Lecture 5
scRNA-seq clustering and cell type annotation

Outline

- scRNA-seq clustering methods
- Cell type annotation

Louvain clustering (Blondel et. al., Journal of Statistical mechanics, 2008)

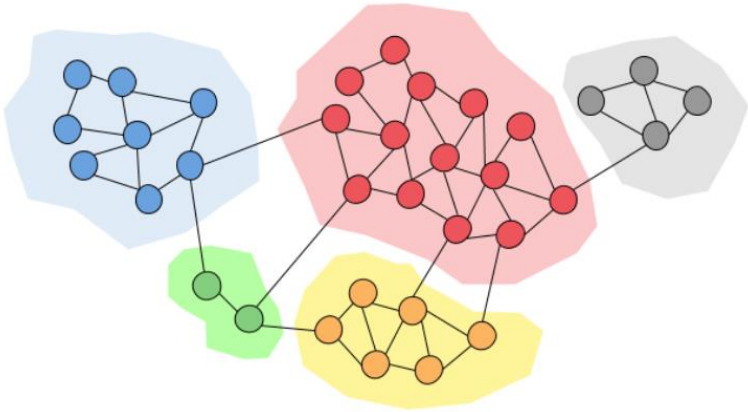
- Community detection method based on the k-nearest neighbor graph
- Clustering results should be mostly consistent with UMAP / tSNE
- Maximize modularity

$$Q = \frac{1}{2m}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[A_{ij} - \frac{k_i k_j}{2m}\right]\delta(c_i, c_j),$$

where:

- A_{ij} represents the edge weight between nodes i and j; see Adjacency matrix;
- k_i and k_j are the sum of the weights of the edges attached to nodes i and j, respectively;
- m is the sum of all of the edge weights in the graph;
- N is the total number of nodes in the graph;
- c_i and c_j are the communities to which the nodes i and j belong; and
- δ is Kronecker delta function:

$$\delta(c_i, c_j) = \begin{cases} 1 & \text{if } c_i \text{ and } c_j \text{ are the same cluster} \\ 0 & \text{otherwise} \end{cases}$$



Louvain clustering (Blondel et. al., Journal of Statistical mechanics, 2008)

- Community detection is similar to clustering but only requires a network
- Maximizing the modularity Q is NP hard

- Louvain algorithm two phases:
  - Step 1: finding local maxima
    - Each node in the network is assigned to its own community and there is a pre-determined order of nodes
    - For each node i, move i to the community of each neighboring node, calculate ΔQ
    - Move i to the community where ΔQ increases most and is positive
    - Go to the next node
    - Stop if no modularity increase can occur
  - Step 2: reduce each community to a single node and build a graph

  - Repeat both steps on the new network and stop if Q can not be increased

Louvain clustering (Blondel et. al., Journal of Statistical mechanics, 2008)

