revealed a significant proportion of inter-tissue effector T cells with a highly migratory nature. As well as tumor-infiltrating CD8⁺ T cells undergoing exhaustion, we observed two clusters of cells exhibiting states preceding exhaustion, and a high ratio of “pre-exhausted” to exhausted T cells was associated with better prognosis of lung adenocarcinoma. Additionally, we observed further heterogeneity within the tumor regulatory T cells (Tregs), characterized by the bimodal distribution of *TNFRSF9*, an activation marker for antigen-specific Tregs. The gene signature of those activated tumor Tregs, which included *IL1R2*, correlated with poor prognosis in lung adenocarcinoma. Our study provides a new approach for patient stratification and will help further understand the functional states and dynamics of T cells in lung cancer.

Seven CD8 and nine CD4 clusters were identified



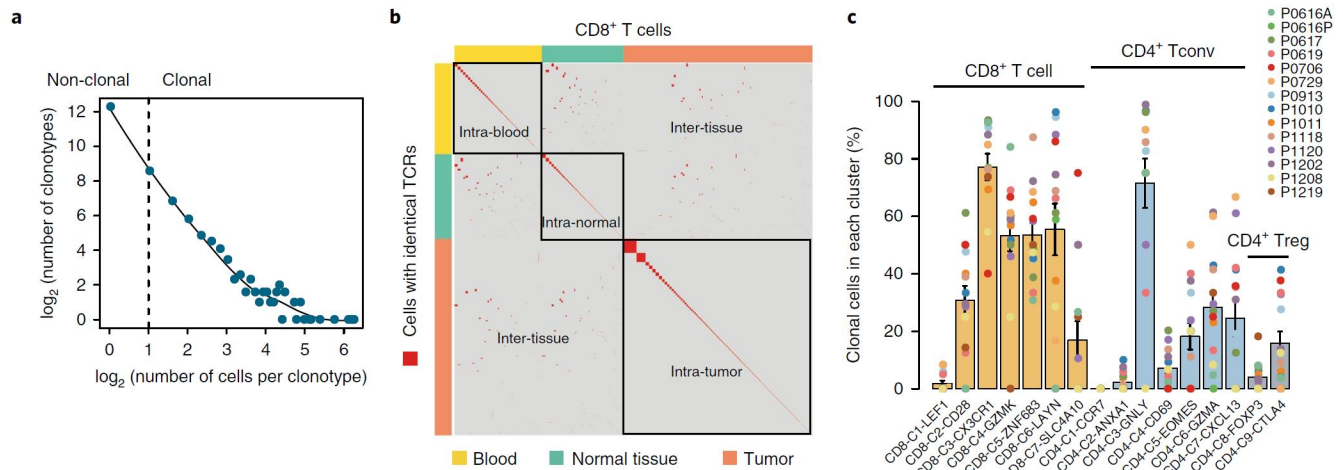
Guo et al. Nat Med. 2018

T cells clustered primarily based on their tissue origins and subtypes



Guo et al. Nat Med. 2018

T cell clusters CD8-C3-CX3CR1 and CD4-C3-GNLY showed the highest proportions of clonal cells



Guo et al. Nat Med. 2018

nature
medicine

ARTICLES

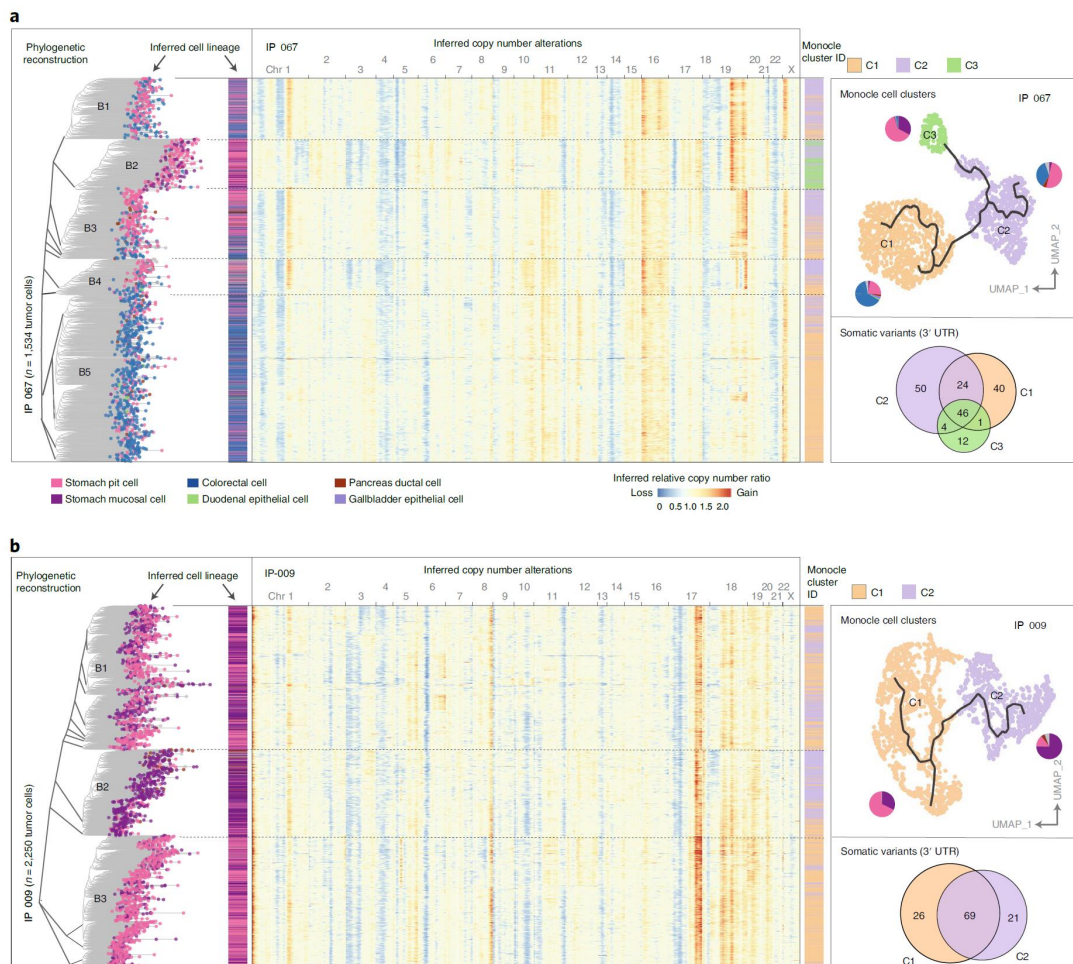
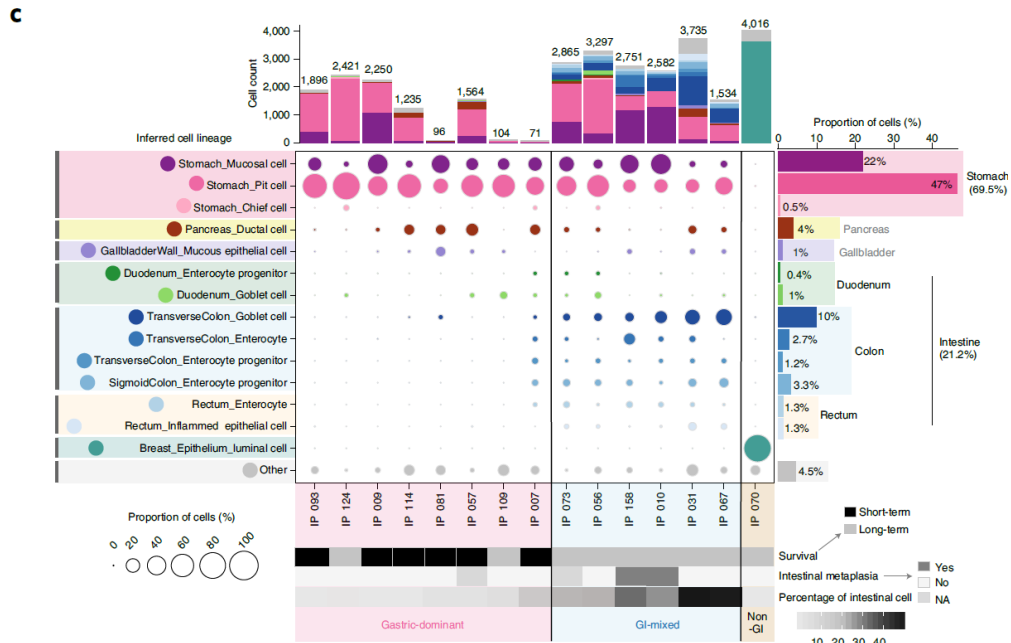
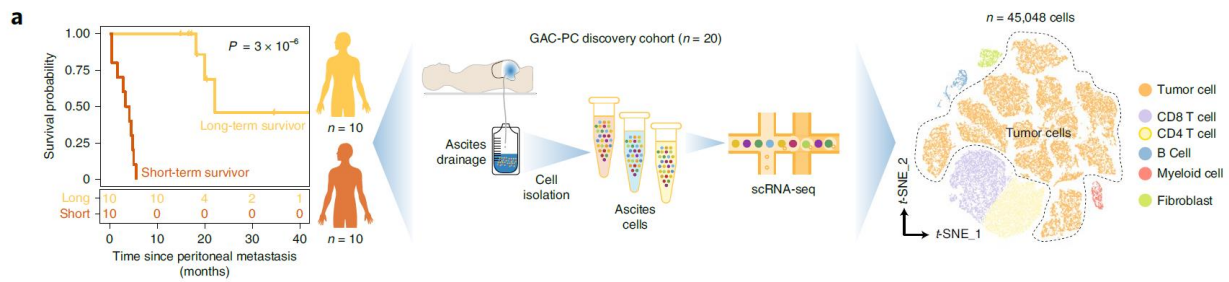
<https://doi.org/10.1038/s41591-020-1125-8>

Check for updates

Single-cell dissection of intratumoral heterogeneity and lineage diversity in metastatic gastric adenocarcinoma

Ruiping Wang¹, Minghao Dang¹, Kazuto Harada^{2,12}, Guangchun Han¹, Fang Wang³, Melissa Pool Pizzi², Meina Zhao², Ghia Tatlonghari², Shaojun Zhang¹, Dapeng Hao¹, Yang Lu⁴, Shuangtao Zhao¹, Brian D. Badgwell⁵, Mariela Blum Murphy², Namita Shanbhag², Jeannelyn S. Estrella⁶, Sinchita Roy-Chowdhuri⁶, Ahmed Adel Fouad Abdelhakeem², Yuanxin Wang¹, Guang Peng⁷, Samir Hanash⁷, George A. Calin⁸, Xingzhi Song¹, Yanshuo Chu¹, Jianhua Zhang¹, Mingyao Li⁹, Ken Chen³, Alexander J. Lazar^{6,10}, Andrew Futreal¹, Shumei Song², Jaffer A. Ajani² and Linghua Wang^{1,11}

Intratumoral heterogeneity (ITH) is a fundamental property of cancer; however, the origins of ITH remain poorly understood. We performed single-cell transcriptome profiling of peritoneal carcinomatosis (PC) from 15 patients with gastric adenocarcinoma (GAC), constructed a map of 45,048 PC cells, profiled the transcriptome states of tumor cell populations, incisively explored ITH of malignant PC cells and identified significant correlates with patient survival. The links between tumor cell lineage/state compositions and ITH were illustrated at transcriptomic, genotypic, molecular and phenotypic levels. We uncovered the diversity in tumor cell lineage/state compositions in PC specimens and defined it as a key contributor to ITH. Single-cell analysis of ITH classified PC specimens into two subtypes that were prognostically independent of clinical variables, and a 12-gene prognostic signature was derived and validated in multiple large-scale GAC cohorts. The prognostic signature appears fundamental to GAC carcinogenesis and progression and could be practical for patient stratification.



Thank you

An abstract graphic on the right side of the slide, featuring a series of curved, overlapping lines in shades of blue and purple, resembling a circuit board or a stylized wave. These lines are interspersed with small, light blue dots, creating a sense of depth and movement.

Contact

Semin Lee

Email: seminlee@unist.ac.kr

KSBi-BIML 2021

Single-cell RNA-sequencing analysis of cancer

Hyoung-oh Jeong, PhD

Email : hyoung-oh@unist.ac.kr

Ulsan National Institute of Science and Technology

Organized by

 한국생명정보학회
Korean Society for Bioinformatics

본 강의 자료는 한국생명정보학회가 주관하는 KSBi-BIML 2021 워크샵 온라인 수업을 목적으로 제작된 것으로 해당 목적 이외의 다른 용도로 사용할 수 없음을 분명하게 알립니다. 수업 목적으로 배포 및 전송 받은 경우에도 이를 다른 사람과 공유하거나 복제, 배포, 전송할 수 없습니다.

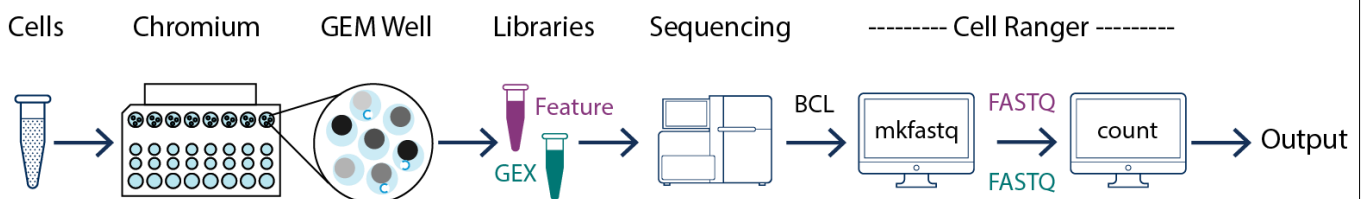
만약 이러한 사항을 위반할 경우 발생하는 모든 법적 책임은 전적으로 불법 행위자 본인에게 있음을 경고합니다.

 한국생명정보학회
Korean Society for Bioinformatics

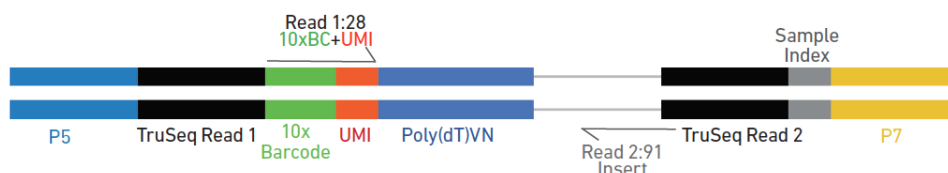
Contents

1. Identify doublets (Scrublet)
2. Remove cell free mRNA contamination (SoupX)
3. single-cell RNA-seq analysis (Seurat)
4. Cell type annotation (SingleR)
5. Data Integration (Seurat)
6. Inferring copy number alterations (InferCNV)
7. Trajectory analysis (monocle)
8. Cell-cell communication (CellPhoneDB)
9. Gene Regulatory Networks (SCENIC)

Generating FASTQ



Library1_S1_L001_R1_001.fastq.gz
Library1_S1_L001_R2_001.fastq.gz
Library1_S1_L001_I1_001.fastq.gz



```
@ST-E00104:1062:H2NV3CCX2:2:1101:4980:1959 2:N:0:ACAGAGGT
NCACCTTCTGTTTTCGGATTGAAGAAGATGTACATTTTTGTCAACTCGTACTTTATCAGATAAGCAACTCACGTATTTGGATCTTTATT
+
#AA-A-AAAJJJJ-7JF<FFJ-7FA-AFJJJJJ--<<AFJ-<FJ<AA--F7A7<-A<-A--7--7A-<7---77-A<<---7AFA---
@ST-E00104:1062:H2NV3CCX2:2:1101:5446:1959 2:N:0:ACAGAGGT
NTCCCGCCCGCCGCCCTCTTAGAGACTCGGGTCTTCTGTTCCACACGTCGGTTTCGGTGACCGATATTGTTGTCACTGCTGCGGGGTAAG
+
#AA-AFAJJJ<-AJF<JF-7A--7<<7F--7A-77---<-F<-<FJ-J<-FJAFA-7A<-A-A-----<-7-7-----<F<7A-----7-
@ST-E00104:1062:H2NV3CCX2:2:1101:5507:1959 2:N:0:ACAGAGGT
NATGAATAAGAGGTGGACACACAGCATGCTCCGGCAGCACGGCGTGGTGTGTCCTCCGACACATCCCTTCATTCCATGGACTTAGAGGCG
+
#-AAA7FJJJ<JFJ7FJJ7<J-FJAJJ-<FFFJ-77FAFJJJ<AAAJAA-F-FJ-FJA-FAA-A7A-77AA-A-<7F<-F7A-A<F<<FF-
```

Cell Ranger



Products Research Areas Resources **Support** Company

Support > Single Cell Gene Expression > Software

SEARCH

Q&A

NEW

CONTACT SUPPORT

SOFTWARE > DOWNLOADS

CELL RANGER

Introduction

Downloads

Tutorials

Running Pipelines

Understanding Outputs

Algorithms Overview

Advanced

LOUPE

Introduction

Download

Tutorial

Software Downloads



Cell Ranger 5.0.1

Single Cell Analysis Pipelines



Loupe Browser 5.0.0

Interactive Analysis

Please follow the [install instructions](#) after downloading the Cell Ranger package below.

Cell Ranger - 5.0.1 (December 16, 2020)

- Self-contained, relocatable tar file. Does not require centralized installation.
- Contains binaries pre-compiled for CentOS/RedHat 6.0+ and Ubuntu 12.04+.
- [Download - Linux 64-bit - 955 MB](#) - md5sum: 8b9d217c160d52902ebcde8608765119

curl

wget

<https://support.10xgenomics.com/single-cell-gene-expression/software/downloads/latest>



References - 2020-A (July 7, 2020)

- Human reference (GRCh38) dataset required for Cell Ranger.
- [Download - 11 GB](#) - md5sum: dfd654de39bff23917471e7fcc7a00cd
- [Build steps](#)

curl

wget

```
curl -O https://cf.10xgenomics.com/supp/cell-exp/refdata-gex-GRCh38-2020-A.tar.gz
```

- Mouse reference dataset required for Cell Ranger.
- [Download - 9.7 GB](#) - md5sum: 886eeddde8731ffb58552d0bb81f533d
- [Build steps](#)

curl

wget

```
curl -O https://cf.10xgenomics.com/supp/cell-exp/refdata-gex-mm10-2020-A.tar.gz
```

- Human reference (GRCh38) and mouse dataset required for Cell Ranger.
- [Download - 9.9 GB](#) - md5sum: c34194518ef19a4d0c409b598bb7363e
- [Build steps](#)

curl

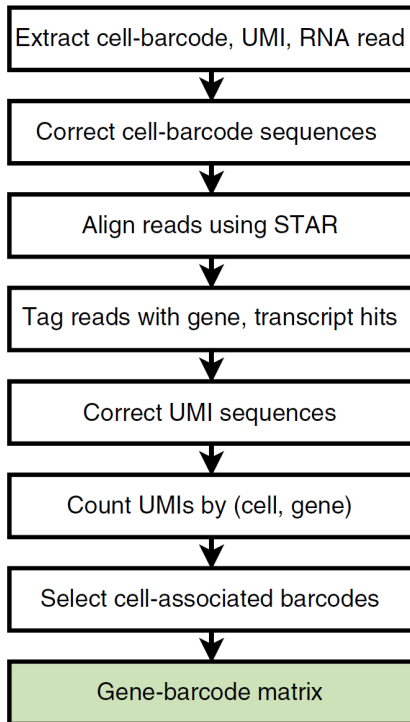
wget

```
curl -O https://cf.10xgenomics.com/supp/cell-exp/refdata-gex-GRCh38-and-mm10-2020-A.tar.gz
```

<https://support.10xgenomics.com/single-cell-gene-expression/software/downloads/latest>



Cell Ranger pipeline workflow



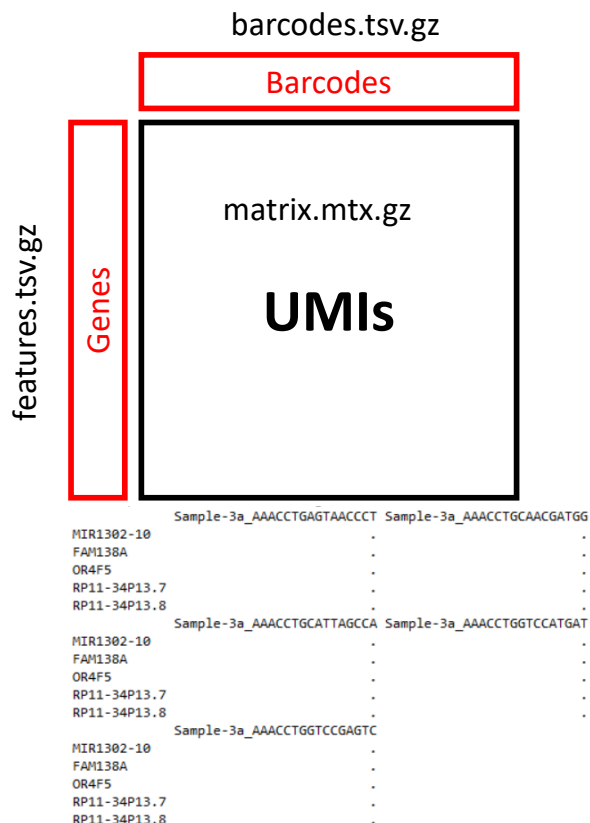
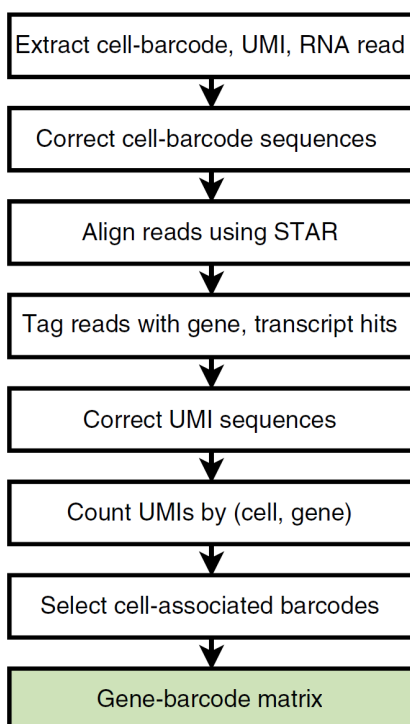
```

$ cellranger count \
--id=sample-1a \
--transcriptome=refdata-cellranger-hg19-1.2.0 \
--fastqs=sample-1a/ \
--sample=sample-1a \
--localcores 20 \
--localmem 24
    
```