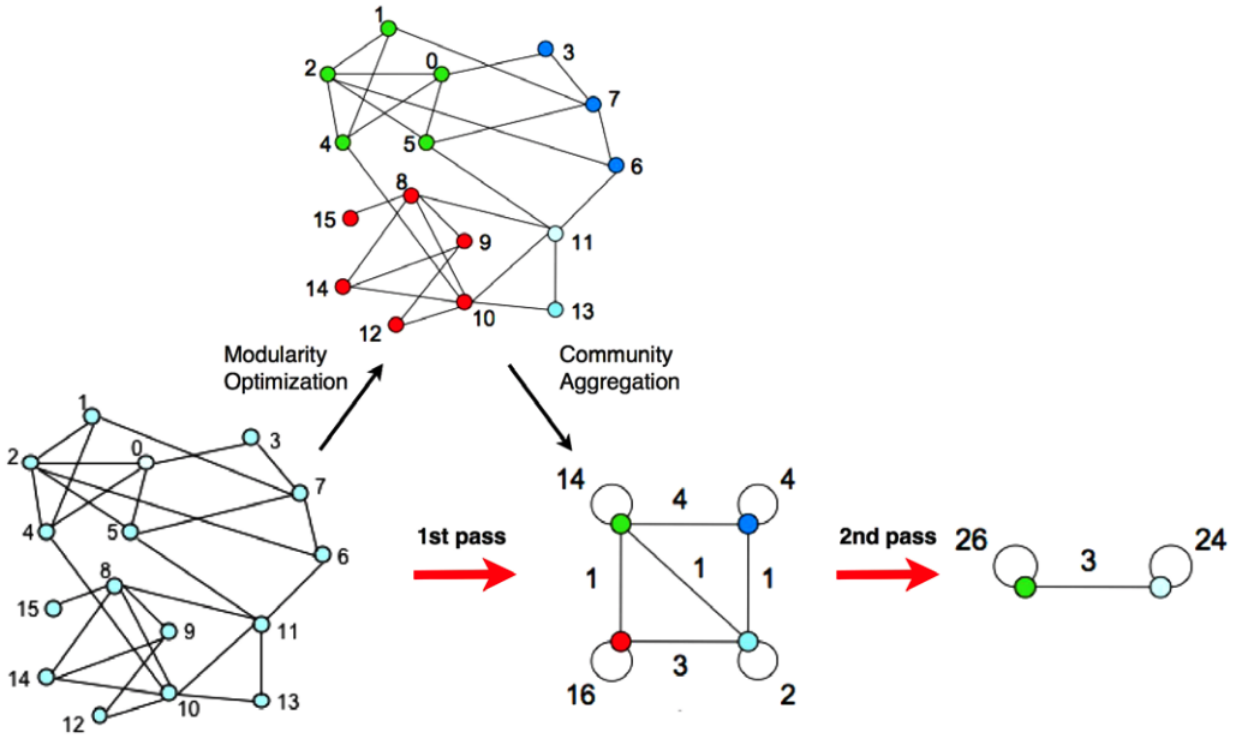


Figure 1. Visualization of the steps of our algorithm. Each pass is made of two phases: one where modularity is optimized by allowing only local changes of communities; one where the communities found are aggregated in order to build a new network of communities. The passes are repeated iteratively until no increase of modularity is possible.

- The algorithm provides a decomposition of the network into communities for different levels of organization
- Computational complexity: linear in # of edges $O(N)$
- Resolution γ (Reichardt and Bornholdt, Physical Review E 2006):

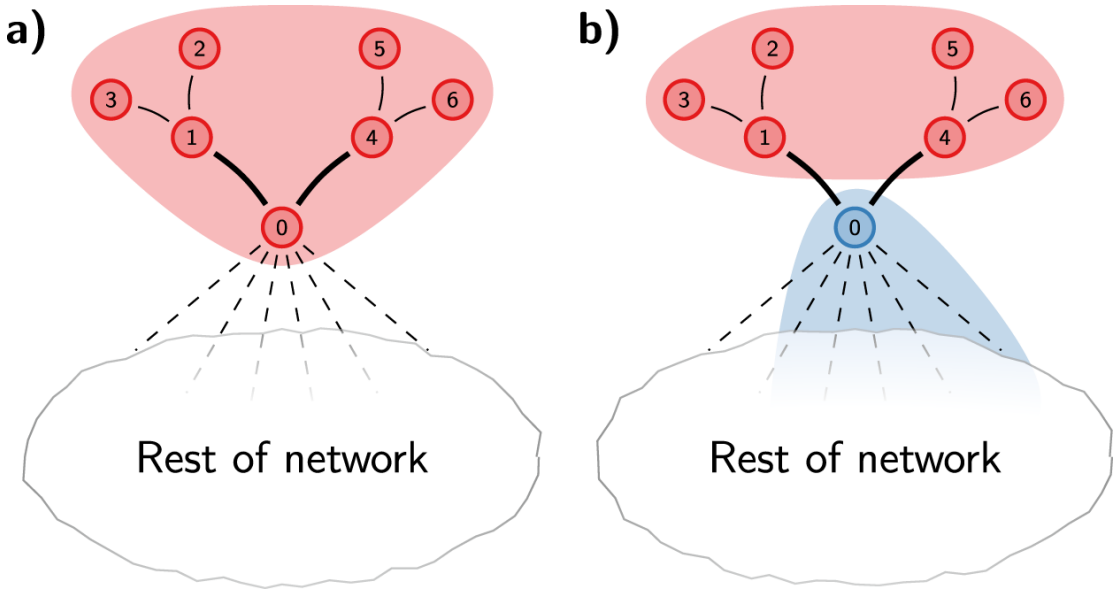
$$Q = \frac{1}{2m}\sum_i \sum_j \left[A_{ij} - \gamma \frac{k_i k_j}{2m}\right] \delta(c_i, c_j)$$

  - Smaller γ -> fewer number of clusters ($\gamma = 0$, one cluster)
  - Can not manually set the number of clusters (automatic determination given γ)

Louvain clustering (Blondel et. al., Journal of Statistical mechanics, 2008)

- Implementation in Seurat
  - Construct weighted graph by KNN after PCA with $k = 20$ by default
  - Weights set Jaccard similarity in the neighbors: proportion of shared overlap in their local neighbors
  - Default resolution 0.8

- Problem of Louvain clustering: may find arbitrarily badly connected communities
  - Only consider individual node movements



Leiden clustering (Traag et. al., Scientific reports, 2019)

- Guarantee that the communities are well connected

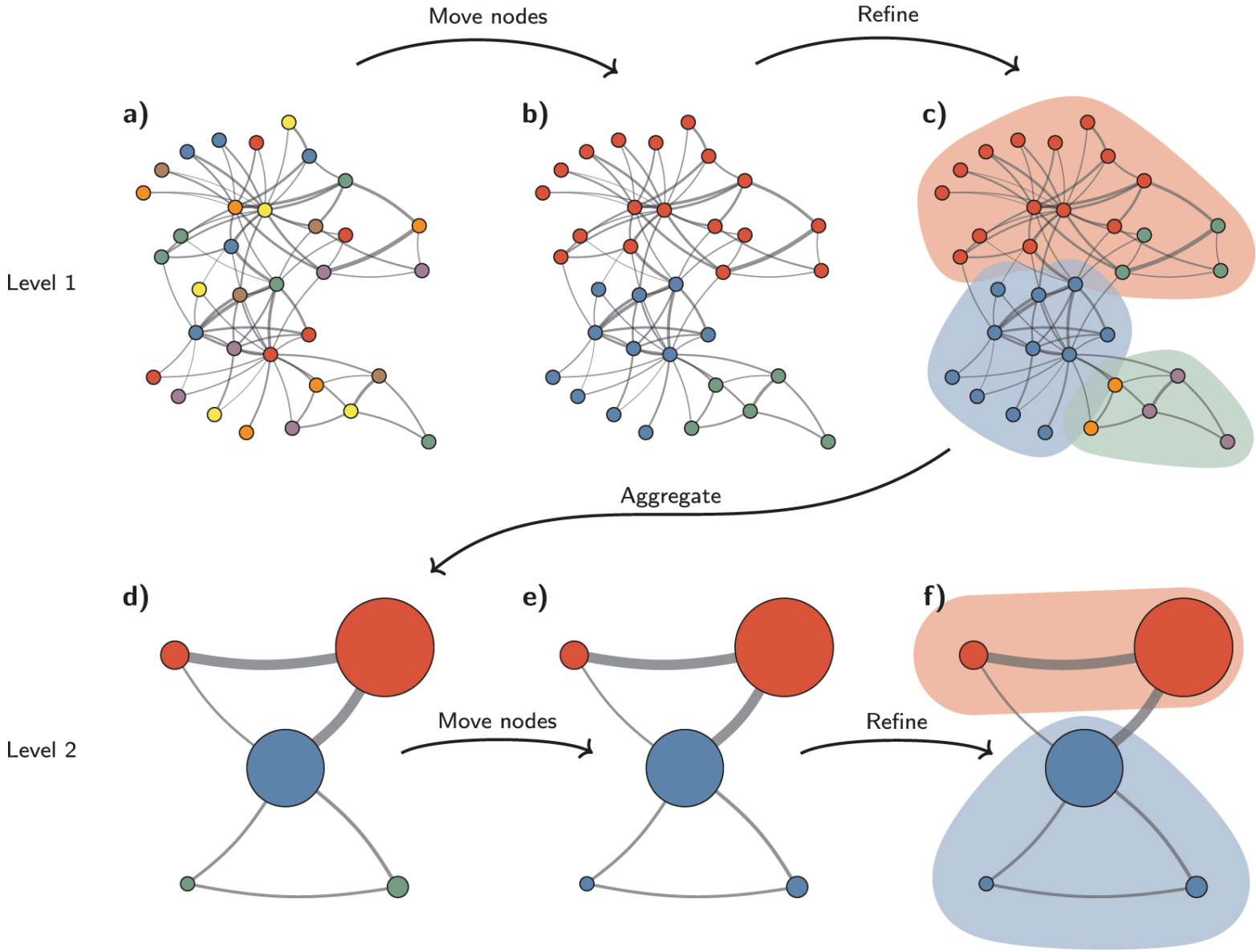
- An updated phase 1 in Leiden clustering:
  - Local moving of the nodes like in the Louvain clustering to get an initial partition $\mathcal{P}$
  - Refinement $\mathcal{P}_{\text{refined}}$ by splitting a community in the initial partition into multiple subcommunities
    - $\mathcal{P}_{\text{refined}}$ starts with a singleton partition
    - Locally merge nodes if they are not on the same community in $\mathcal{P}_{\text{refined}}$ but are within the initial partition $\mathcal{P}$
    - A node randomly select which community to merge among communities that increase Q

- Phase 2: create aggregate network where each node is a community in phase 1

- Computationally faster than Louvain clustering by an improved implementation of local moving phase