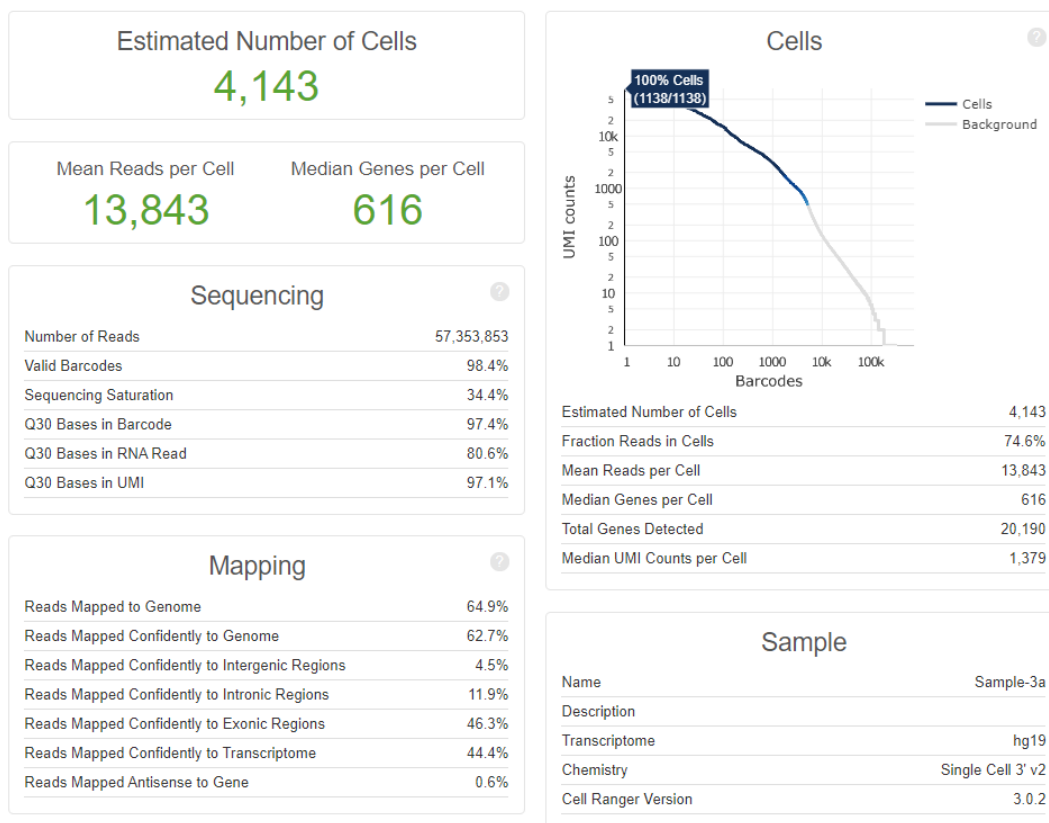
Output directory
[OUTPUTDIR] /outs/filtered_feature_bc_matrix/

1. barcodes.tsv.gz
2. features.tsv.gz
3. matrix.mtx.gz

Cell Ranger pipeline workflow



[OUTPUTDIR] /outs/web_summary.html



Sign in to RStudio

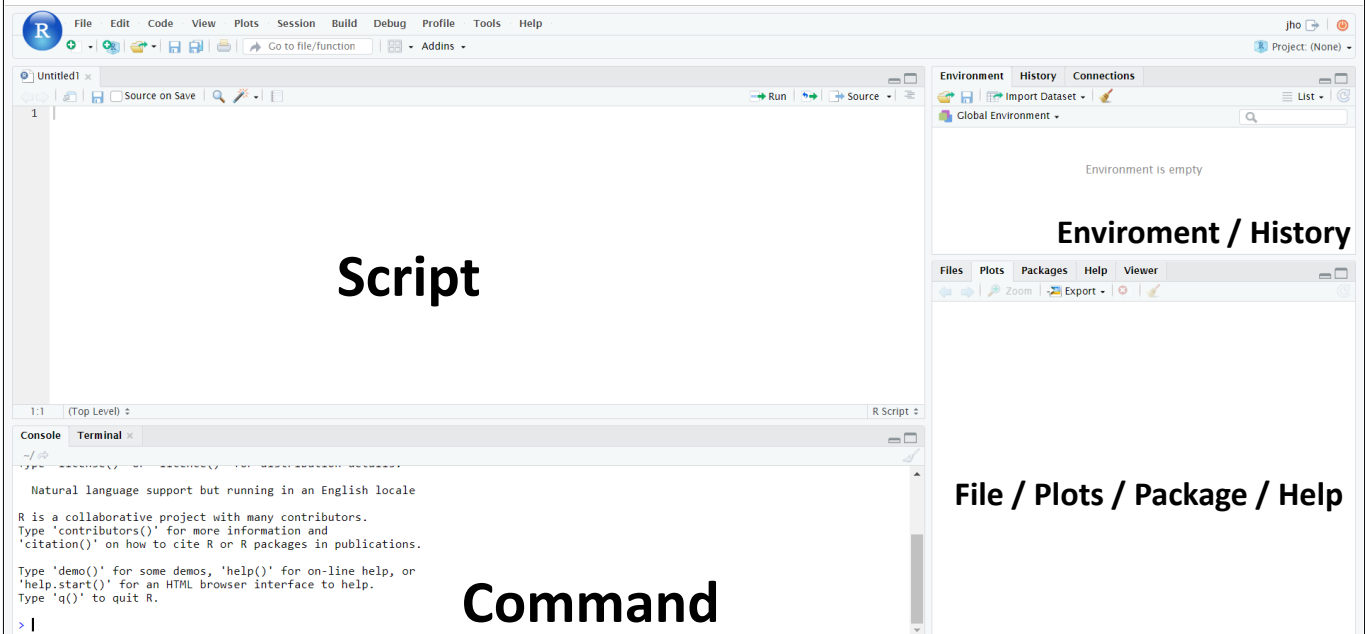
Username:

Password:

☐ Stay signed in

Sign In

Rstudio



Test dataset

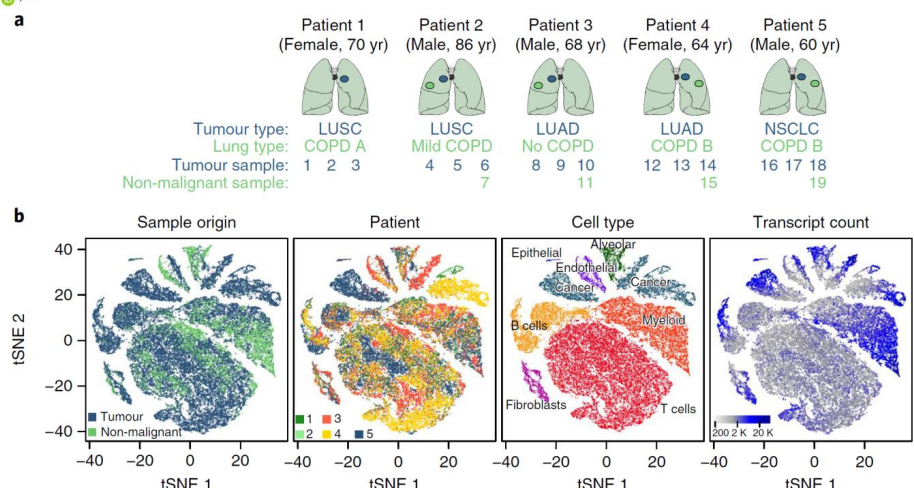


Phenotype molding of stromal cells in the lung tumor microenvironment

Diether Lambrechts^{1,2*}, Els Wauters^{3,4}, Bram Boeckx^{1,2}, Sara Aibar^{5,6}, David Nittner^{7,8}, Oliver Burton^{6,9}, Ayse Bassez^{1,2}, Herbert Decaluwé^{10,11}, Andreas Pircher¹², Kathleen Van den Eynde¹³, Birgit Weynand¹³, Erik Verbeken¹³, Paul De Leyn¹¹, Adrian Liston^{6,9}, Johan Vansteenkiste^{3,4}, Peter Carmeliet^{1,12,14}, Stein Aerts^{5,6} and Bernard Thienpont^{1,15*}

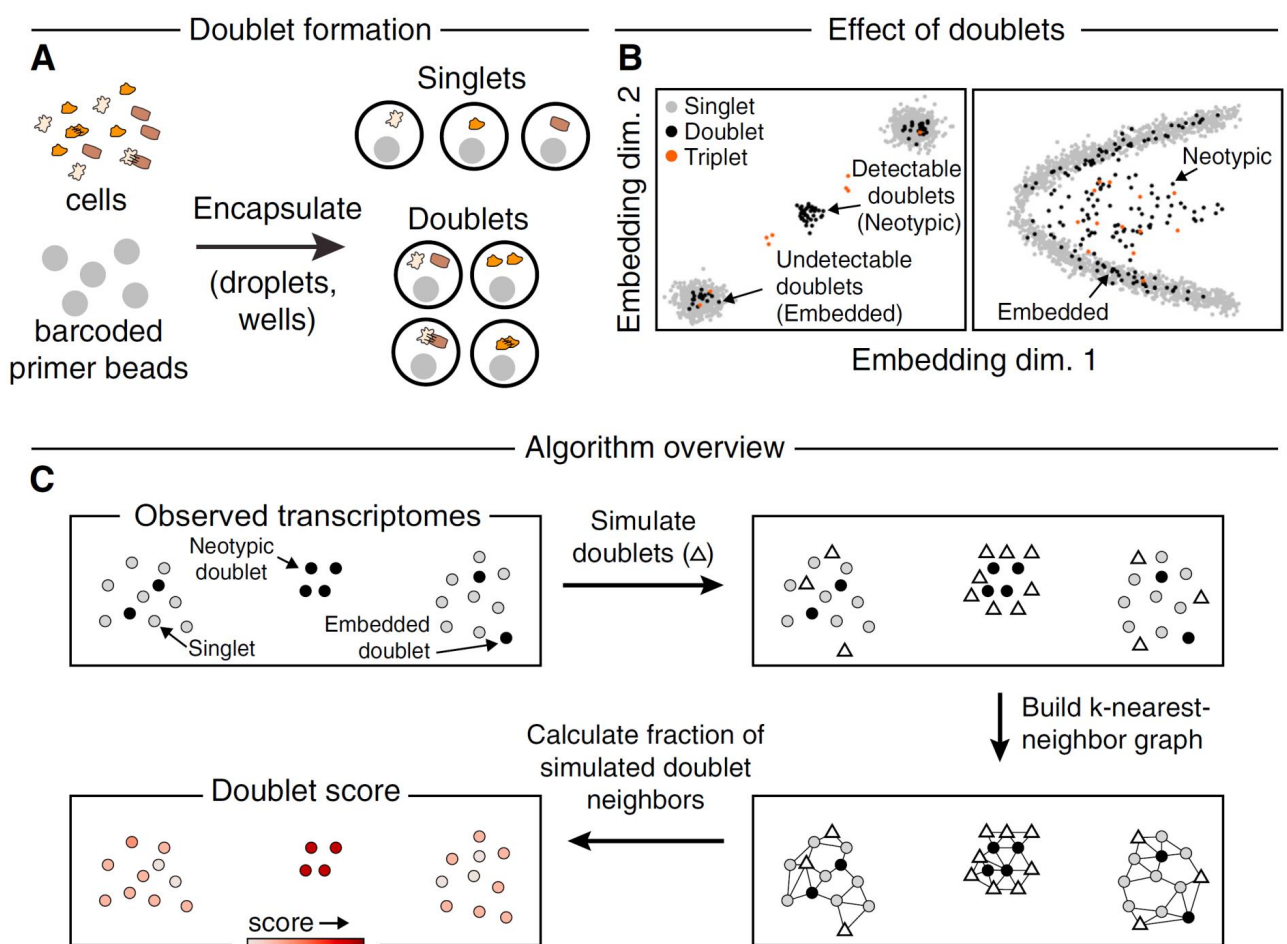
- **52,698** single cells from **lung tumors** and distal non-malignant lung samples

- Input dataset :
 ~/Resource/RawData/Leuven/filtered_feature_bc_matrix



Identify doublets (Scrublet)

<https://github.com/AllonKleinLab/scrublet>



Load counts matrix and gene list (Python)

- `import sys`
- `import os`
- `import scrublet as scr`
- `import scipy.io`
- `import numpy as np`
- `import pandas as pd`

- `counts_matrix = scipy.io.mmread(input_dir + '/matrix.mtx').T.tocsc()`
- `genes = np.array(scr.load_genes(input_dir + '/features.tsv', delimiter = '\t', column = 1))`
- `barcodes = np.loadtxt(input_dir + '/barcodes.tsv', dtype = 'str')`

Initialize Scrublet object (Python)

- `scrub = scr.Scrublet(counts_matrix, expected_doublet_rate = 0.06)`
 - *expected_doublet_rate*: the expected fraction of transcriptomes that are doublets, typically 0.05-0.1.
 - *sim_doublet_ratio*: the number of doublets to simulate, relative to the number of observed transcriptomes. This should be high enough that all doublet states are well-represented by simulated doublets. (default = 2)
 - *n_neighbors*: Number of neighbors used to construct the KNN classifier of observed transcriptomes and simulated doublets. (default = `round(0.5*sqrt(n_cells))`)

Run the default pipeline (Python)

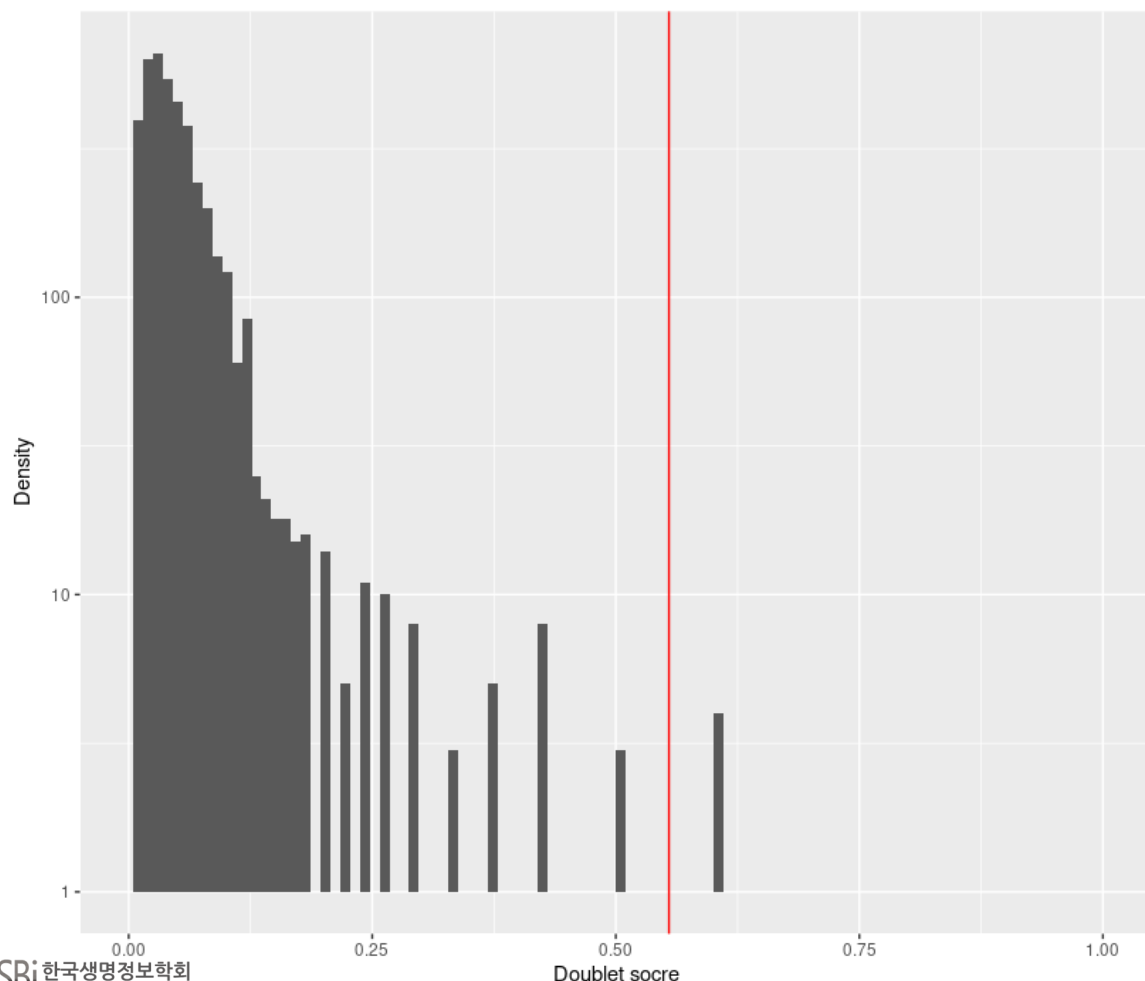
- `doublet_scores, predicted_doublets =`
`scrub.scrub_doublets(min_counts = 2, min_cells = 3,`
`min_gene_variability_pctl = 85, n_prin_comps = 30)`
- *min_gene_variability_pctl* : Keep the most highly variable genes (default: 85.0)
 - *n_prin_comps* : Number of principal components used to embed the transcriptomes prior to k-nearest-neighbor graph construction.

Save result (Python)

- `scrub_obs = pd.DataFrame({ "barcodes" : barcodes,`
`"doublet_scores_obs" : scrub.doublet_scores_obs_, "threshold" :`
`scrub.threshold_})`
- `scrub_sim = pd.DataFrame({ "doublet_scores_sim" :`
`scrub.doublet_scores_sim_, "threshold" : scrub.threshold_})`
- `scrub_obs.to_csv(out_dir + "/scrublet.doublet_scores_obs.txt", index =`
`False, header = None, sep = "\t")`
- `scrub_sim.to_csv(out_dir + "/scrublet.doublet_scores_sim.txt", index =`
`False, header = None, sep = "\t")`

Plot doublet score histograms (R)

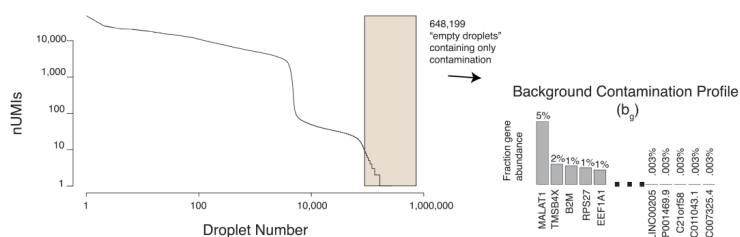
- `db.score <- read.table(file = paste0(outdir,`
 `"/scrublet.doublet_scores_obs.txt"), header = F, sep = "\t")`
- `colnames(db.score) <- c("barcodes", "scores", "threshold")`
- `db.score$barcodes <- do.call(rbind,`
 `strsplit(as.character(db.score$barcodes), split = "\\-"))[,1]`
- `ggplot(data = db.score, aes(x = scores)) + geom_histogram(bins = 100) +`
 `xlim(0,1) + xlab("Doublet socre") + ylab("Density") +`
 `scale_y_continuous(trans = 'log10') + geom_vline(xintercept =`
 `unique(db.score$threshold), color = "red")`



Remove cell-free mRNA contamination (SoupX)

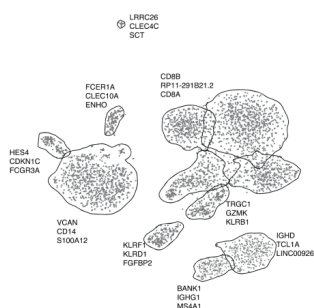
<https://github.com/constantAmateur/SoupX>

1. Determine the expression profile of contamination

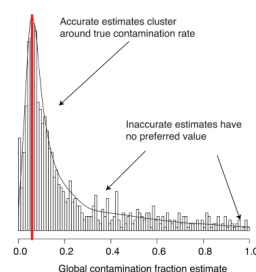


2. Estimate or set the global contamination rate

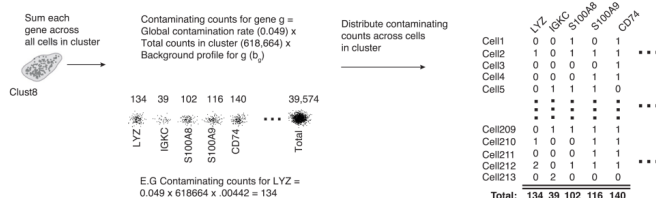
2.1 Marker genes for each cluster identified



2.2 Set contamination to most common estimate



3. Remove contamination from cells one cluster at a time



Load counts matrix and gene list

- library(SoupX)
- library(DropletUtils)
- cellranger.dir <- "~/BIML-2021-SingleCellRNAseq/Resource/RawData/Leuven"
- outdir <- "~/BIML-2021-SingleCellRNAseq/Result/SoupX"
- sc <- load10X(cellranger.dir)

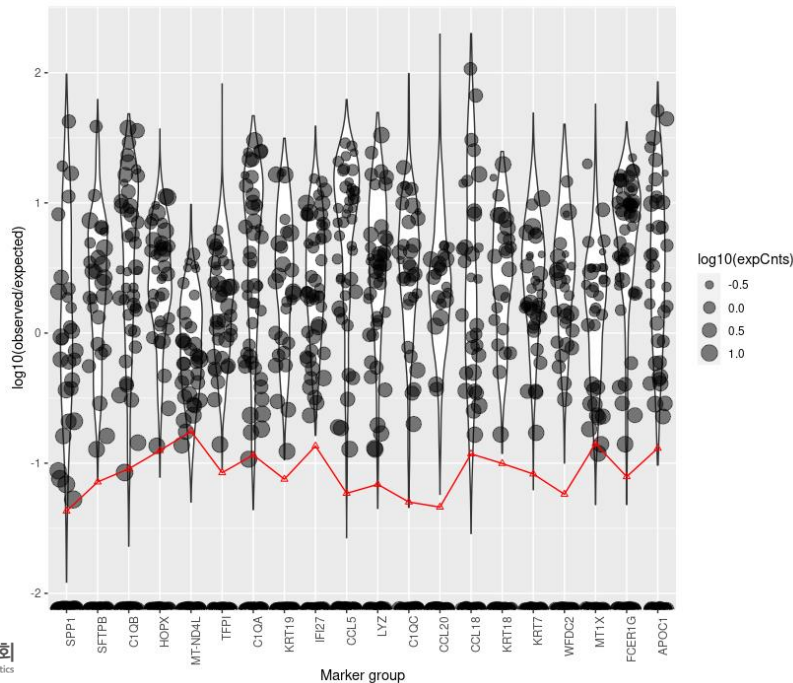
Genes to estimate the contamination fraction

- head(sc\$soupProfile[order(sc\$soupProfile\$est, decreasing = TRUE),], n = 20)

	est	counts
MT-CO3	0.074241866	167989
MT-CO1	0.062408352	141213
MT-ND2	0.053987535	122159
MT-CO2	0.046211075	104563
MT-ND4	0.043158562	97656
MT-CYB	0.033867114	76632
MT-ATP6	0.029836578	67512
MT-ND1	0.027338706	61860
MALAT1	0.026268315	59438
MT-ND3	0.018138299	41042
TMSB4X	0.016731588	37859
FTL	0.011879476	26880
B2M	0.007407437	16761
FTH1	0.007134757	16144
PTMA	0.005107556	11557
RPL41	0.004816314	10898
RPL10	0.004653679	10530
RPS18	0.004545844	10286
RPLP1	0.004410167	9979
EEF1A1	0.004136161	9359