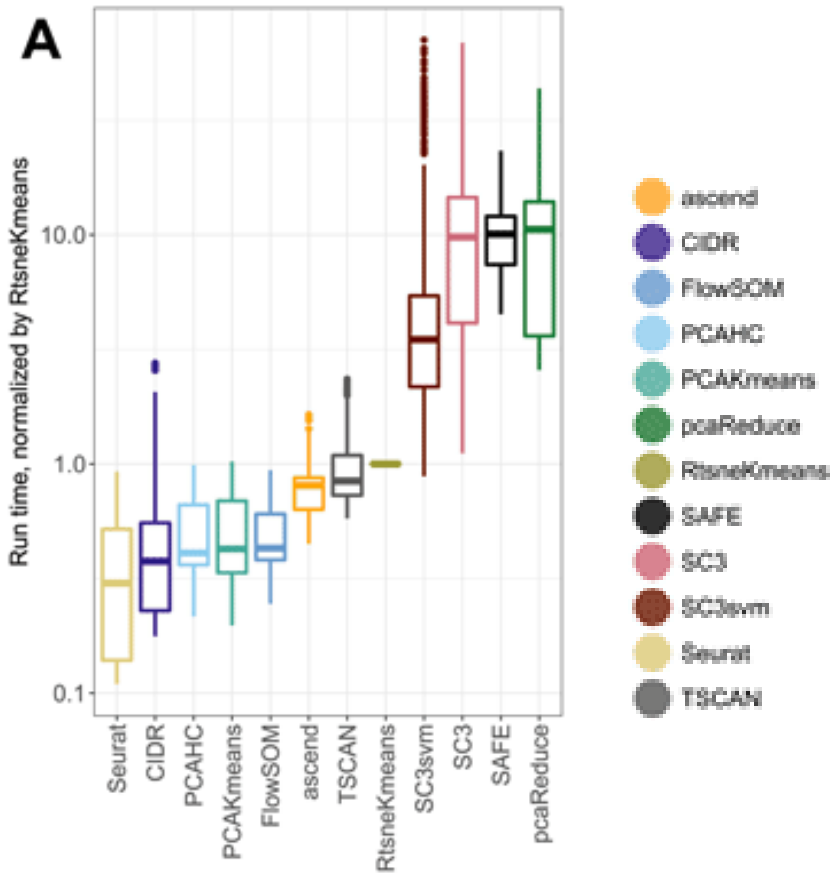
- Default clustering method in Scanpy

Leiden clustering (Traag et. al., Scientific reports, 2019)

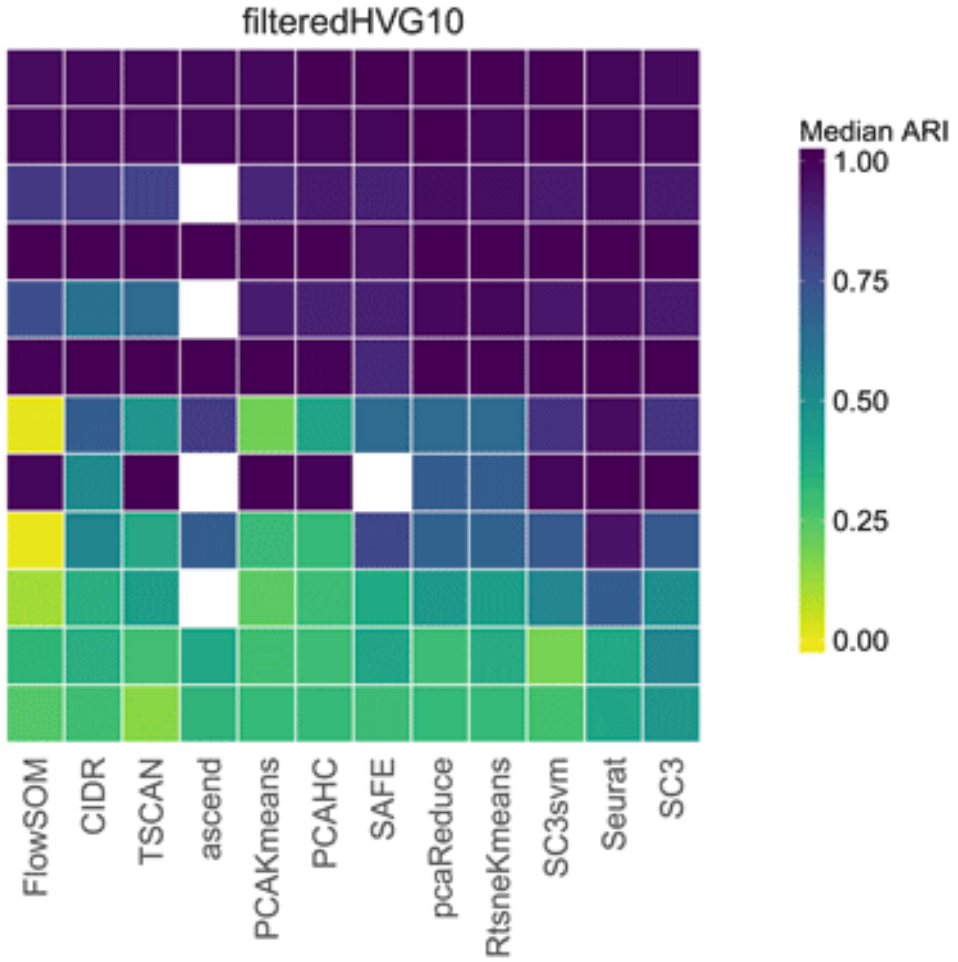


Clustering methods for scRNA-seq

- Benchmarking study (Duo et. al., F1000Research, 2018)