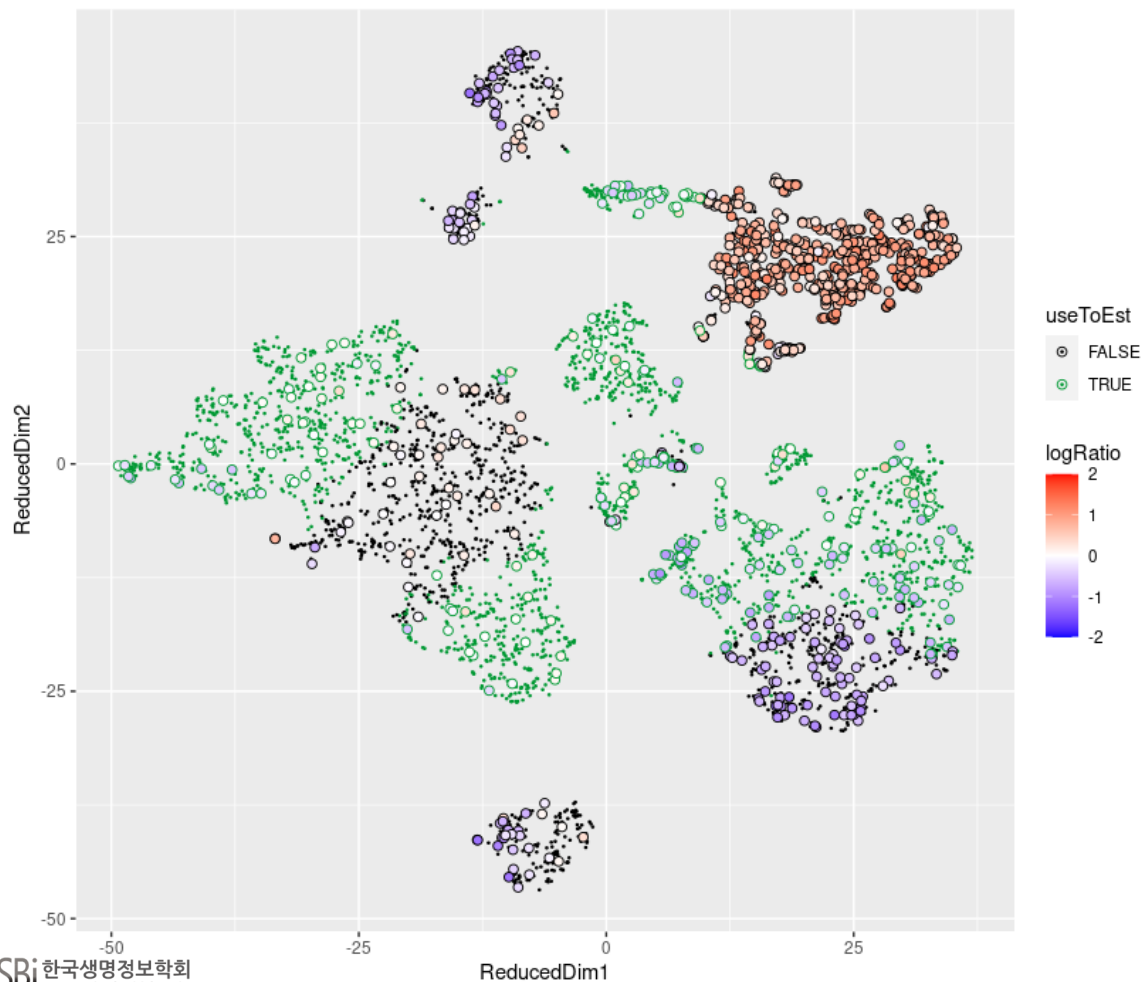
Genes to estimate the contamination fraction

➤ plotMarkerDistribution(sc)



Estimating non-expressing cells

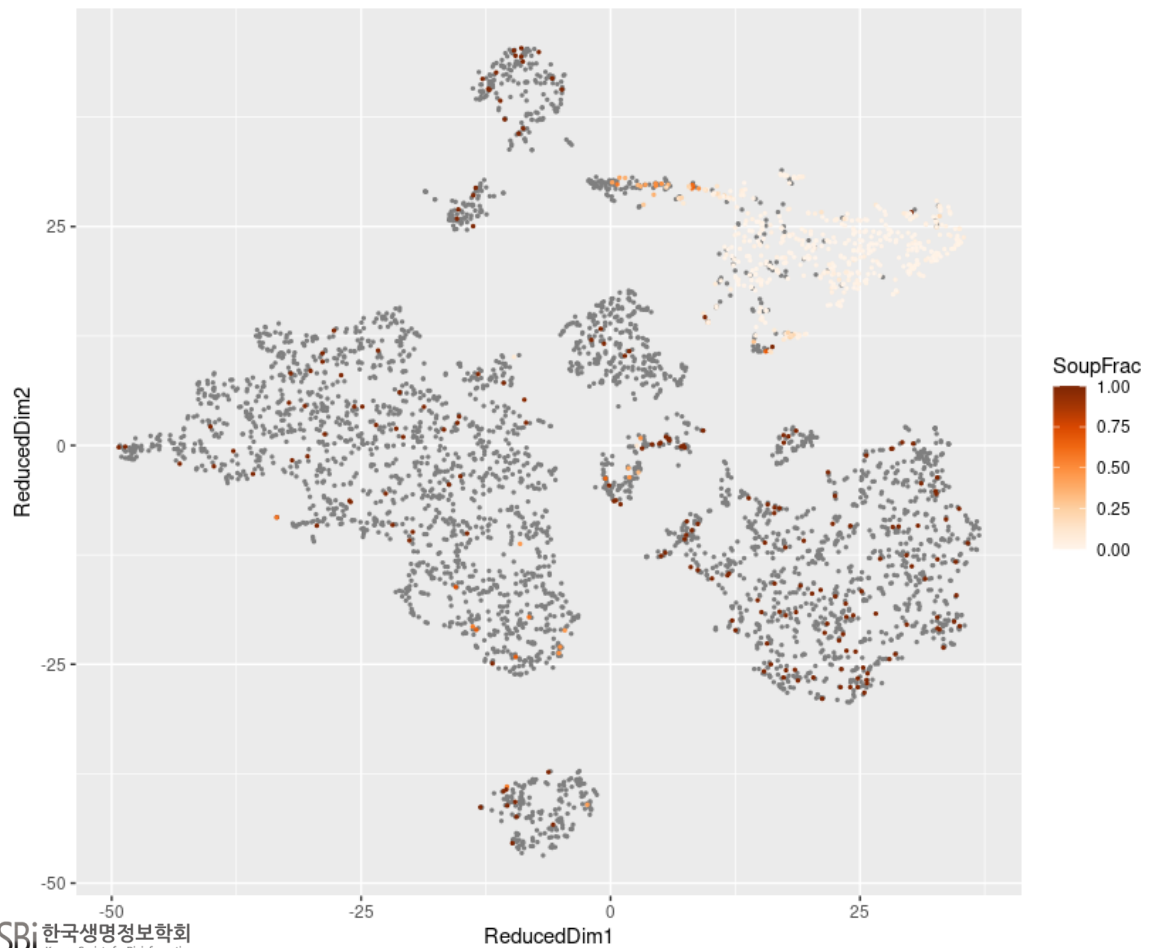
- `Genes <- c("KRT17", "KRT18", "KRT19")`
- `useToEst <- estimateNonExpressingCells(sc,
nonExpressedGeneList = list(KRT = Genes))`
- `plotMarkerMap(sc, geneSet = Genes, useToEst =
useToEst)`



Calculating the contamination fraction

- `sc <- calculateContaminationFraction(sc, list(KRT = Genes), useToEst = useToEst)`
- `out <- adjustCounts(sc)`
- `write10xCounts(path = paste0(outdir, "/strainedCounts"), x = out)`
- `plotChangeMap(sc, out, "KRT19")`

Change in expression due to soup correction



The automated method

- `sc <- load10X(cellranger.dir)`
- `sc <- autoEstCont(sc)`
- `out <- adjustCounts(sc)`

scRNA-seq analysis (Seurat)

<https://satijalab.org/seurat/index.html>

Seurat 4.0.0 Install Get started Vignettes ▾ Extensions FAQ News Reference Archive



Official release of Seurat 4.0

We are excited to release Seurat v4.0! This update brings the following new features and functionality:

- **Integrative multimodal analysis.** The ability to make simultaneous measurements of multiple data types from the same cell, known as multimodal analysis, represents a new and exciting frontier for single-cell genomics. In Seurat v4, we introduce weighted nearest neighbor (WNN) analysis, an unsupervised strategy to learn the information content of each modality in each cell, and to define cellular state based on a weighted combination of both modalities. In our new preprint, we generate a CITE-seq dataset featuring paired measurements of the transcriptome and 228 surface proteins, and leverage WNN to define a multimodal reference of human PBMC. You can use WNN to analyze multimodal data from a variety of technologies, including CITE-seq, ASAP-seq, 10X Genomics ATAC + RNA, and SHARE-seq.
 - Preprint: [Integrated analysis of multimodal single-cell data](#)
 - Vignette: [Multimodal clustering of a human bone marrow CITE-seq dataset](#)
 - Portal: [Click here](#)
 - Dataset: [Download here](#)
- **Rapid mapping of query datasets to references.** We introduce Azimuth, a workflow to leverage high-quality reference datasets to rapidly map new scRNA-seq datasets (queries). For example, you can map any scRNA-seq dataset of human

Links

Download from CRAN at
<https://cloud.r-project.org/package=Seurat>

Browse source code at
<https://github.com/satijalab/seurat/>

Report a bug at
<https://github.com/satijalab/seurat/issues>

License

[GPL-3](#) | file [LICENSE](#)

Community

[Code of conduct](#)

Citation

[Citing Seurat](#)

Developers

Paul Hoffman

Author, maintainer 

Satija Lab and Collaborators

Funder

[All authors...](#)

<https://satijalab.org/seurat/>

Load raw count data

- `library(Seurat)`
- `library(dplyr)`
- `raw.data <- Read10X(data.dir = "~/BIML-2021-SingleCellRNAseq/Result/SoupX/strainedCounts")`

Create A Seurat Object

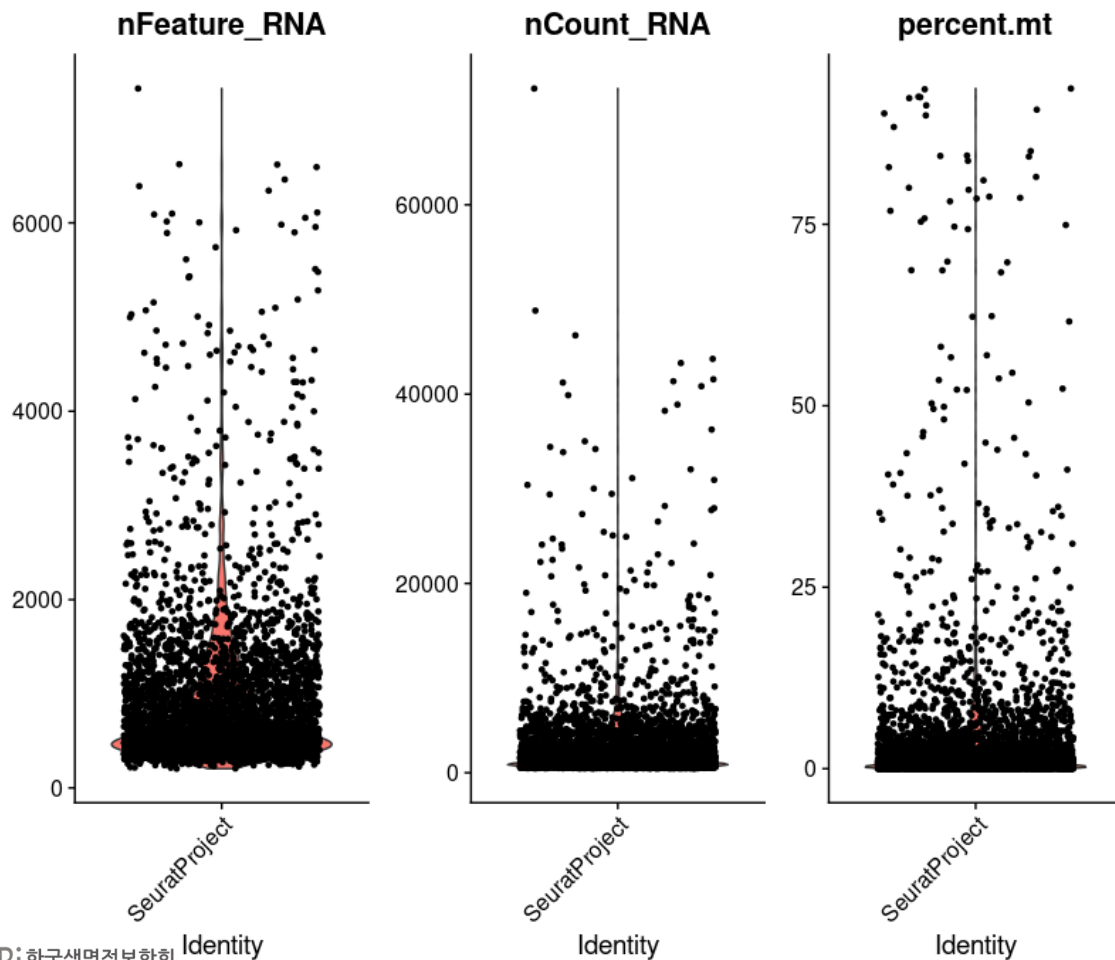
- `seurat.obj <- CreateSeuratObject(counts = raw.data, min.cells = 3, min.features = 200)`
 - **counts** : Unnormalized data such as raw counts or TPMs
 - **min.cells** : Include features detected in at least this many cells
 - **min.features** : Include cells where at least this many features are detected

QC metrics

- The number of unique genes detected in each cell.
 - Low-quality cells or empty **droplets** will often have **very few genes**
 - Cell **doublets** or multiplets may exhibit an **aberrantly high gene count**
- Similarly, the total number of molecules detected within a cell (correlates strongly with unique genes)
- The percentage of reads that map to the mitochondrial genome
 - **Low-quality / dying cells** often exhibit **extensive mitochondrial contamination**

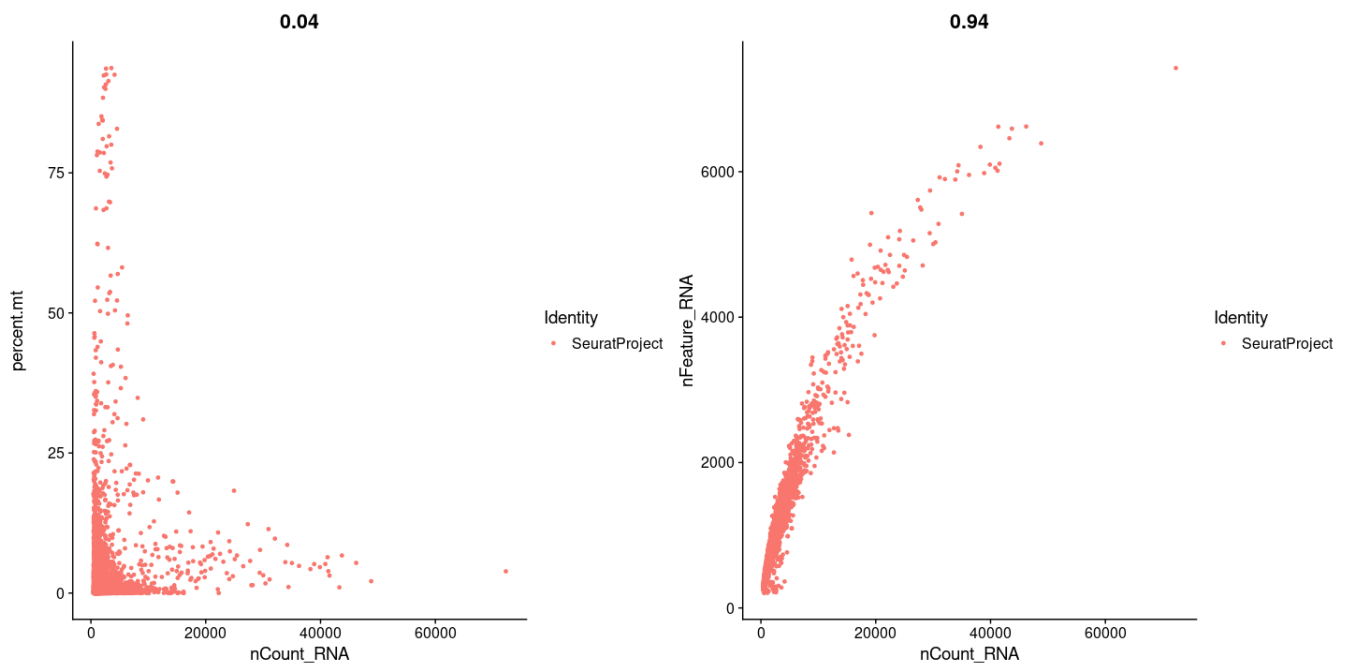
QC metrics

```
➤ seurat.obj[["percent.mt"]] <-  
  PercentageFeatureSet(seurat.obj, pattern = "^MT-")  
  
➤ >VlnPlot(object = seurat.obj, features =  
  c("nFeature_RNA", "nCount_RNA", "percent.mt"),  
  ncol = 3)
```



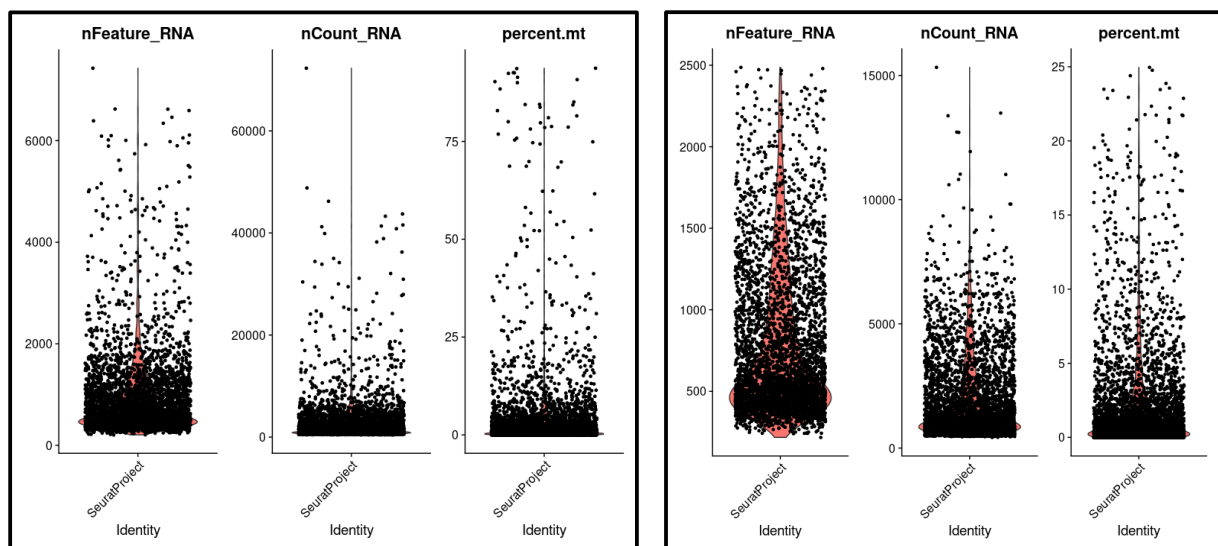
QC metrics

- `seurat.feature.plot1 <- FeatureScatter(object = seurat.obj, feature1 = "nCount_RNA", feature2 = "percent.mt")`
- `seurat.feature.plot2 <- FeatureScatter(object = seurat.obj, feature1 = "nCount_RNA", feature2 = "nFeature_RNA")`
- `CombinePlots(plots = list(seurat.feature.plot1, seurat.feature.plot2))`



Filter low-quality cells

➤ `seurat.obj <- subset(x = seurat.obj, subset =
nFeature_RNA > 200 & nFeature_RNA < 2500 &
percent.mt < 5)`



Filtered doublet

- `doublet.file <- "~/BIML-2021-SingleCellRNAseq/Result/Scrublet/scrublet.doublet_scores_obs.txt"`
- `doublet.df <- read.table(doublet.file , header = F, sep = "\t")`
- `colnames(doublet.df) <- c("barcodes", "c", "threshold")`
- `doublet.df$Doublet <- "singlet"`
- `doublet.df$Doublet[doublet.df$scores > doublet.df$threshold] <- "doublet"`

Filtered doublet

- `doublet.df <- join(data.frame(barcodes = rownames(seurat.obj@meta.data), seurat.obj@meta.data), doublet.df, by = "barcodes")`
- `seurat.obj@meta.data$Doublet <- doublet.df$Doublet`
- `seurat.obj <- subset(x = seurat.obj, subset = Doublet == "singlet")`

Normalizing the data

```
➤ seurat.obj <- NormalizeData(object = seurat.obj, normalization.method = "LogNormalize")
```

- ***normalization.method*** : Method for normalization.
 - **LogNormalize**: Feature counts for each cell are divided by the total counts for that cell and multiplied by the `scale.factor`. This is then natural-log transformed using `log1p`.
 - **CLR**: Applies a centered log ratio transformation
 - **RC**: Relative counts. Feature counts for each cell are divided by the total counts for that cell and multiplied by the `scale.factor`. No log-transformation is applied. For counts per million (CPM) set `scale.factor = 1e6`
- ***scale.factor*** : Sets the scale factor for cell-level normalization (default : 10000)

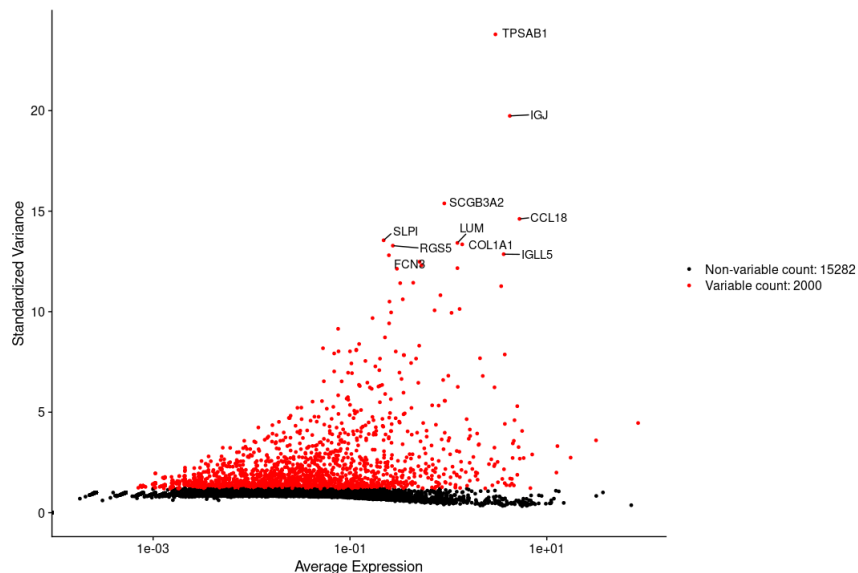
Identification of highly variable features

```
➤ seurat.obj <- FindVariableFeatures(object = seurat.obj,  
selection.method = "vst", nfeatures = 2000)
```

- ***selection.method*** : How to choose top variable features
 - **vst** : Fits a line to the relationship of $\log(\text{variance})$ and $\log(\text{mean})$ using local polynomial regression (loess). Feature variance is then calculated on the standardized values after clipping to a maximum
 - **mean.var.plot (mvp)** : Divides features into `num.bin` (default 20) bins based on their average expression, and calculates z-scores for dispersion within each bin
 - **dispersion (disp)** : selects the genes with the highest dispersion values
- ***nfeatures*** : Number of features to select as top variable features; only used when `selection.method` is set to '*dispersion*' or '*vst*'

Identification of highly variable features

- `v.genes <- VariableFeatures(object = seurat.obj)`
- `LabelPoints(plot = VariableFeaturePlot(seurat.obj), points = v.genes[1:10], repel = T)`



Scaling the data

- `seurat.obj <- ScaleData(object = seurat.obj, features = rownames(seurat.obj), vars.to.regress = "percent.mt")`
 - **features** : Vector of features names to scale/center (Default : all features)
 - **vars.to.regress** : Variables to regress out. For example, nUMI, or percent.mito. (Optional)
 - **model.use** : Use a linear model or generalized linear model for the regression Options are 'linear', 'poisson', and 'negbinom' (default : linear)

Dimensional reduction

➤ `seurat.obj <- RunPCA(object = seurat.obj, features = v.genes)`

- ***npcs*** : Total Number of PCs to compute and store (default : 50)
- ***features*** : Features to compute PCA on

```
PC_1
Positive: TYROBP, CD74, HLA-DPB1, FCER1G, HLA-DRA, AIF1, ALOX5AP, HLA-DPA1, LYZ, SRGN
          C1orf162, HLA-DRB1, HLA-DQA1, CD68, C1QA, C1QB, HLA-DQB1, MS4A6A, MS4A4A, CTSS
          CAPG, MS4A7, C1QC, FCGR3A, RNASE6, LST1, HLA-DRB5, CD14, APOC1, HLA-DMB
Negative: SPARC, DCN, CALD1, COL1A2, BGN, CIR, COL6A2, SPARCL1, COL3A1, RARRES2
          C1S, MGP, IGFBP7, LUM, COL1A1, A2M, NNMT, COL6A3, PCOLCE, MYL9
          THY1, COL6A1, TPM2, MMP2, IGFBP4, COL5A2, MFAP4, SFRP2, AEBP1, TAGLN

PC_2
Positive: CCL5, IL32, CD7, GZMA, GZMB, NKG7, PRF1, CTSW, CCL4, KLRB1
          CD69, CD27, HOPX, TIGIT, GNLY, CD8A, ITM2A, IFNG, CXCL13, GZMH
          KLRD1, TNFRSF18, LTB, GZMK, CD8B, BIRC3, KRT19, KRT18, SFTPB, KRT7
Negative: CST3, CD68, LYZ, AIF1, FCER1G, TYROBP, CTSB, FTL, CTSI, MS4A4A
          GPNMB, LGALS1, C1QA, C1QB, TIMP1, PSAP, GLUL, GPX1, MS4A7, GRN
          APOC1, FTH1, FCGRT, C1QC, CD14, MS4A6A, APOE, C1orf162, TMEM176B, CTSS

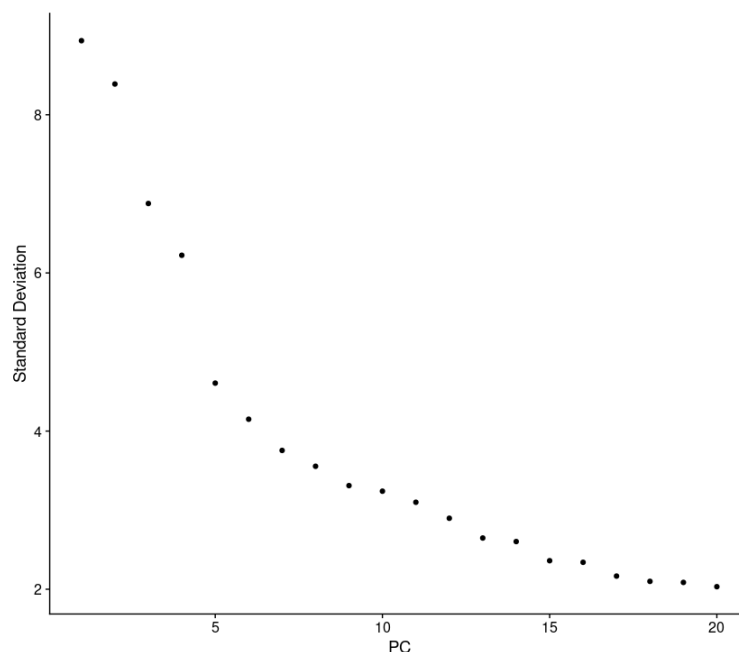
PC_3
Positive: KRT19, KRT18, KRT7, WFDC2, SFTPB, AGR2, SLC34A2, CCND1, KRT8, EPCAM
          SPINT2, MALL, SFTA2, TACSTD2, UBE2C, S100A14, HOPX, MGST1, CLDN4, HMGB3
          ERRFI1, NAPS, PAEP, GPRC5A, TFPI, AQP3, SDR16C5, SPINK1, LSR, NPC2
Negative: CCL5, IL32, COL1A2, SRGN, DCN, COL3A1, GZMA, COL6A2, CD7, LUM
          NKG7, GZMB, RARRES2, SFRP2, COL5A2, CCL4, COL6A3, PCOLCE, CTSW, BGN
          C1S, CD69, RGS1, CALD1, MFAP4, COL14A1, PRF1, AEBP1, TPM2, IGF1

PC_4
Positive: RAMP2, CLEC14A, CLDN5, PLVAP, VWF, ECSCR, RAMP3, PCDH17, ELTD1, EGFL7
          CALCRL, DARC, HSPG2, EMCN, FCN3, TSPAN7, AQP1, ESAM, GNG11, HYAL2
          CCL14, CYR1, CD34, CD93, JAM2, FAM167B, ARHGAP29, LDB2, CDH5, SPARCL1
          COL1A1, LUM, DCN, RARRES2, COL3A1, COL1A2, SFRP2, COL6A3, SERPINF1, C1S
          MFAP4, FBLN1, IGF1, PCOLCE, CCDC80, COL14A1, NBL1, COL5A2, FGF7, COL5A1
          MOXD1, COL8A1, MXRA8, RARRES1, MEG3, COL6A1, ISLR, AEBP1, FMO, OLFML3

PC_5
Positive: MMP2, SFRP2, IGF1, PTGDS, LUM, SERPINF1, MOXD1, POSTN, FGF7, FBLN1
          C7, CCDC80, COL8A1, LXN, RARRES1, LSAMP, COL1A1, ISLR, FBLN2, CTHRC1
          PLA2G2A, DPYSL3, MFAP4, COL16A1, VCAN, EFEMP1, CXCL12, NBL1, C3, IGFBP4
          HIGD1B, NDUFA4L2, COX4I2, RGS5, PTN, PPP1R14A, GJA4, PDGFRB, EGFL6, LHFP
          CSRP2, KCNJ8, NOTCH3, FOXS1, TPPP3, SEPT4, FAM162B, MAP1B, NR2F2, SMO,
          KCNK3, GJC1, MYO1B, PTP4A3, CCDC102B, KCNK17, FRZB, TINAGL1, TBX2, LAMC3
```

Determine the 'dimensionality'

➤ `ElbowPlot(object = seurat.obj)`

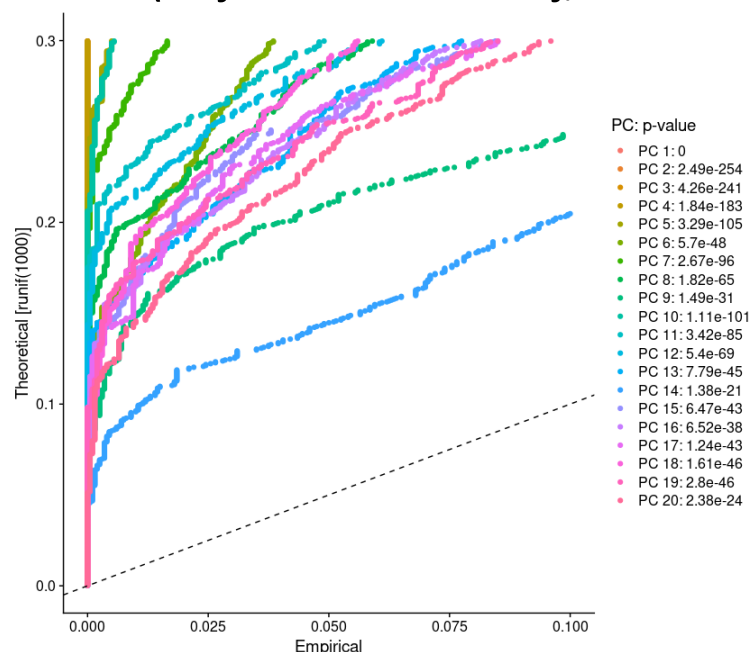


Determine the 'dimensionality'

- `seurat.obj <- JackStraw(object = seurat.obj, num.replicate = 100)`
 - ***dims*** : Number of PCs to compute significance for
 - ***num.replicate*** : Number of replicate samplings to perform
- `JackStrawPlot(object = seurat.obj, dims = 1:20)`
 - ***dims*** : Which dimensions to examine

Determine the 'dimensionality'

- `JackStrawPlot(object = seurat.obj, dims = 1:20)`



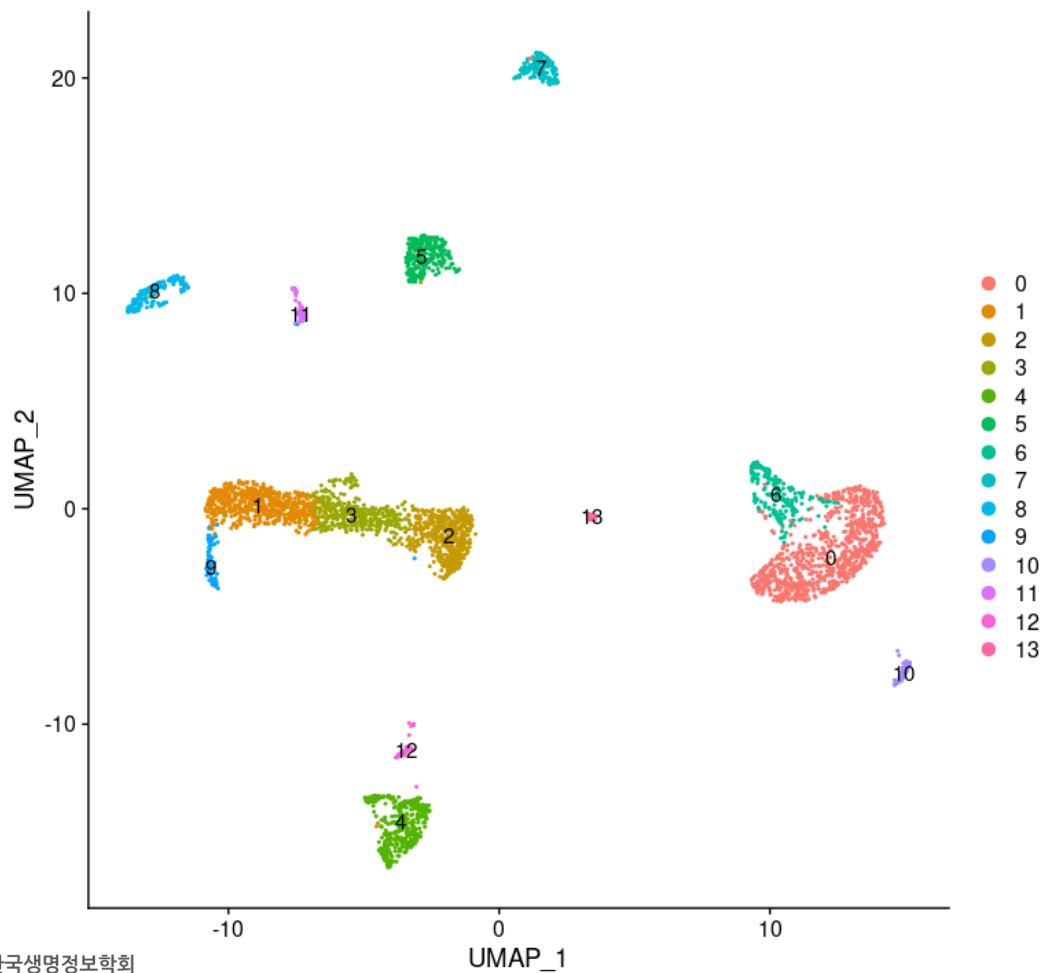
Clustering analysis

- `seurat.obj <- FindNeighbors(object = seurat.obj, dims = 1:17)`
- `seurat.obj <- FindClusters(object = seurat.obj, resolution = 0.5)`

Run non-linear dimensional reduction

Option 1. Run UMAP

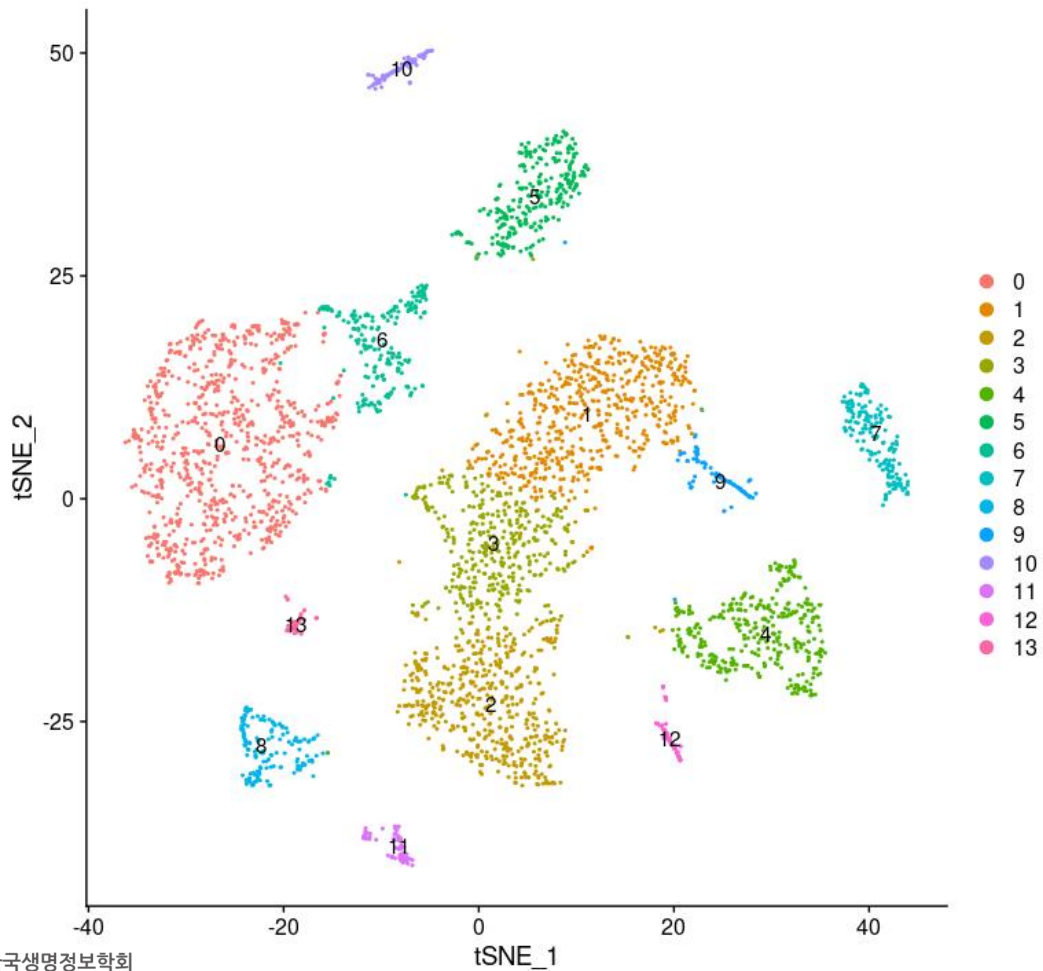
- `seurat.obj <- RunUMAP(object = seurat.obj, dims = 1:17)`
- `DimPlot(object = seurat.obj, reduction = "umap")`



Run non-linear dimensional reduction

Option 2. T-Distributed Stochastic Neighbor Embedding (tSNE)

- `seurat.obj <- RunTSNE(object = seurat.obj, dims = 1:17)`
- `DimPlot(object = seurat.obj, reduction = "tsne")`



Save Seurat object

➤ `saveRDS(object = seurat.obj, file = "~/BIML-2021-SingleCellRNAseq/Result/Seurat/Seurat.RDS")`

Finding differentially expressed features

➤ `seurat.markers <- FindAllMarkers(object = seurat.obj, only.pos = T, min.pct = 0.25, logfc.threshold = 0.25)`

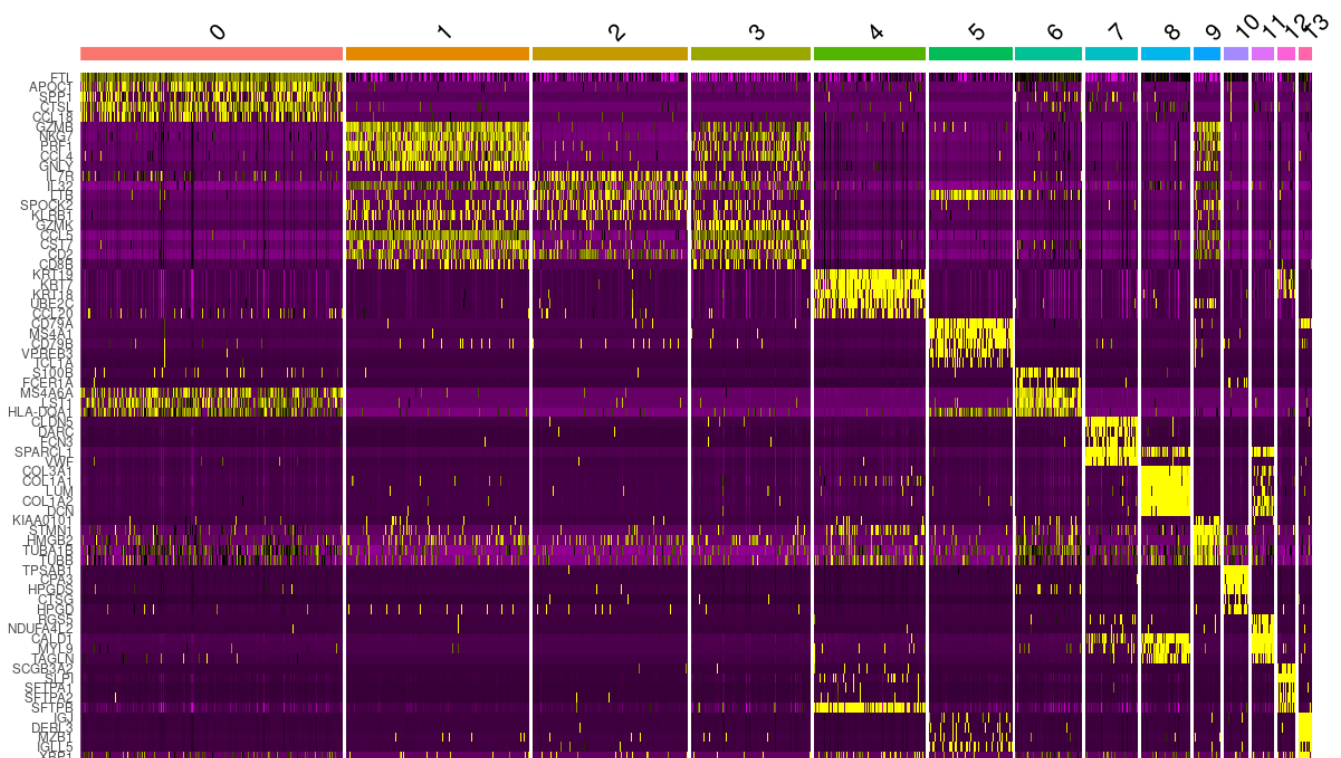
- **logfc.threshold** : Limit testing to genes which show, on average, at least X-fold difference (log-scale) between the two groups of cells (Default : 0.25)
- **test.use** : Denotes which test to use.
 - *wilcox* (default), *bimod*, *roc*, *t*, *negbinom*, *poisson*, *LR*, *MAST*, *DESeq2*
- **min.pct** : only test genes that are detected in a minimum fraction of min.pct cells in either of the two populations (Default : 0.1)
- **only.pos** : Only return positive markers (default : FALSE)

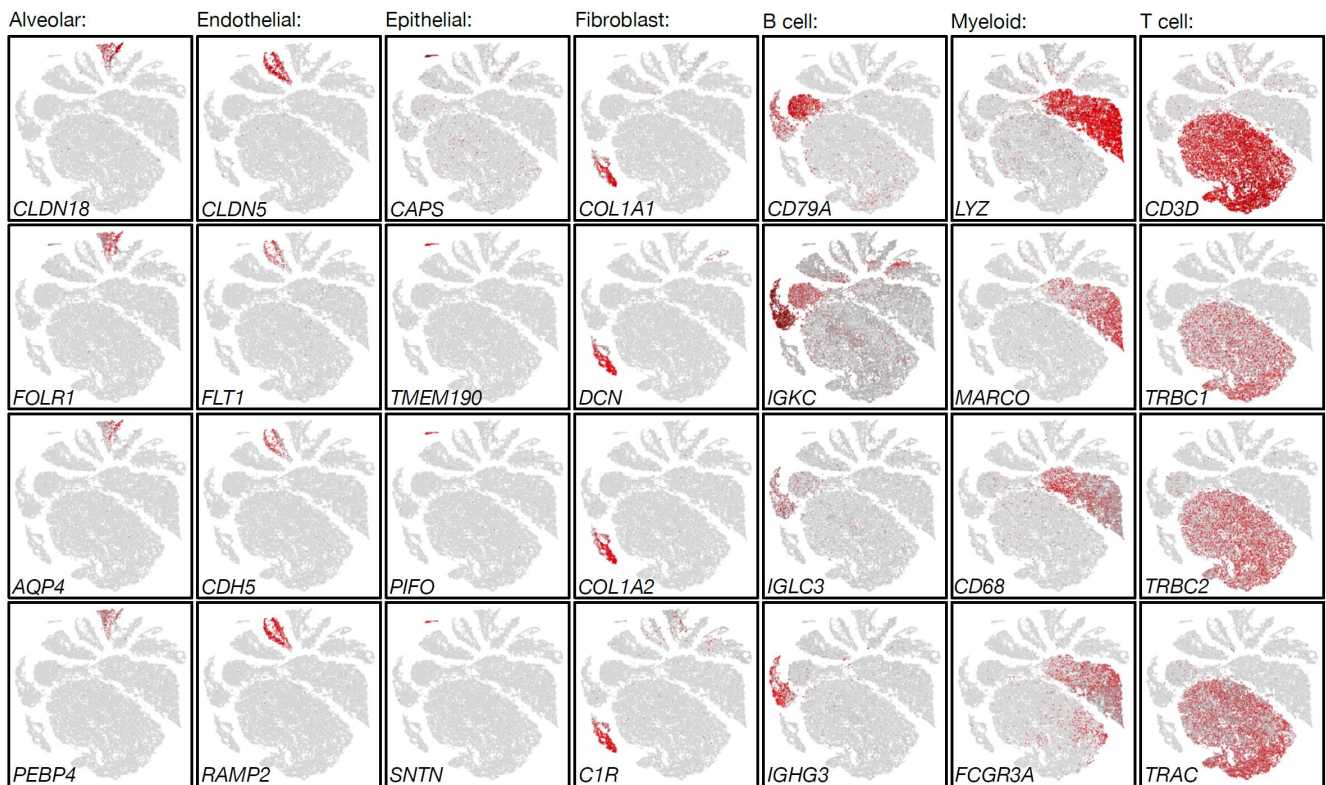
Finding differentially expressed features

	p_val	avg_logFC	pct.1	pct.2	p_val_adj	cluster	gene
FTL	0.000000e+00	3.2485606	1.000	0.792	0.000000e+00	0	FTL
APOC1	0.000000e+00	3.1473435	0.787	0.086	0.000000e+00	0	APOC1
SPP1	0.000000e+00	3.0291203	0.577	0.027	0.000000e+00	0	SPP1
CTSL	0.000000e+00	2.5713692	0.803	0.089	0.000000e+00	0	CTSL
LYZ	0.000000e+00	2.5071625	0.933	0.061	0.000000e+00	0	LYZ
FCER1G	0.000000e+00	2.4770294	0.986	0.145	0.000000e+00	0	FCER1G
CD68	0.000000e+00	2.4704571	0.888	0.088	0.000000e+00	0	CD68
APOE	0.000000e+00	2.4695281	0.626	0.062	0.000000e+00	0	APOE
CTSB	0.000000e+00	2.4049268	0.925	0.224	0.000000e+00	0	CTSB
C1QB	0.000000e+00	2.3862444	0.730	0.081	0.000000e+00	0	C1QB
C1QA	0.000000e+00	2.3593286	0.751	0.097	0.000000e+00	0	C1QA
S100A8	0.000000e+00	2.3561258	0.628	0.028	0.000000e+00	0	S100A8
CTSD	0.000000e+00	2.3253321	0.941	0.419	0.000000e+00	0	CTSD
TYROBP	0.000000e+00	2.3151500	0.994	0.174	0.000000e+00	0	TYROBP
GLUL	0.000000e+00	2.2266139	0.917	0.183	0.000000e+00	0	GLUL
FTH1	0.000000e+00	2.1965144	1.000	0.901	0.000000e+00	0	FTH1
GPNMB	0.000000e+00	2.1177286	0.756	0.040	0.000000e+00	0	GPNMB
AIF1	0.000000e+00	2.0594121	0.937	0.076	0.000000e+00	0	AIF1