Computational cost



Performance

Adjusted rand index (ARI)

- A measurement comparing clustering results with true labels
- Invariant to permutations of labels

- Rand index

  Given a set of n elements $S = \{o_1, \ldots, o_n\}$ and two partitions of S to compare, $X = \{X_1, \ldots, X_r\}$, a partition of *S* into *r* subsets, and $Y = \{Y_1, \ldots, Y_s\}$, a partition of *S* into *s* subsets, define the following:

  - a, the number of pairs of elements in S that are in the **same** subset in X and in the **same** subset in Y
  - b, the number of pairs of elements in S that are in **different** subsets in X and in **different** subsets in Y
  - c, the number of pairs of elements in S that are in the **same** subset in X and in **different** subsets in Y
  - d, the number of pairs of elements in S that are in **different** subsets in X and in the **same** subset in Y

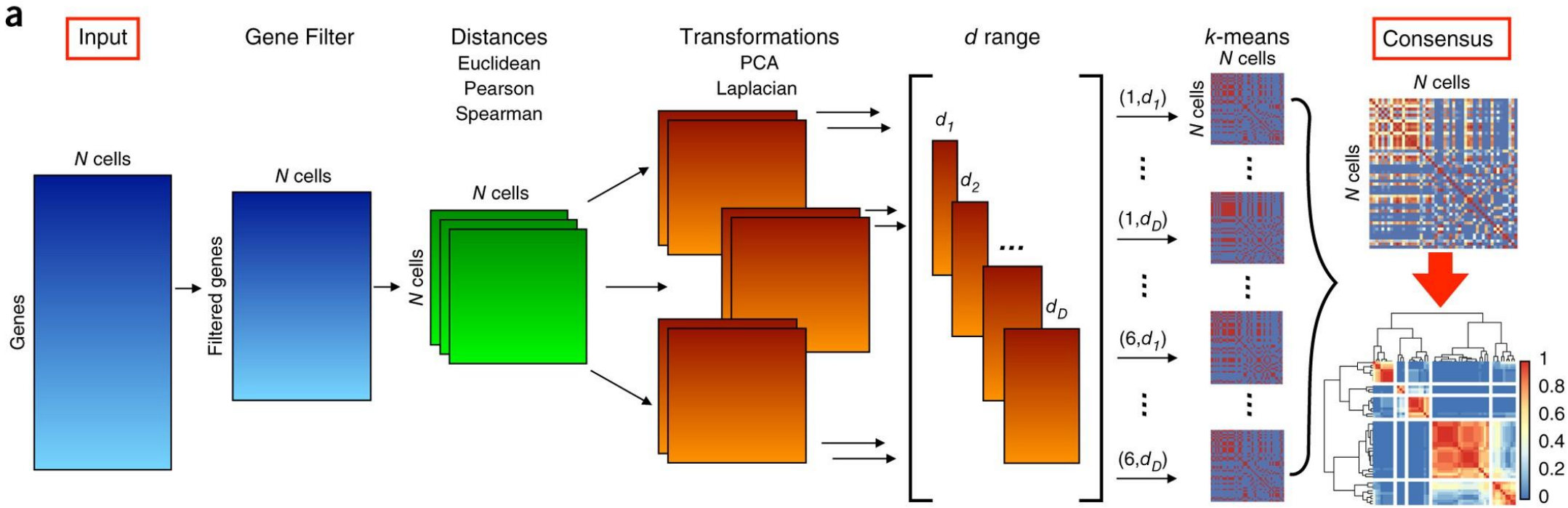
  The Rand index, R, is:[1][2]

$$R = \frac{a+b}{a+b+c+d} = \frac{a+b}{\binom{n}{2}}$$

- Adjusted rand index: adjust by a null model under permutations

SC3 (Kiselev et. al. Nature Methods 2017)

- Run k-means with different data processing methods

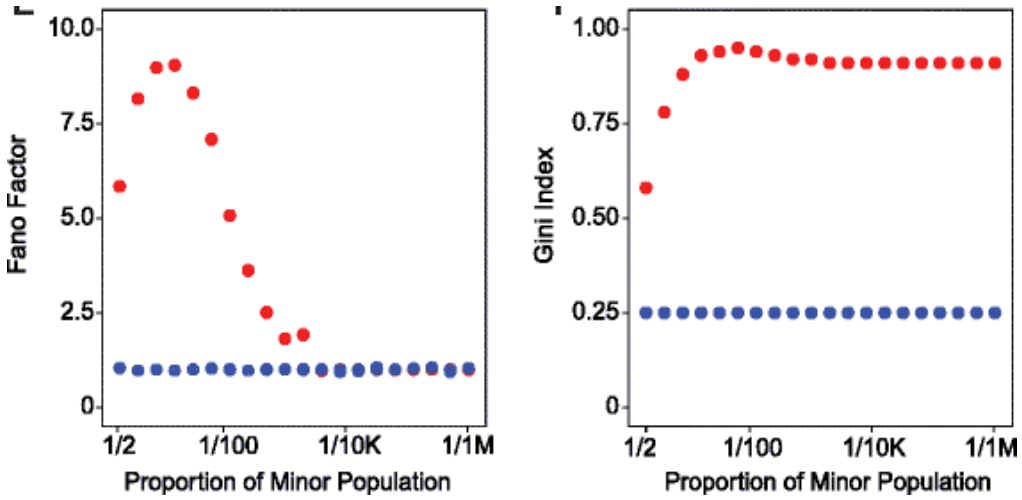


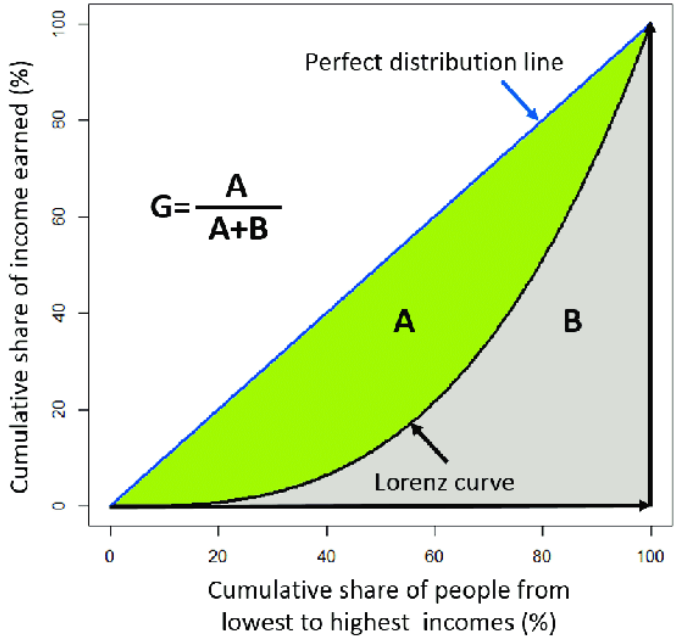
- Get a consensus clustering result across different k-means rounds
  - Calculate cell-cell similarity matrix by the averaging binary similarity matrix across all clustering results
  - Perform hierarchical clustering with complete agglomeration
- Increase robustness compared to a single-round of k-means

Identify rare cell types: GiniClust (3 versions, Yuan group)

- A gene that is only expressed highly in a rare cell type may be filtered out in the HVG selection step

- Then the rare cell type may not be identified as a separate cluster
- Gini index can better identify marker gene for rare cell types
  - Gini index is a robust version of CV
  - Fano factor: σ^2/μ (not scale invariant)





$$G = \frac{\sum_i \sum_j |x_i - x_j|}{2n \sum_i x_i} = \frac{\frac{1}{n^2} \sum_i \sum_j |x_i - x_j|}{2\mu}$$

$$\frac{\sqrt{\frac{1}{n^2} \sum_i \sum_j (x_i - x_j)^2}}{2\mu} = \frac{\sqrt{\frac{1}{n^2} \sum_i \sum_j (x_i - \mu - x_j + \mu)^2}}{2\mu}$$

$$= \frac{\sqrt{2\sigma^2}}{2\mu} = \frac{\mathrm{CV}}{\sqrt{2}}$$

- Consensus clustering using both Gini index and Fano factors

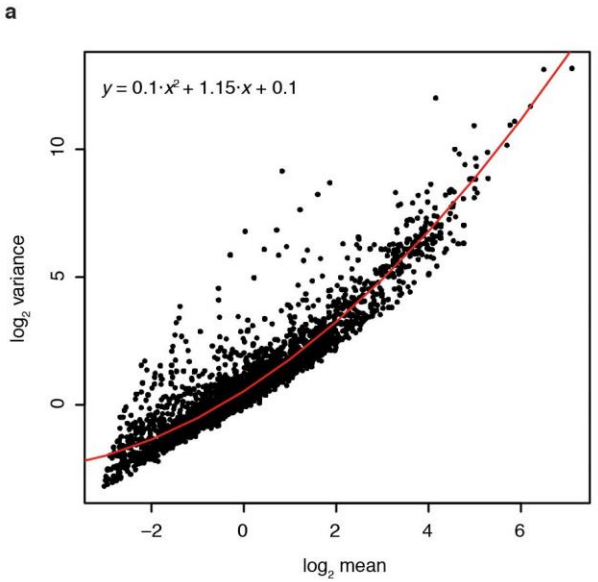
Identify rare cell types: RaceID (Grun et. al., Nature 2015)