Finding differentially expressed features

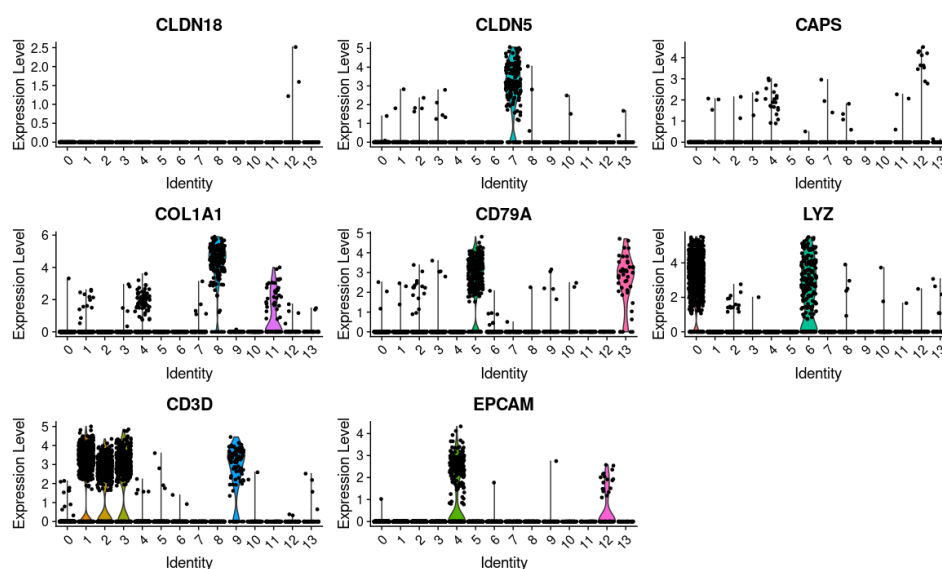
- `seurat.top.genes <- seurat.markers %>% group_by(cluster) %>% top_n(n = 10, wt = avg_logFC)`
- `DoHeatmap(object = seurat.obj, features = seurat.top.genes$gene) + NoLegend()`





Visualizing marker expression

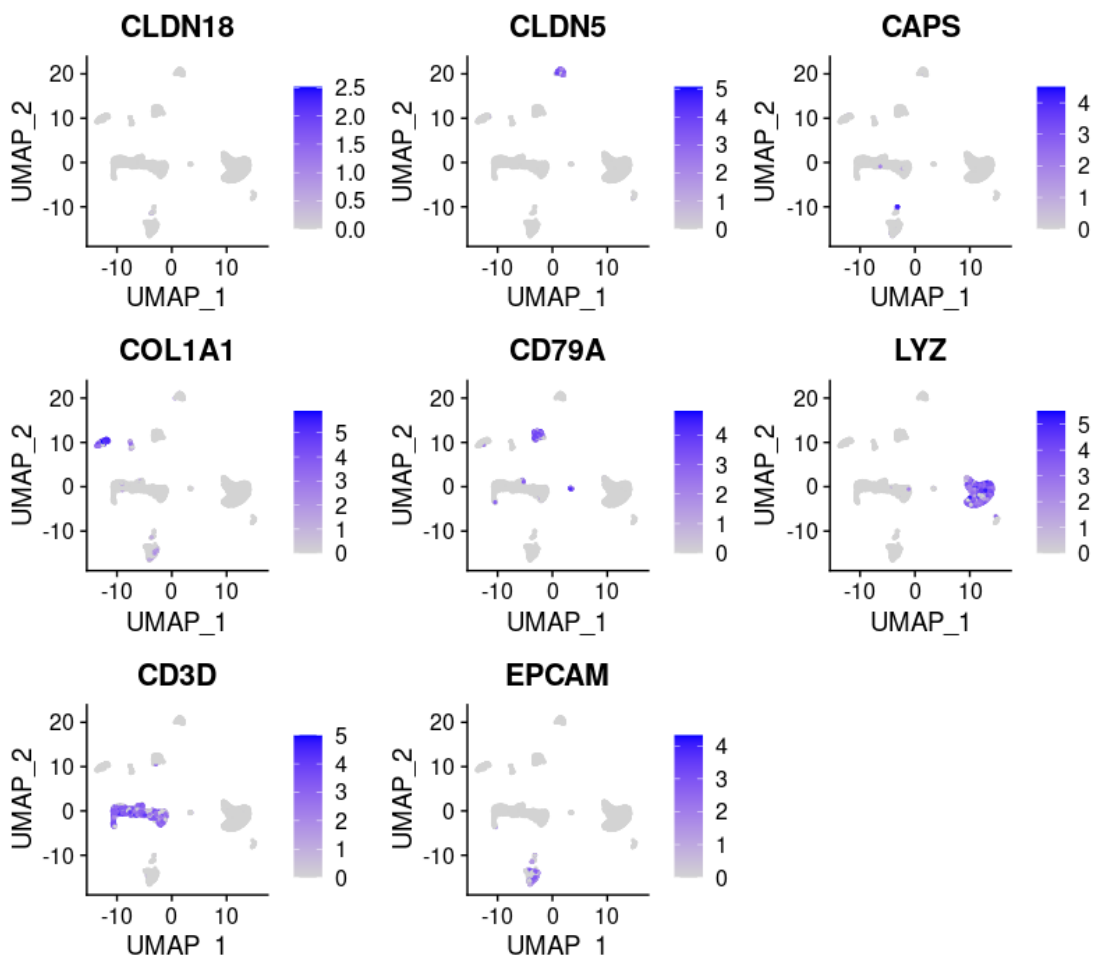
- `cell.markers <- c("CLDN18", "CLDN5", "CAPS", "COL1A1", "CD79A", "LYZ", "CD3D", "EPCAM")`
- `VlnPlot(object = seurat.obj, features = cell.markers)`



Visualizing marker expression

➤ `FeaturePlot(object = seurat.obj, reduction = "umap", features = cell.markers)`

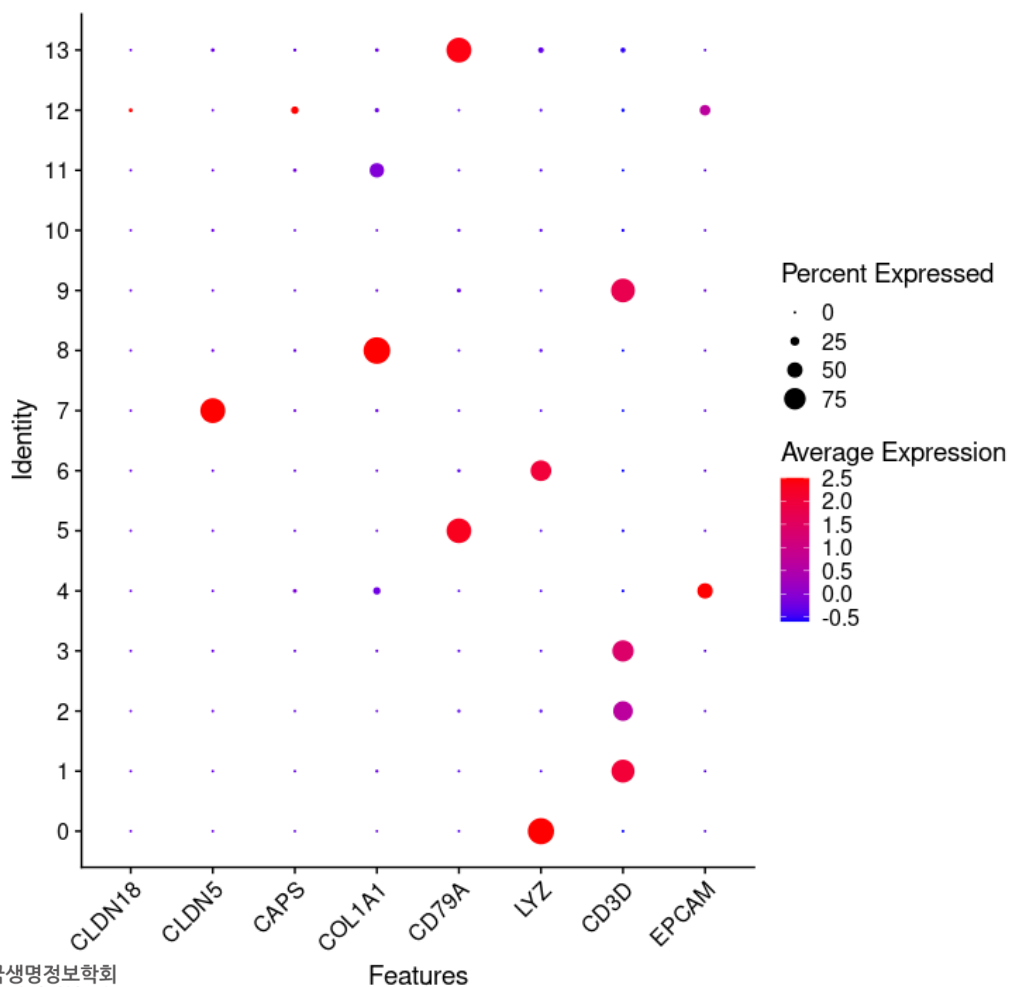
- **features** : Vector of features to plot
- **reduction** : Which dimensionality reduction to use. If not specified, first searches for *umap*, then *tsne*, then *pca*
- **label** : Whether to label the clusters



Visualizing marker expression

➤ `DotPlot(object = seurat.obj, features = cell.markers, cols = c("blue", "red")) + RotatedAxis()`

- ***features*** : Input vector of features
- ***cols*** : Colors to plot, can pass a single character giving the name of a palette from *RColorBrewer::brewer.pal.info*
- ***group.by*** : Factor to group the cells by



Parallelization in Seurat with future

- library(future)
- options(future.globals.maxSize = 50000 * 1024^2)
- plan("multiprocess", workers = 10)
- plan()

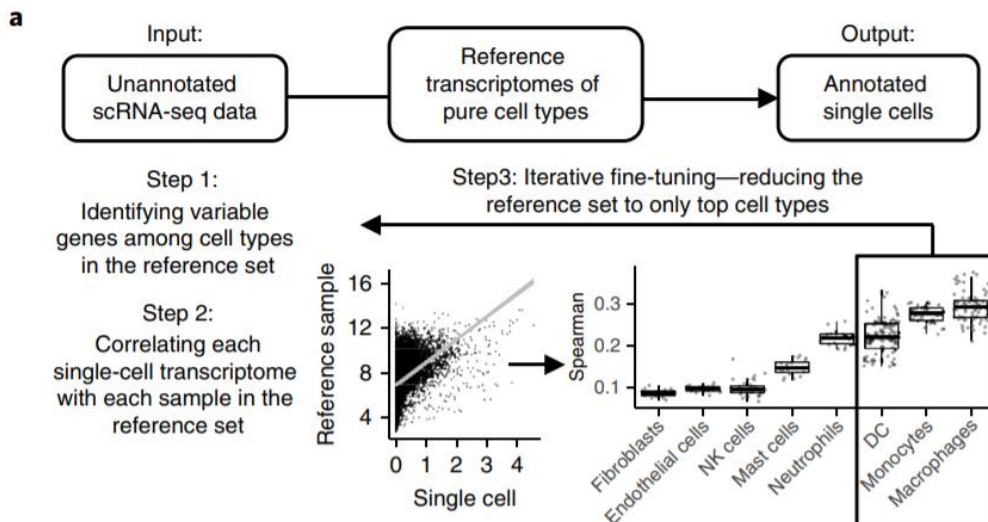
NormalizeData, ScaleData, JackStraw, FindMarkers,
FindIntegrationAnchors, FindClusters

Cell type annotation (SingleR)

<https://bioconductor.org/packages/release/bioc/vignettes/SingleR/inst/doc/SingleR.html>

Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage

Dvir Aran^{1,7}, Agnieszka P. Looney^{2,7}, Leqian Liu^{3,7}, Esther Wu², Valerie Fong², Austin Hsu⁴, Suzanna Chak², Ram P. Naikawadi², Paul J. Wolters², Adam R. Abate^{3,5,6}, Atul J. Butte¹ and Mallar Bhattacharya^{2*}



Load data

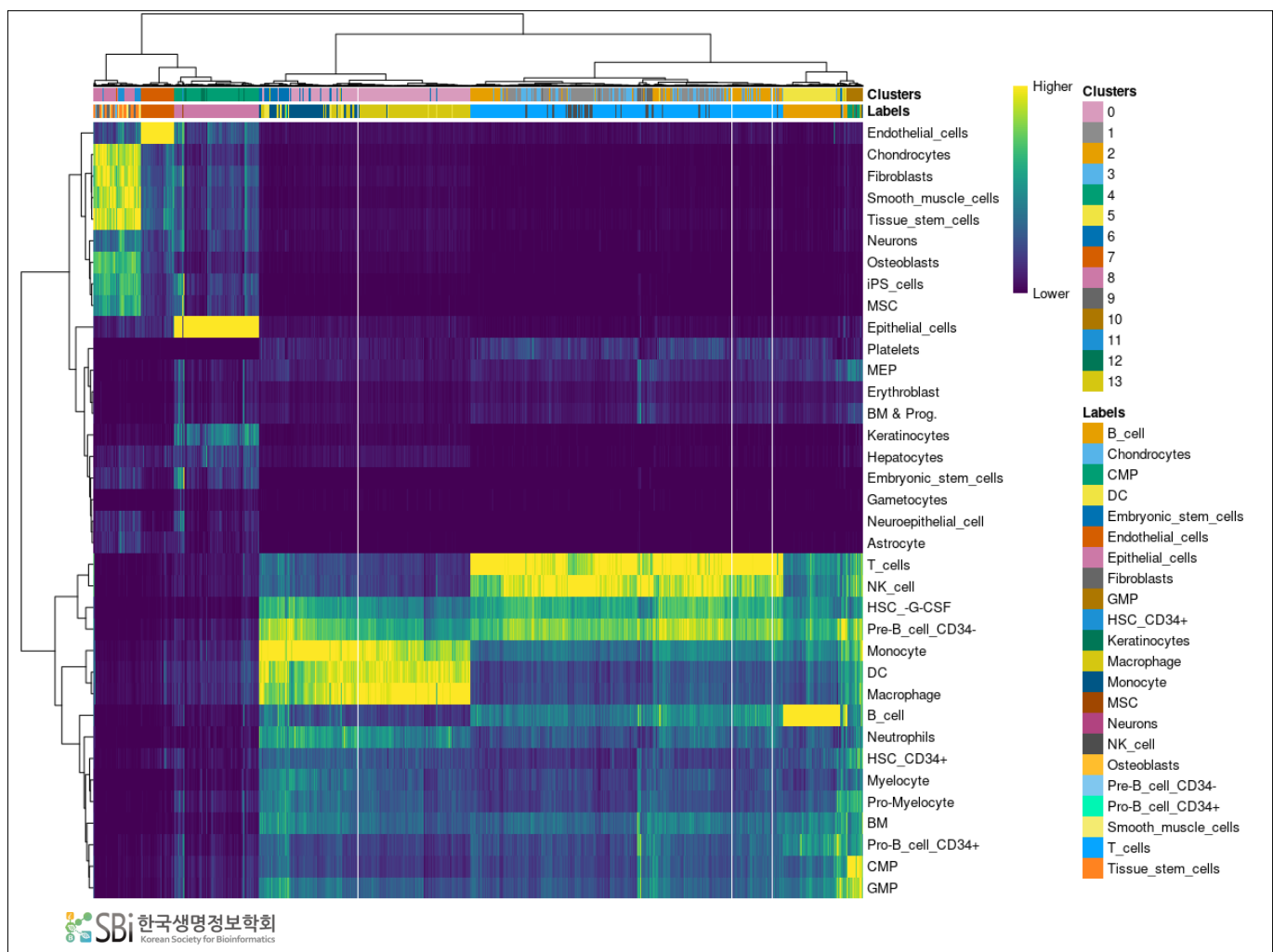
- library(SingleR)
- library(Seurat)
- library(scater)
- seurat.obj <- readRDS(file = "~/BIML-2021-SingleCellRNAseq/Result/Seurat/Seurat.RDS")
- hpca.se <- HumanPrimaryCellAtlasData()

Available references

Data retrieval	Organism	Samples	Sample types	No. of main labels	No. of fine labels	Cell type focus
<code>HumanPrimaryCellAtlasData()</code>	human	713	microarrays of sorted cell populations	37	157	Non-specific
<code>BlueprintEncodeData()</code>	human	259	RNA-seq	24	43	Non-specific
<code>DatabaseImmuneCellExpressionData()</code>	human	1561	RNA-seq	5	15	Immune
<code>NovershternHematopoieticData()</code>	human	211	microarrays of sorted cell populations	17	38	Hematopoietic & Immune
<code>MonacoImmuneData()</code>	human	114	RNA-seq	11	29	Immune
<code>ImmGenData()</code>	mouse	830	microarrays of sorted cell populations	20	253	Hematopoietic & Immune
<code>MouseRNAseqData()</code>	mouse	358	RNA-seq	18	28	Non-specific

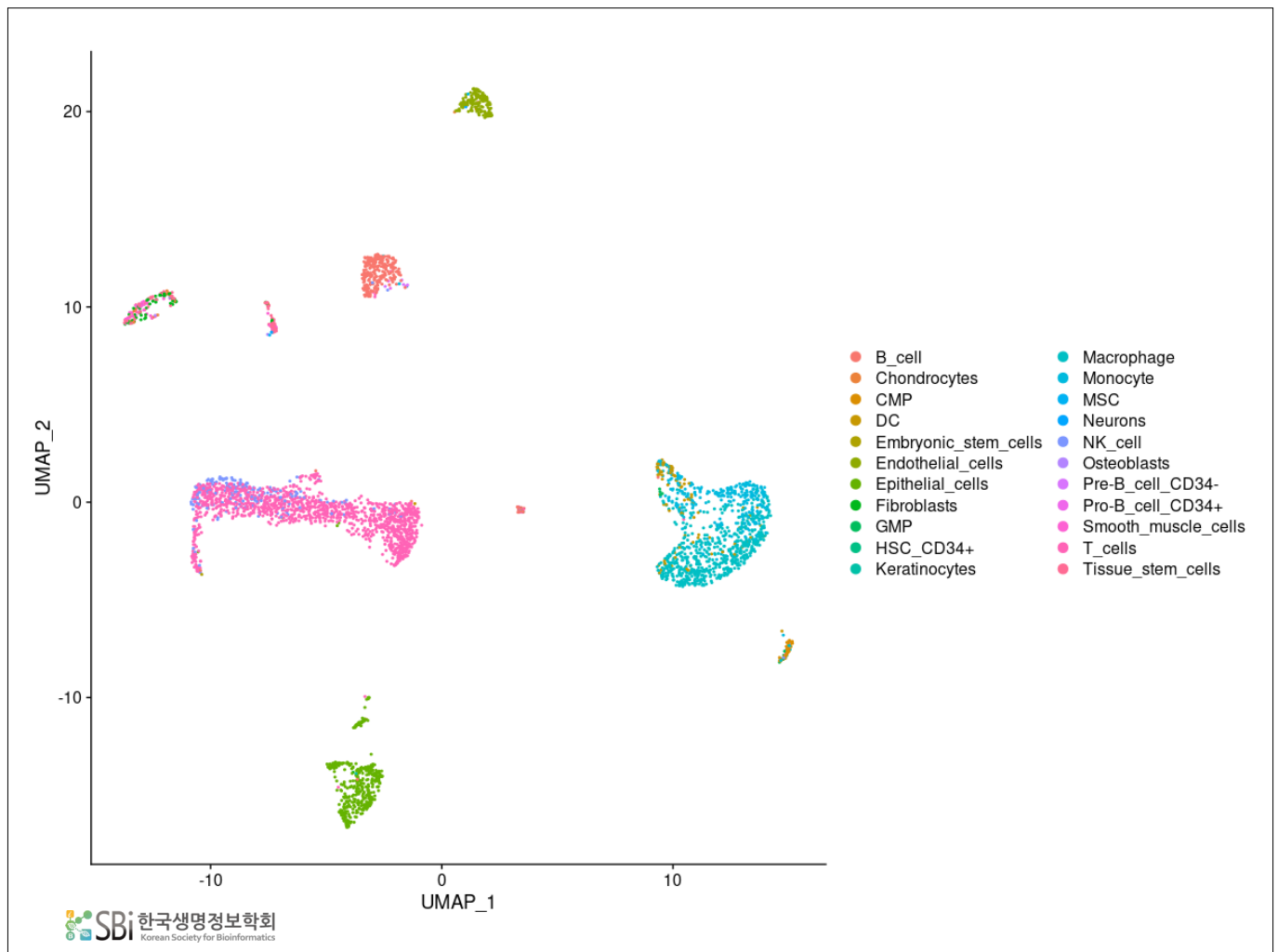
Annotate cell types

- `luad.sc <- as.SingleCellExperiment(x = seurat.obj)`
- `luad.sc <- logNormCounts(luad.sc)`
- `pred.hesc <- SingleR(test = luad.sc, ref = hpca.se, labels = hpca.se$label.main)`
- `plotScoreHeatmap(pred.hesc, clusters = colData(luad.sc)$seurat_clusters)`



Annotate single-cell

- `seurat.obj <- AddMetaData(object = seurat.obj, metadata = pred.hesc$labels, col.name = "SingleR.labels")`
- `DimPlot(object = seurat.obj, reduction = "umap", label = F, group.by = "SingleR.labels")`



Annotate single-cell

```
➤ count.df <- data.frame()

➤ for (temp.cluster in unique(seurat.obj@meta.data$seurat_clusters)){

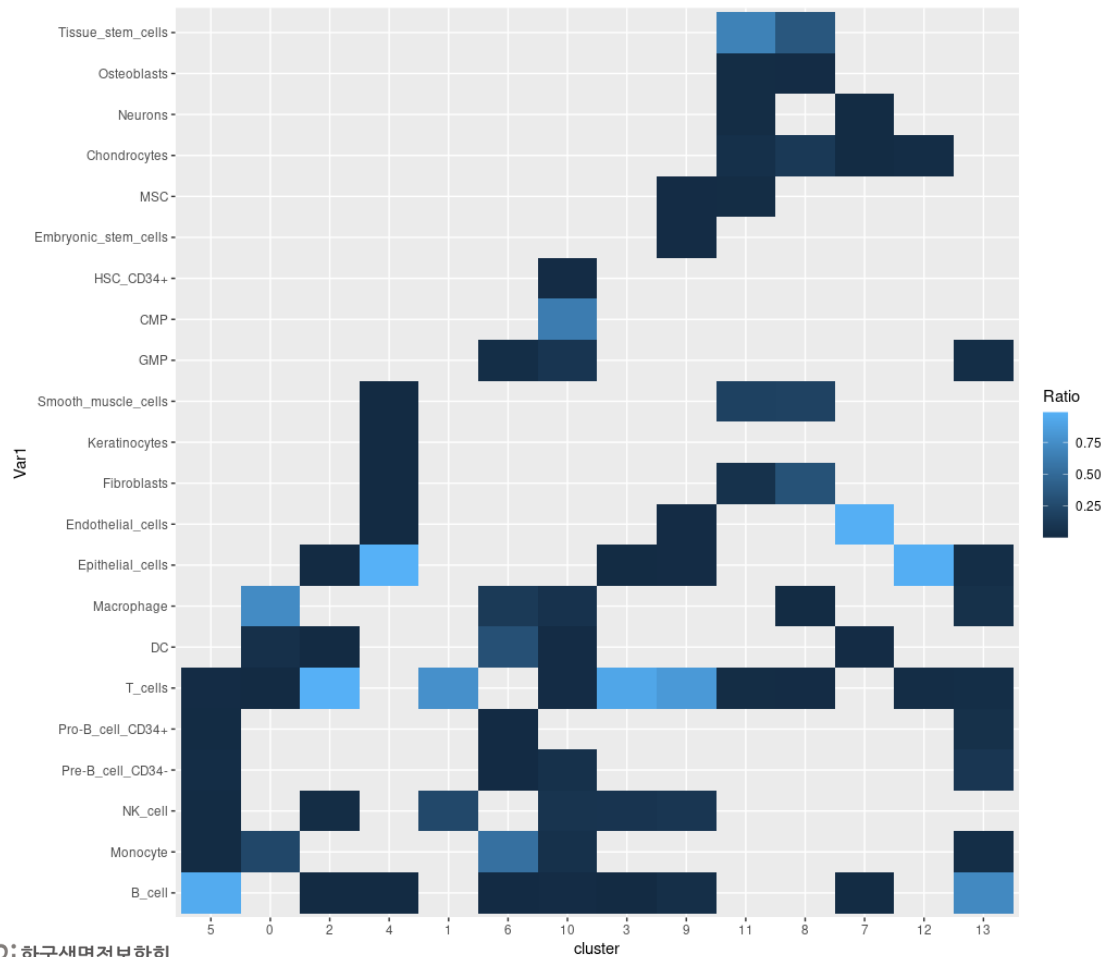
    temp.df <- data.frame(cluster = temp.cluster,
                           table(subset(seurat.obj@meta.data,
                                           seurat_clusters ==
                                           temp.cluster)$SingleR.labels))

    temp.df$Ratio <- temp.df$Freq/sum(temp.df$Freq)

    count.df <- rbind(count.df, temp.df)

}

➤ ggplot(data = count.df, aes(x = cluster, y = Var1, fill = Ratio)) + geom_tile()
```



Data Integration (Seurat)

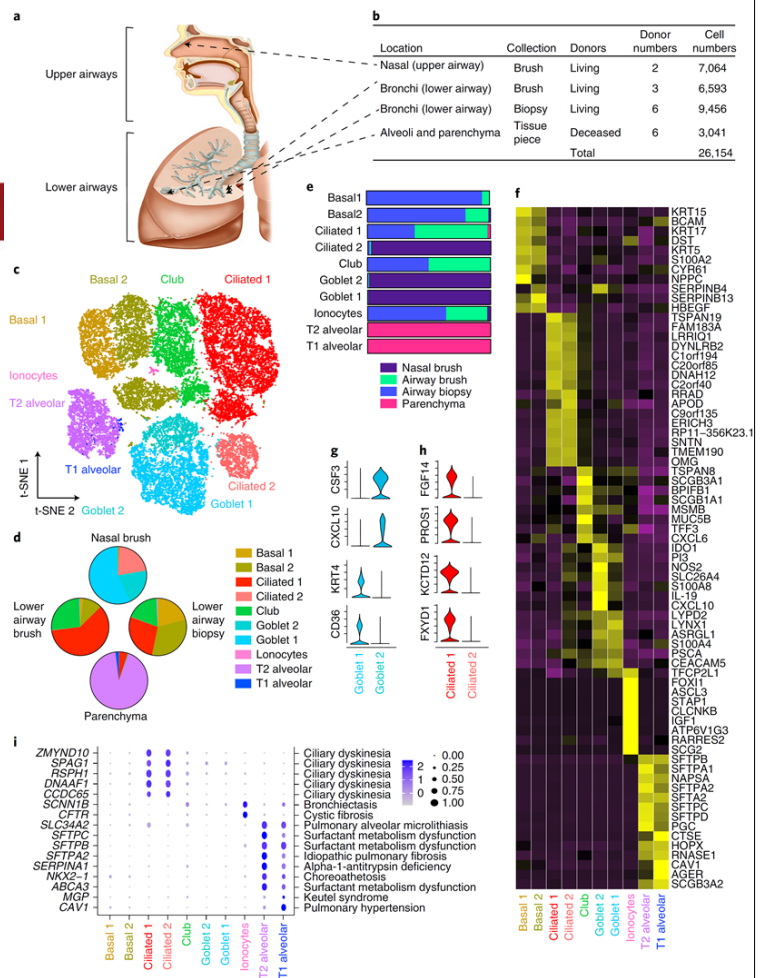
<https://satijalab.org/seurat/index.html>

Reference dataset



A cellular census of human lungs identifies novel cell states in health and in asthma

Felipe A. Vieira Braga^{1,2,38}, Gozde Kar^{1,2,38}, Marijn Berg^{3,4,39}, Orestes A. Carpaij^{4,5,38}, Krzysztof Polanski^{1,38}, Lukas M. Simon⁴, Sharon Brouwer^{3,4}, Tomás Gomes¹, Laura Hesse^{3,4}, Jian Jiang^{3,4}, Eirini S. Fasoulis^{1,2}, Mirjana Eremova¹, Roser Vento-Tormo¹, Carlos Talavera-López¹, Marnix R. Jonker^{3,4}, Karen Affleck⁷, Subarna Palit^{4,8,9,10}, Paulina M. Strzelecka^{1,12}, Helen V. Firth¹, Krishnaa T. Mahbubani¹³, Ana Cvejic^{1,12}, Kerstin B. Meyer¹, Kourosh Saeb-Parsy¹³, Marjan Luinge^{3,4}, Corry-Anke Brandsma^{3,4}, Wim Timens^{3,4}, Ilias Angelidis¹⁴, Maximilian Strunz¹⁴, Gerard H. Koppelman¹⁵, Antoon J. van Oosterhout¹, Herbert B. Schiller¹⁴, Fabian J. Theis^{4,16}, Maarten van den Berge^{4,5}, Martijn C. Nawijn^{3,4,19*} and Sarah A. Teichmann^{1,2,37,39*}

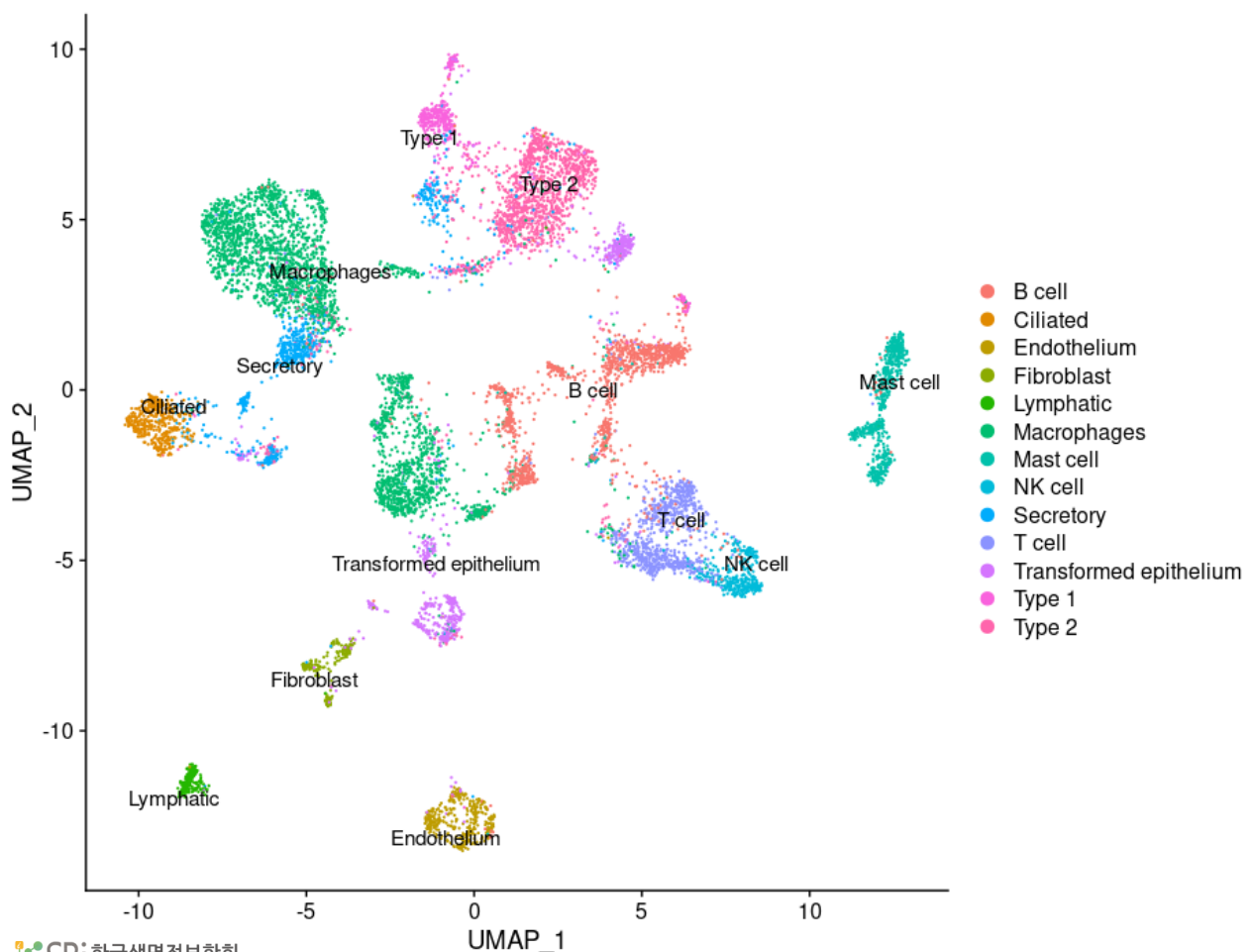


Load data sets

- library(Seurat)
- <- readRDS(file = "~/BIML-2021-SingleCellRNAseq/Result/Seurat/Seurat.RDS")
- sanger.count <- read.csv(file = "~/BIML-2021-SingleCellRNAseq/Resource/RawData/Sanger/GSE130148_raw_counts.csv", row.names = 1)
- sanger.meta <- read.table(file = "~/BIML-2021-SingleCellRNAseq/Resource/RawData/Sanger/GSE130148_barcode_cell_types.txt", header = T, sep = "\t", row.names = 1)
- sanger.obj <- CreateSeuratObject(counts = sanger.count, meta.data = sanger.meta)

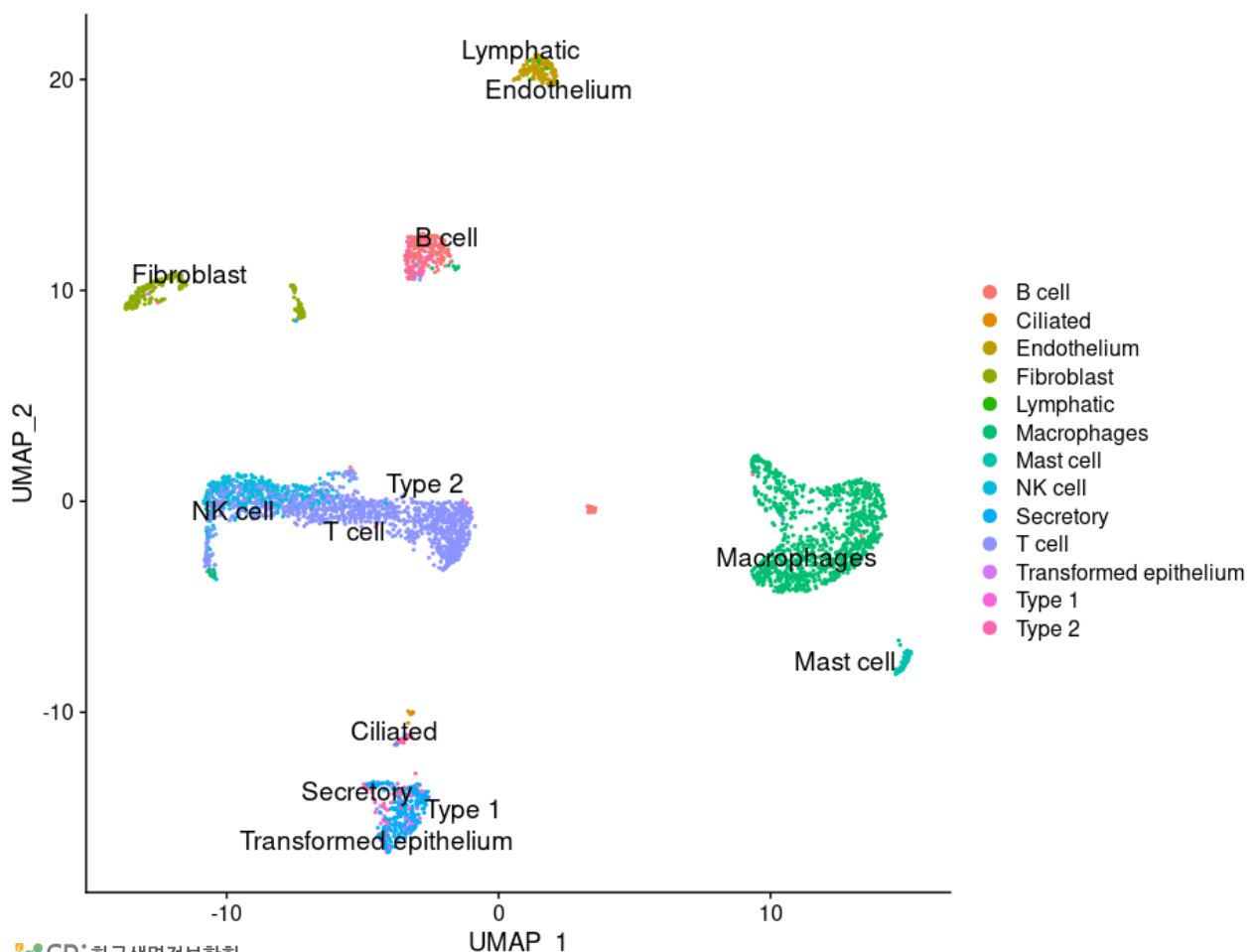
Clustering analysis

- `sanger.obj <- NormalizeData(sanger.obj , verbose = F)`
- `sanger.obj <- FindVariableFeatures(sanger.obj, selection.method = "vst", nfeatures = 2000, verbose = F)`
- `sanger.obj <- ScaleData(sanger.obj, verbose = FALSE)`
- `sanger.obj <- RunPCA(sanger.obj , npcs = 30, verbose = FALSE)`
- `sanger.obj <- RunUMAP(sanger.obj, reduction = "pca", dims = 1:30, verbose = FALSE)`
- `DimPlot(sanger.obj, reduction = "umap", group.by = "celltype", label = TRUE, repel = TRUE) + NoLegend()`



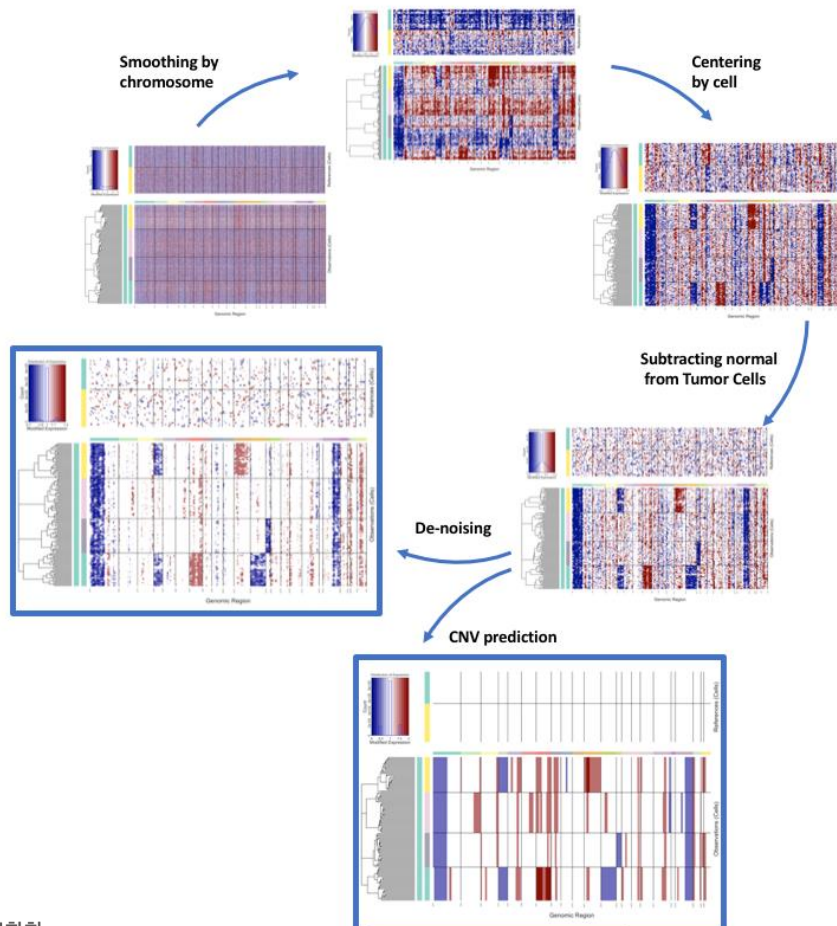
Integration

- `anchors <- FindTransferAnchors(reference = sanger.obj, query = seurat.obj, dims = 1:30)`
- `predictions <- TransferData(anchorset = anchors, refdata = sanger.obj$celltype, dims = 1:30)`
- `query <- AddMetaData(seurat.obj, metadata = predictions)`
- `DimPlot(query, reduction = "umap", group.by = "predicted.id", label = T, label.size = 3, repel = T) + NoLegend()`



Inferring copy number alterations (InferCNV)

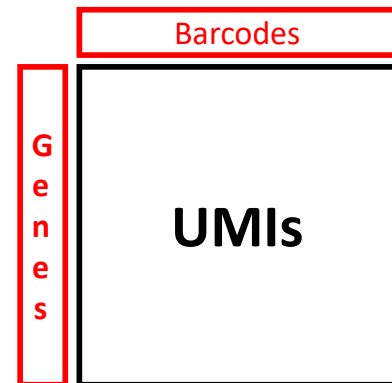
<https://github.com/broadinstitute/inferCNV/wiki>



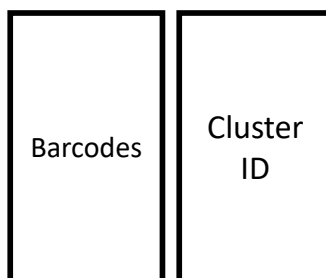
Input data files

1. read count matrix
2. cell type annotations
3. gene ordering file

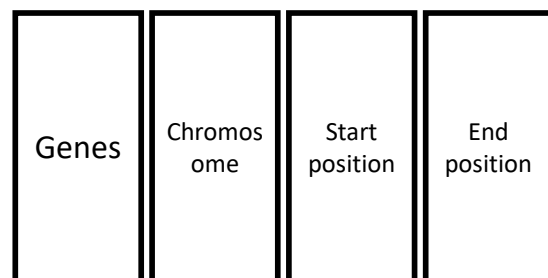
read count matrix



cell type annotations



gene ordering file



Make input data

- library(Seurat)
- library(infercnv)
- `seurat.obj <- readRDS(file = "~/BIML-2021-SingleCellRNAseq/Result/Seurat/Seurat.RDS")`
- `gene.order.file <- "~/BIML-2021-SingleCellRNAseq/Resource/Reference/hg19.inferCNV.gtf"`

Make input data

- `setwd("~/BIML-2021-SingleCellRNAseq/Result/inferCNV")`
- `write.table(x = as.matrix(seurat.obj@assays$RNA@counts), file =
"./inferCNV.matrix", quote = F, sep = "\t")`
- `infer.cluster.df <- data.frame(id =
rownames(seurat.obj@meta.data), cluster = paste0("C",
seurat.obj@meta.data$seurat_clusters))`
- `write.table(x = infer.cluster.df, file = "./inferCNV.annotation",
quote = F, sep = "\t", col.names = F, row.names = F)`