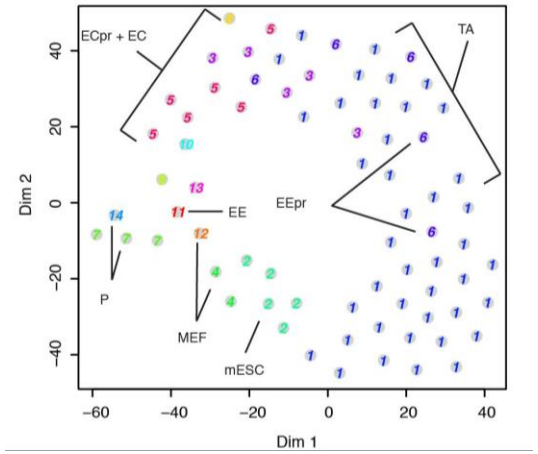
- Rare cell types (tiny clusters) are challenging to identify in a clustering algorithm (like k-means)

- Core idea:
  - Apply a clustering algorithm (k-means)
  - Detect outlier cells within each cluster
    - Fit mean-variance relationship across genes within a cluster
    - Assume that each gene follows a NB distribution, identify cells where expression levels for a few genes (2 by default) are off

  - Outlier cells are further merged to form rare cell type clusters

- Computational cost is relatively high



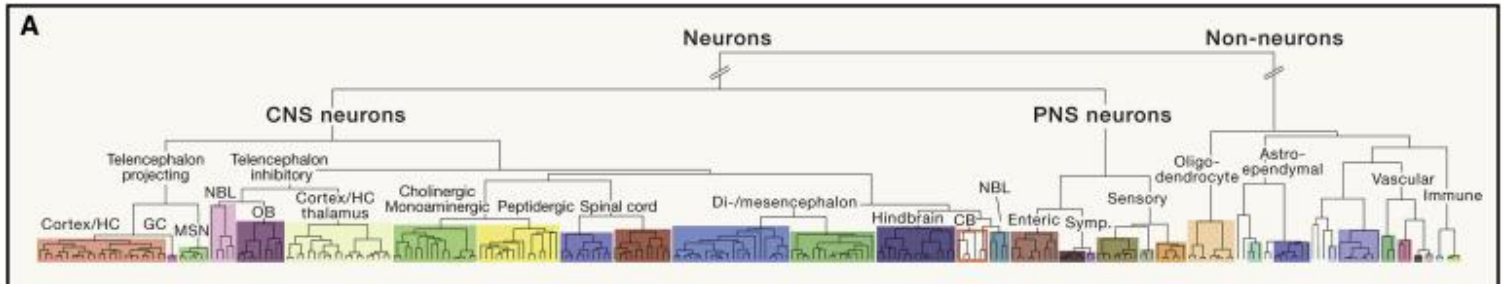


Definition of a cell type

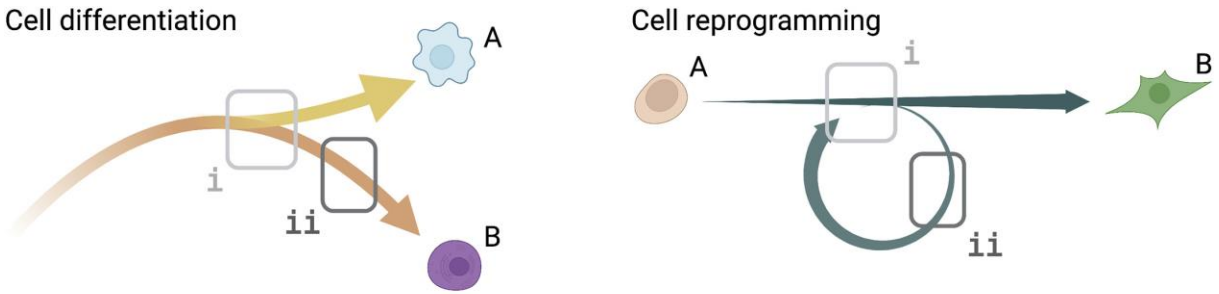
- A cellular phenotype that is robust across datasets, identifiable based on expression of specific markers (i.e. proteins or gene transcripts), and often linked to specific functions

- Partly subjective and can change over time
  - New technologies allow for a higher resolution view of cells
  - Specific “sub-phenotypes” that were not considered biologically meaningful are found to have important biological implications

- Cell types have hierarchical organization (Zeng, Cell 2022)