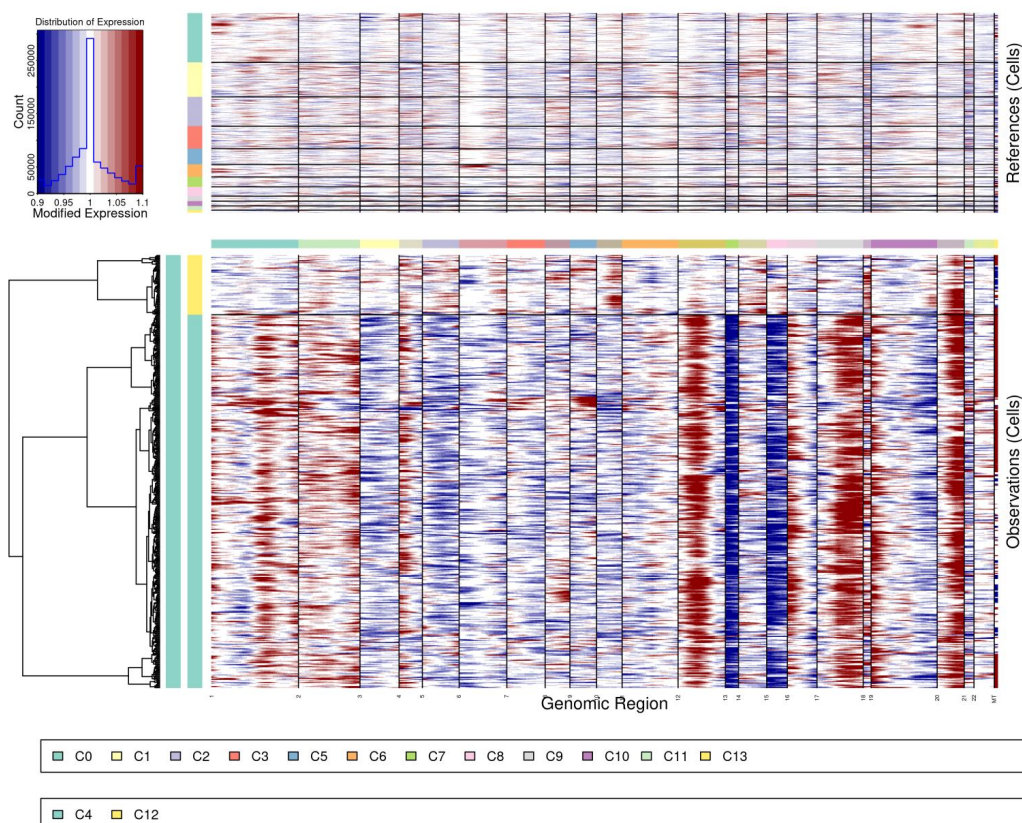
Creating an InferCNV object

- `ref.group <- c("C0", "C1", "C2", "C3", "C5", "C6", "C7", "C8", "C9", "C10", "C11",
"C13")`
 - Reference group is set to the various normal-cell type (non-tumor) as defined in the “cell type annotations” file
- `infercnv.obj <- CreateInfercnvObject(raw_counts_matrix = "./inferCNV.matrix",
annotations_file = "./inferCNV.annotation", gene_order_file = gene.order.file,
ref_group_names = ref.group, delim = "\t", chr_exclude = c("X", "Y"))`
 - **delim** : delimiter used in the input files
 - **chr_exclude** : list of chromosomes in the reference genome annotations that should be excluded from analysis (Default : `c('chrX', 'chrY', 'chrM')`)

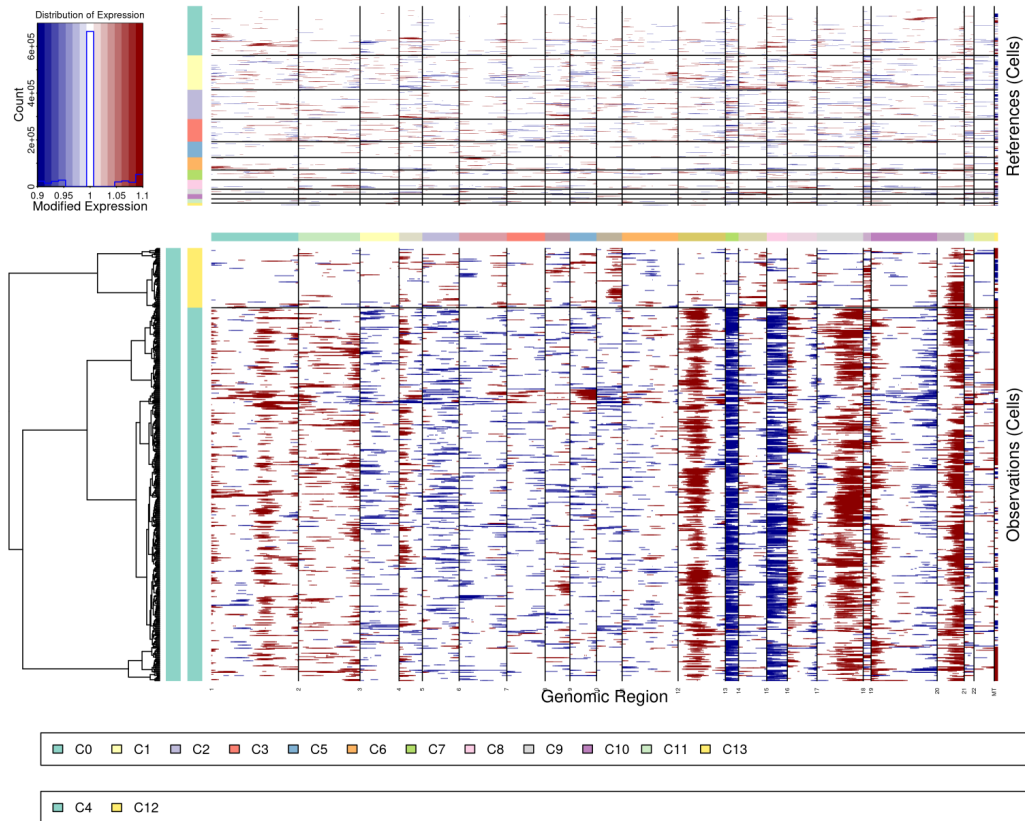
Running InferCNV

- `infercnv.obj <- infercnv::run(infercnv_obj = infercnv.obj, cutoff = 0.1, cluster_by_groups = T, denoise = T, HMM = F, out_dir = "./")`
- **cutoff** : Cut-off for the min average read counts per gene among reference cells. (default: 1)
 - ✓ use 1 for smart-seq, 0.1 for 10x-genomics
 - **cluster_by_groups** : If observations are defined according to groups (ie. patients), each group of cells will be clustered separately. (default : FALSE)
 - **denoise** : If True, turns on denoising according to options below
 - **HMM** : when set to True, runs HMM to predict CNV level (default : FALSE)
 - ❖ This option can take a long time

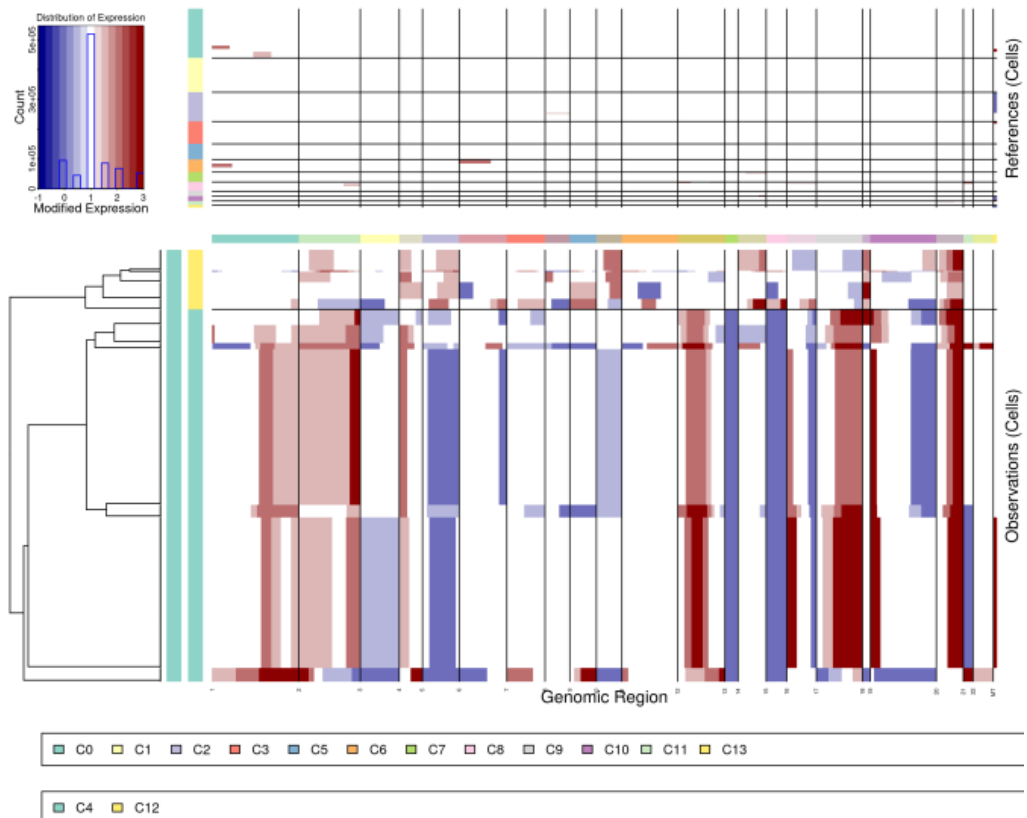
Preliminary infercnv (pre-noise filtering)



inferCNV

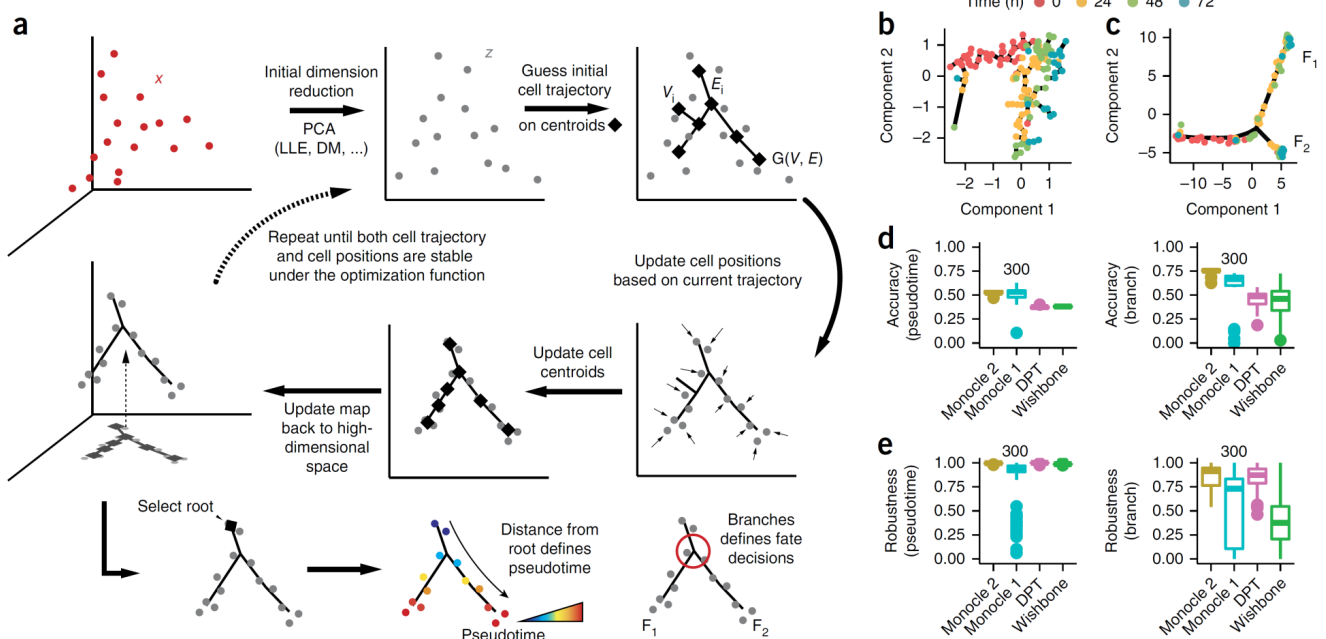


19_HMM_preds.repr_intensities



Trajectory analysis (monocle)

<http://cole-trapnell-lab.github.io/monocle-release/docs/>



Qiu et al. Nat Methods (2017)

T cell subclustering

- library(Seurat)
- library(monocle)
- seurat.obj <- readRDS(file = "~/BIML-2021-SingleCellRNAseq/Result/Seurat/Seurat.RDS")
- subset.obj <- subset(seurat.obj, seurat_clusters %in% c("1", "2", "3", "9"))

T cell subclustering

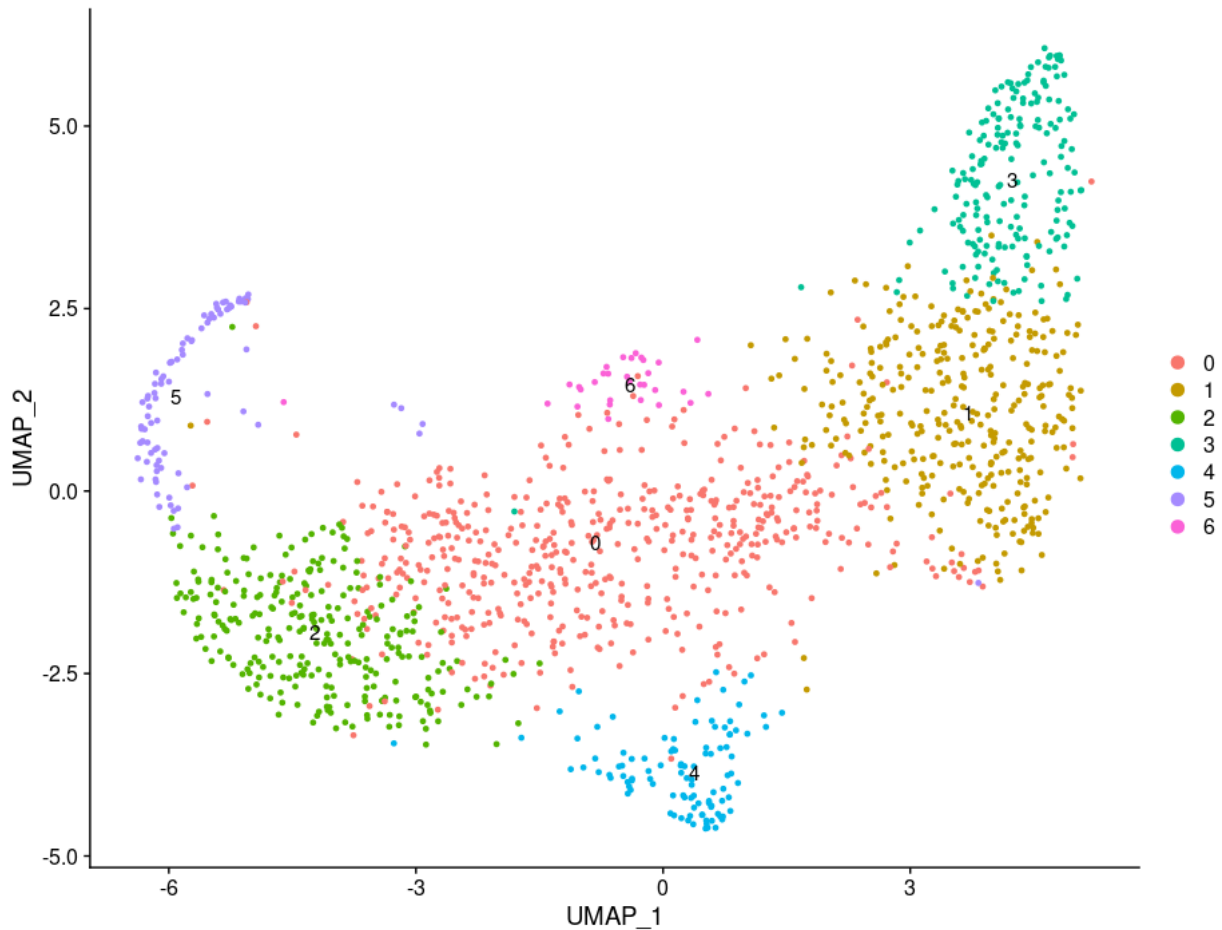
- subset.obj <- NormalizeData(object = subset.obj, normalization.method = "LogNormalize", scale.factor = 10000)
- subset.obj <- FindVariableFeatures(subset.obj, selection.method = "vst", nfeatures = 2000)
- subset.obj <- ScaleData(object = subset.obj, features = rownames(subset.obj), vars.to.regress = "percent.mt")
- subset.obj <- RunPCA(object = subset.obj, features = VariableFeatures(seurat.obj))

T cell subclustering

- `subset.obj <- JackStraw(object = subset.obj, num.replicate = 100)`
- `subset.obj <- ScoreJackStraw(object = subset.obj, dims = 1:20)`
- `ElbowPlot(object = subset.obj)`
- `JackStrawPlot(object = subset.obj, dims = 1:20)`

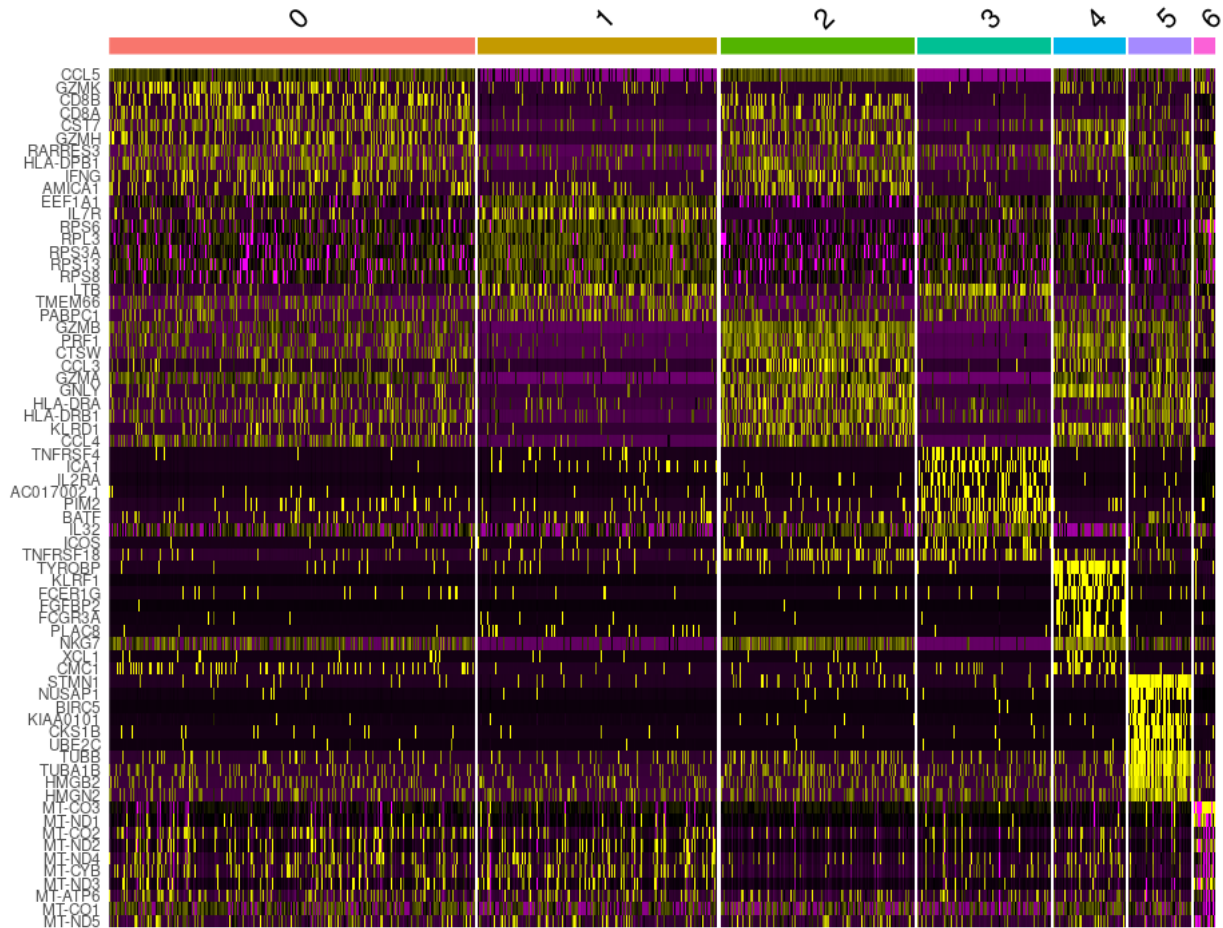
T cell subclustering

- `subset.obj <- FindNeighbors(object = subset.obj, dims = 1:10)`
- `subset.obj <- FindClusters(object = subset.obj, resolution = 0.5)`
- `subset.obj <- RunUMAP(object = subset.obj, dims = 1:10)`
- `DimPlot(object = subset.obj, reduction = "umap", label = T)`



T cell subclustering

- `seurat.markers <- FindAllMarkers(object = subset.obj, only.pos = T, min.pct = 0.25, logfc.threshold = 0.25)`
- `subset.top.genes <- subset.markers %>%
group_by(cluster) %>% top_n(n = 5, wt = avg_logFC)`
- `DoHeatmap(object = subset.obj, features =
subset.top.genes$gene) + NoLegend()`



Create monocle object

- `data <- as(subset.obj@assays$RNA@data, "matrix")`
- `pd <- new('AnnotatedDataFrame', data = subset.obj@meta.data)`
- `gene.df <- data.frame(apply(X = data, 1, FUN = function(x) { sum(x > 0) } ))`
- `fd <- new('AnnotatedDataFrame', data = data.frame(gene_short_name = row.names(gene.df), row.names = row.names(gene.df), num_cells_expressed = gene.df[,1]))`
- `cds <- newCellDataSet(data, phenoData = pd, featureData = fd, expressionFamily = negbinomial.size())`

Estimate size factors and dispersions

- `cds <- estimateSizeFactors(cds)`
- `cds <- estimateDispersions(cds)`

Filtering low-quality cells

- `set.seed(123)`
- `cds <- detectGenes(cds, min_expr = 0.1)`
- `expressed_genes <- row.names(subset(fData(cds),
num_cells_expressed >= 50))`
- `cds <- cds[expressed_genes,]`

Select genes by seurat

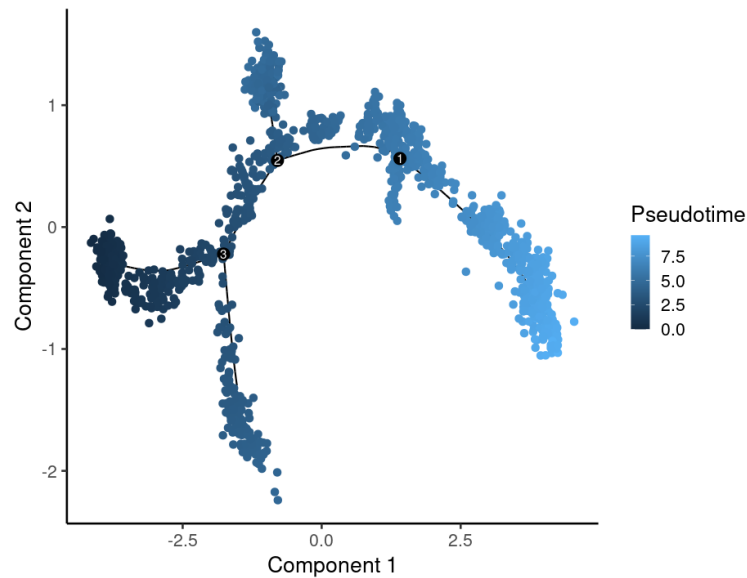
- `subset.markers <- subset(subset.markers, p_val_adj < 0.05)`
- `subset.markers.top.genes <- subset.markers %>%
group_by(cluster) %>% top_n(n = 50, wt = avg_logFC)`
- `order.genes <-
unique(as.character(subset.markers.top.genes$gene))`

Order cells in pseudotime along a trajectory

- `cds <- setOrderingFilter(cds, order.genes)`
- `cds <- reduceDimension(cds = cds,
max_components = 3, method = 'DDRTree')`
- `cds <- orderCells(cds)`

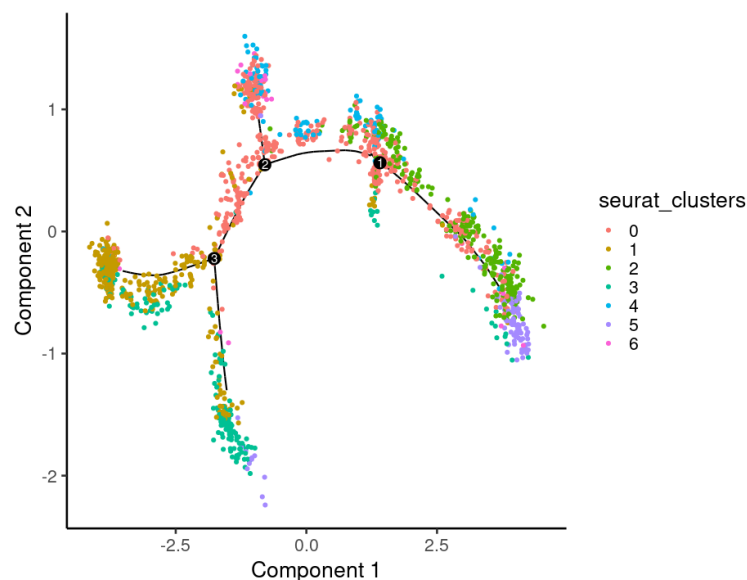
Plot trajectory

➤ `plot_cell_trajectory(cds, color_by = "Pseudotime", cell_size = 3) + theme_classic(base_size = 20)`



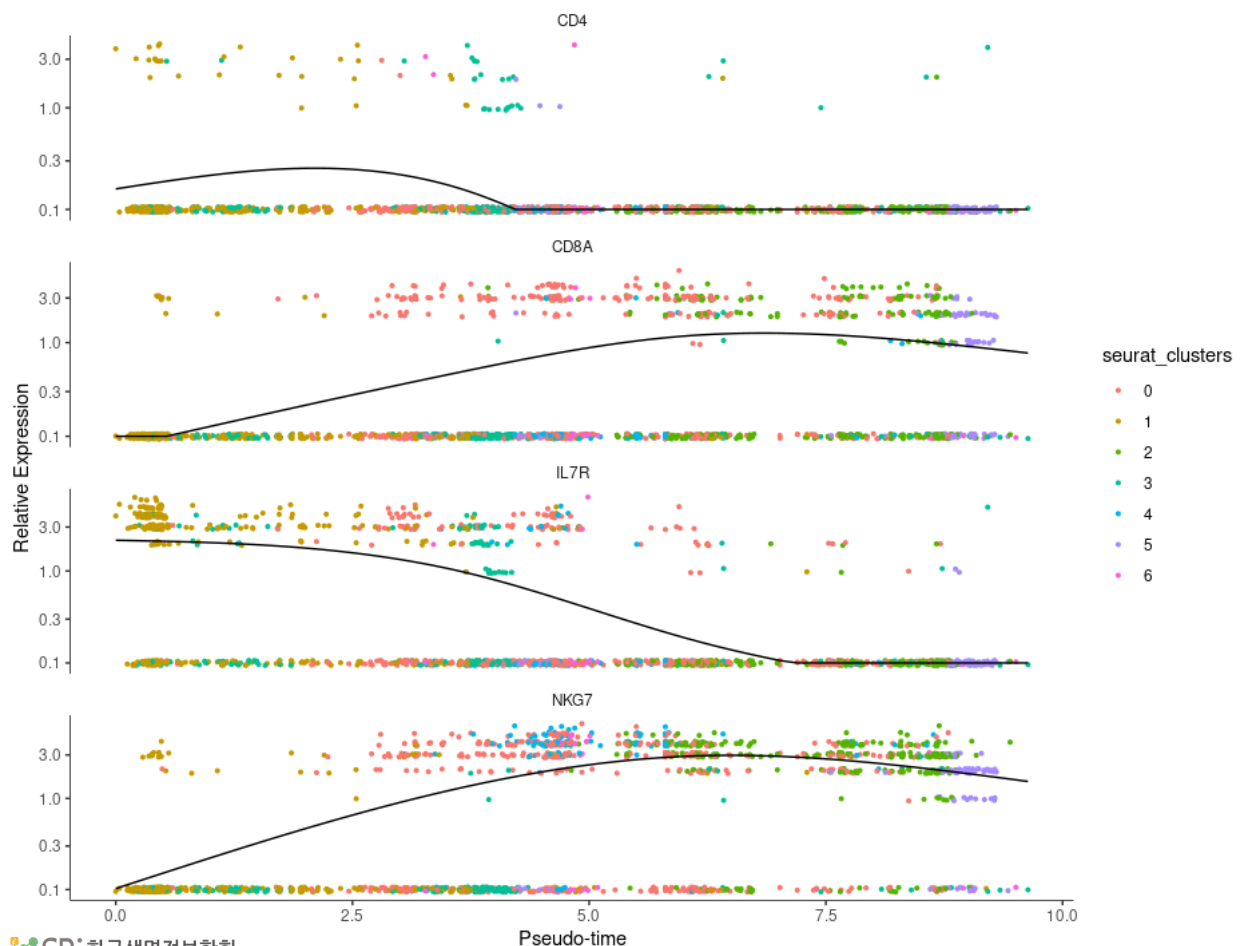
Plot trajectory

➤ `plot_cell_trajectory(cds, color_by = "seurat_clusters") + theme_classic(base_size = 20)`



Plot trajectory

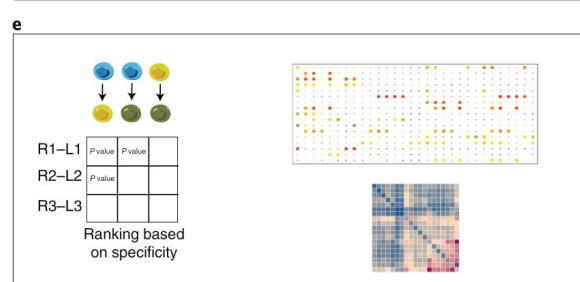
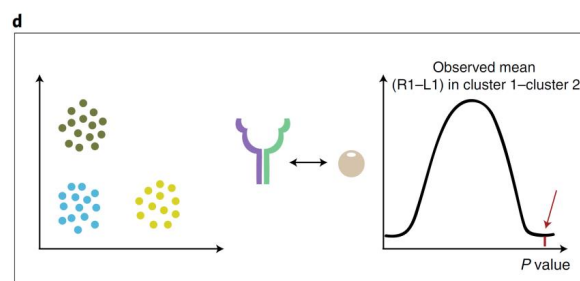
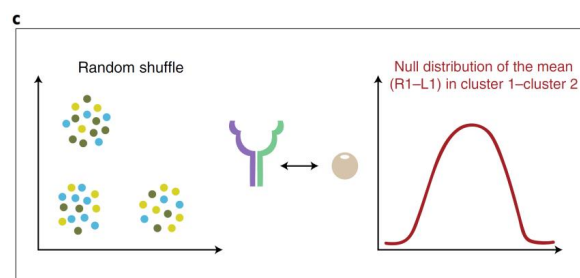
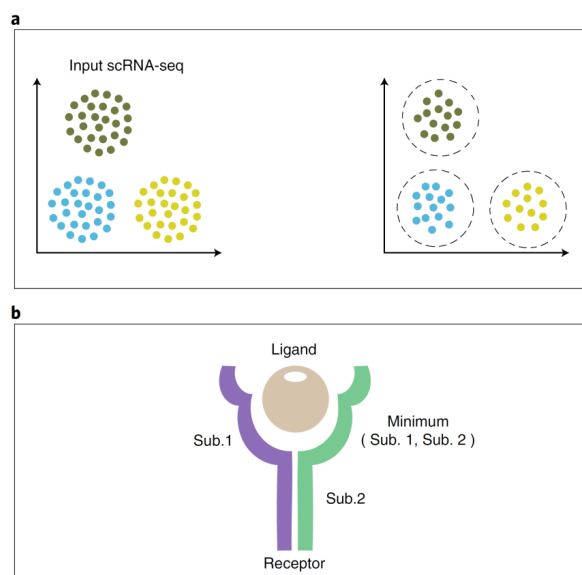
- `my_genes <- row.names(subset(fData(cds),
gene_short_name %in% c("IL7R", "CD4", "CD8A",
"FOXP3", "NKG7")))`
- `cds_subset <- cds[my_genes,]`
- `plot_genes_in_pseudotime(cds_subset, color_by =
"seurat_clusters")`



Cell-cell communication (CellPhoneDB)

<https://github.com/Teichlab/cellphonedb>

CellPhoneDB: inferring cell-cell communication from combined expression of multi-subunit ligand-receptor complexes



Create an Input Data

- `library(Seurat)`
- `seurat.obj <- readRDS(file = "~/BIML-2021-SingleCellRNAseq/Result/Seurat/Seurat.RDS")`
- `seurat.obj <- subset(seurat.obj, seurat_clusters %in% c("6", "9", "12"))`
- `seurat.obj@meta.data$celltypes <-
""seurat.obj@meta.data$celltypes[seurat.obj@meta.data$seurat_clusters %in%
c("6")] <- "Myeloid"`
- `seurat.obj@meta.data$celltypes[seurat.obj@meta.data$seurat_clusters %in%
c("9", "12")] <- "Epithelial"`

Create an Input Data

- `seurat.cpm <- RelativeCounts(data = seurat.obj@assays$RNA@counts, scale.factor = 1e6)`
- `seurat.cpm <- data.frame(Gene = rownames(seurat.cpm), seurat.cpm, check.names = F)`
- `seurat.meta <- data.frame(Cell = rownames(seurat.obj@meta.data), Celltypes =
seurat.obj@meta.data$celltypes)`
- `write.table(x = seurat.cpm, file = "~/BIML-2021-SingleCellRNAseq/Result/CellPhoneDB/CellPhoneDB.input.count.txt", sep = "\t", quote = F,
row.names = F)`
- `write.table(x = seurat.meta, file = "~/BIML-2021-SingleCellRNAseq/Result/CellPhoneDB/CellPhoneDB.input.meta.txt", sep = "\t", quote = F,
row.names = F)`

Running CellPhoneDB (python)

▪ cellphonedb method statistical_analysis --counts-data hgnc_symbol --iterations 100 --threshold 0.1 --output-path ./ --pvalue 0.05 --threads 1 CellPhoneDB.input.meta.txt
CellPhoneDB.input.count.txt

- `--counts-data` : Type of gene identifiers in the counts data (ensembl | gene_name | hgnc_symbol)
- `--iterations` : Number of iterations for the statistical analysis (default : 1000)
- `--threshold` : % of cells expressing the specific ligand/receptor
- `--subsampling` : Enable subsampling
- `--subsampling-num-cells` : Number of cells to subsample to (default : 1/3 of cells)
- `--pvalue` : P-value threshold [0.05]
- `--threads` : Number of threads to use

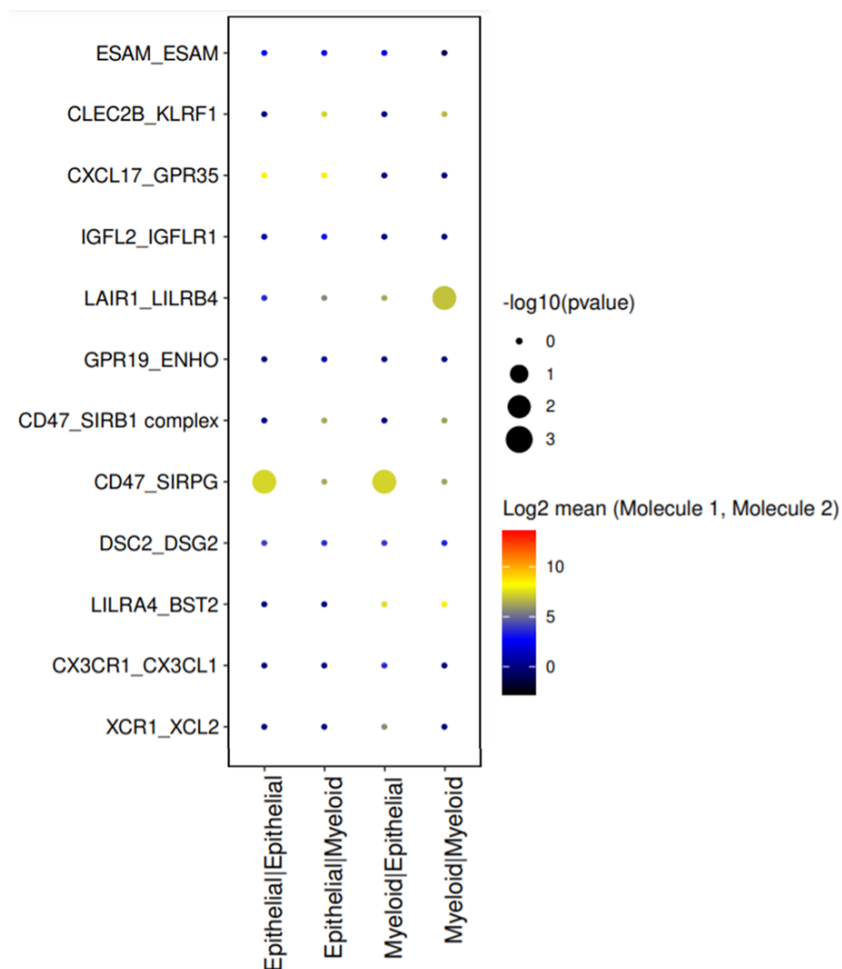
Output

➤ `means.df <- read.table("~/BIML-2021-SingleCellRNAseq/Result/CellPhoneDB/means.txt", header = T, sep = "\t")`

	id_cp_interaction	interacting_pair	partner_a	partner_b	gene_a	gene_b	secreted	receptor_a	receptor_b	annotation_strategy
1	CPI-SS0FEC87269	KLRG2_TNFSF9	simple:A4D1S0	simple:P41273	KLRG2	TNFSF9	True	True	False	InnateDB-All
2	CPI-SS028784FC6	HLA-DPA1_TNFSF9	simple:HLADPA1	simple:P41273	HLA-DPA1	TNFSF9	True	True	False	InnateDB-All
3	CPI-SS00A8596B5	PVR_TNFSF9	simple:P15151	simple:P41273	PVR	TNFSF9	True	True	False	InnateDB-All
4	CPI-SS0B84DAE3D	PVR_CD96	simple:P15151	simple:P40200	PVR	CD96	True	True	True	curated
5	CPI-SS0A8627ED6	PVR_CD226	simple:P15151	simple:Q15762	PVR	CD226	True	True	True	curated
6	CPI-SS00561BBD7	PVR_TIGIT	simple:P15151	simple:Q495A1	PVR	TIGIT	True	True	False	curated
7	CPI-SS0889E281D	CADM1_CADM1	simple:Q9BY67	simple:Q9BY67	CADM1	CADM1	False	False	False	curated
8	CPI-SS0C9696AAA	CRTAM_CADM1	simple:Q95727	simple:Q9BY67	CRTAM	CADM1	False	False	False	curated
9	CPI-SC07ECBFA7	COL1A1_a10b1 complex	simple:P02452	complex:a10b1 complex	COL1A1		True	False	False	curated
10	CPI-SC0B6F1A278	COL4A1_a10b1 complex	simple:P02462	complex:a10b1 complex	COL4A1		True	False	False	curated

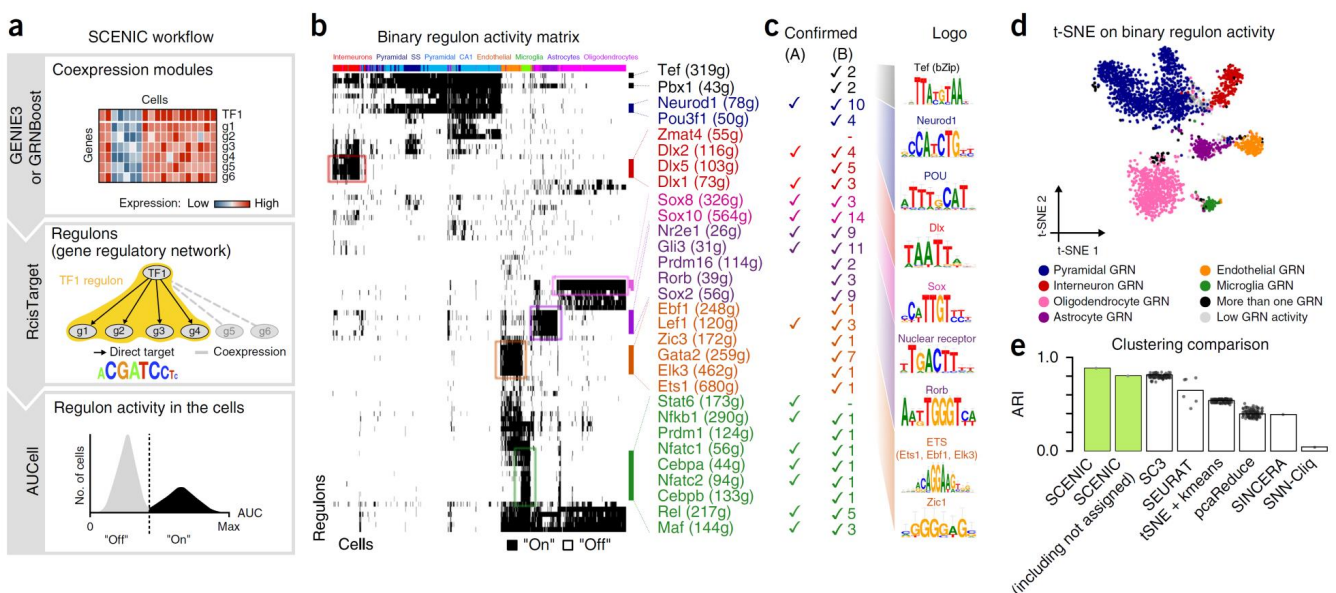
Plotting statistical method results

- `cellphonedb plot dot_plot --means-path ./means.txt --pvalues-path ./pvalues.txt --output-path ./ --output-name dot_plot.pdf`
 - `--means-path` : The means output file
 - `--pvalues-path` : The pvalues output file
 - `--output-path` : Output folder
 - `--rows` : File with a list of rows to plot, one per line
 - `--columns` : File with a list of columns to plot, one per line



Gene Regulatory Networks (SCENIC)

<https://github.com/aertslab/SCENIC>



S Aibar et al. Nat Methods (2017)

Load Seurat object

- `library(Seurat)`
- `library(SCENIC)`
- `library(plyr)`
- `library(AUCell)`
- `library(foreach)`
- `seurat.obj <- readRDS(file = "~/BIML-2021-SingleCellRNAseq/Result/Seurat/Seurat.RDS")`
- `seurat.obj <- subset(seurat.obj, seurat_clusters %in% c("1", "2", "3", "9"))`
- `setwd("~/BIML-2021-SingleCellRNAseq/Result/SCENIC")`
- `cistarget.dir <- "~/BIML-2021-SingleCellRNAseq/Resource/Reference/cisTarget"`

Co-expression network

- `scenicOptions <- initializeScenic(org = "hgnc", dbDir = cistarget.dir, nCores = 1)`
- `expr <- as.matrix(seurat.obj@assays$RNA@counts)`
- `genes <- geneFiltering(expr, scenicOptions = scenicOptions)`
- `expr <- expr[genes, ]`
- `runCorrelation(expr, scenicOptions)`
- `expr <- log2(expr + 1)`
- `runGenie3(expr, scenicOptions)`

Build and score the GRN

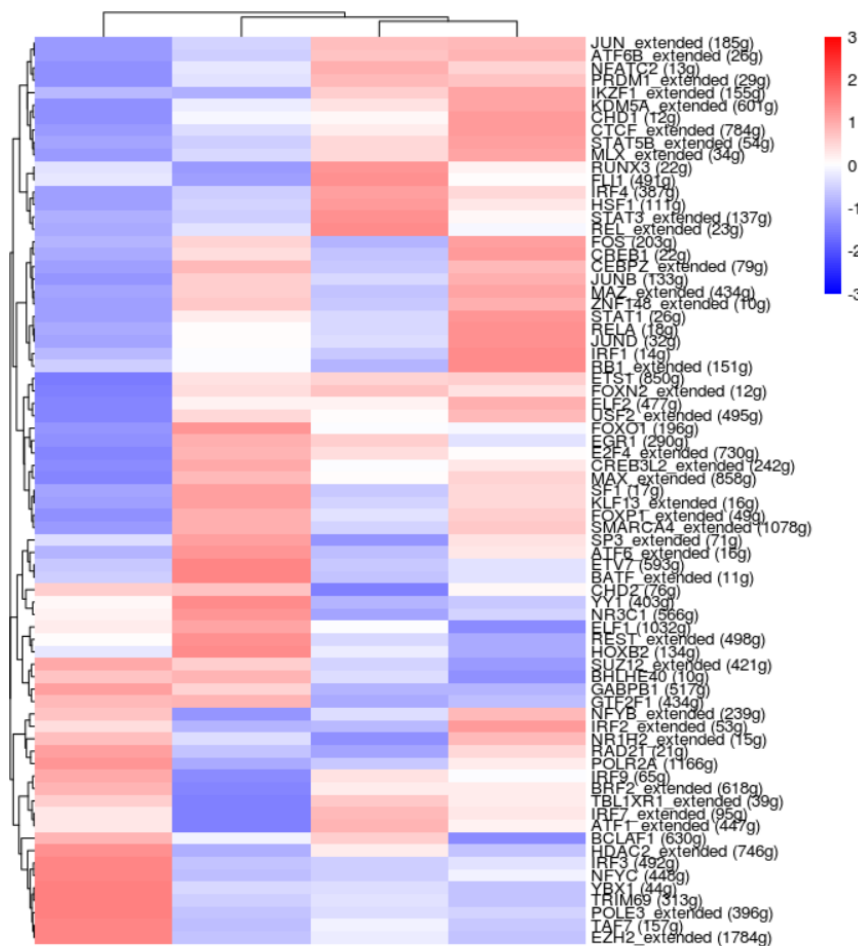
- `scenicOptions@settings$verbose <- TRUE`
- `scenicOptions@settings$nCores <- 1`
- `scenicOptions@settings$seed <- 123`
- `scenicOptions@settings$db <- scenicOptions@settings$db["10kb"]`
- `runSCENIC_1_coexNetwork2modules(scenicOptions)`
- `runSCENIC_2_createRegulons(scenicOptions)`
- `runSCENIC_3_scoreCells(scenicOptions, expr)`

Regulators for known clusters

- `cellInfo <- data.frame(Clusters =
as.character(seurat.obj@meta.data$seurat_clusters))`
- `rownames(cellInfo) <- rownames(seurat.obj@meta.data)`
- `regulon.auc <- loadInt(scenicOptions, "aucell_regulonAUC")`
- `regulon.auc <-
regulon.auc[onlyNonDuplicatedExtended(rownames(regulo
n.auc)),]`

Regulators for known clusters

- `regulon.auc.cluster <- sapply(split(rownames(cellInfo),
cellInfo$Clusters), function(cells)
rowMeans(getAUC(regulon.auc)[,cells]))`
- `regulon.auc.cluster.scale <- t(scale(t(regulon.auc.cluster), center =
T, scale = T))`
- `pheatmap::pheatmap(regulon.auc.cluster.scale, color =
colorRampPalette(c("blue","white","red"))(100), breaks = seq(-3,
3, length.out = 100), treeheight_row = 10, treeheight_col = 10,
border_color = NA)`



Thank you



Contact

Hyoung-oh Jeong

Email: hyoung-oh@unist.ac.kr

Organized by