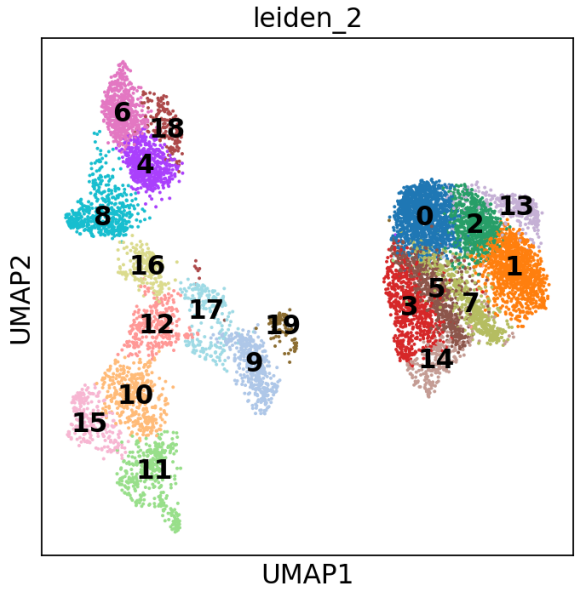


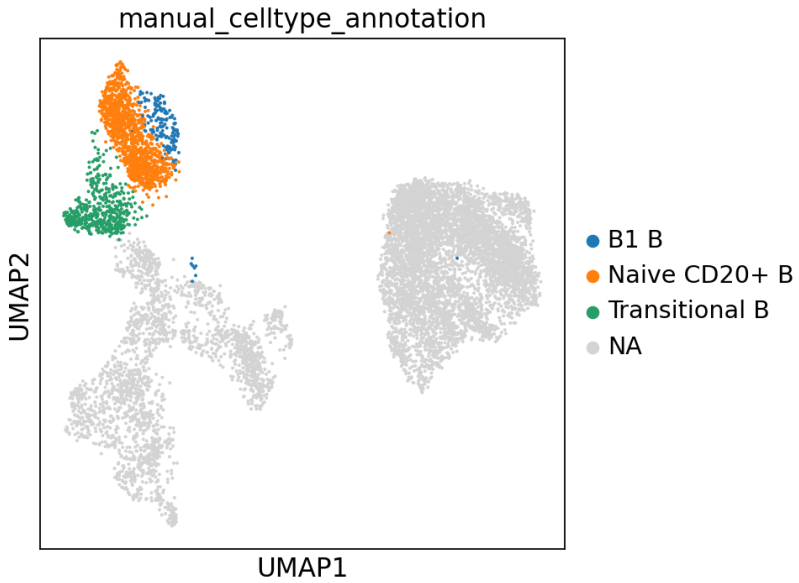
- Dynamic changes of cell types



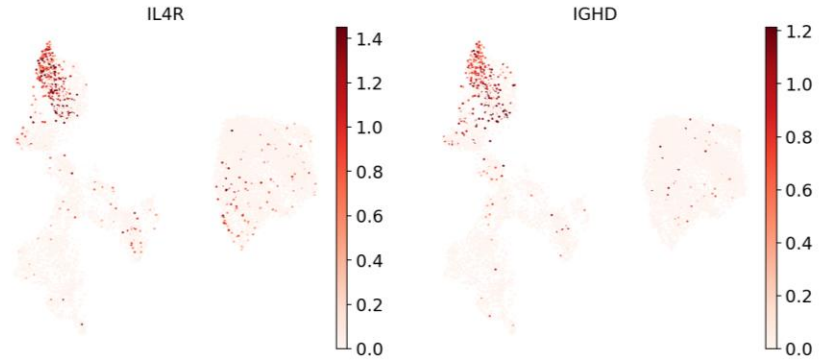
Cell type annotation

- Assign a cell type to each cluster
- Marker genes: Genes that are known to be associated with a particular cell type

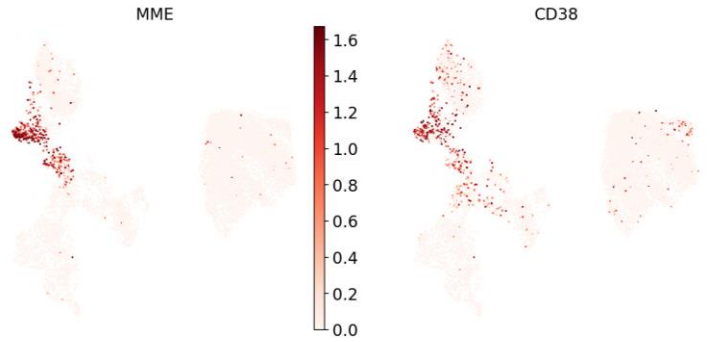




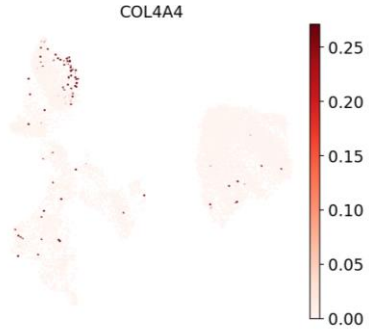
Markers for Naive CD20+ B cells



Markers for Translational B cells



B1 B cells marker



Cell type annotation

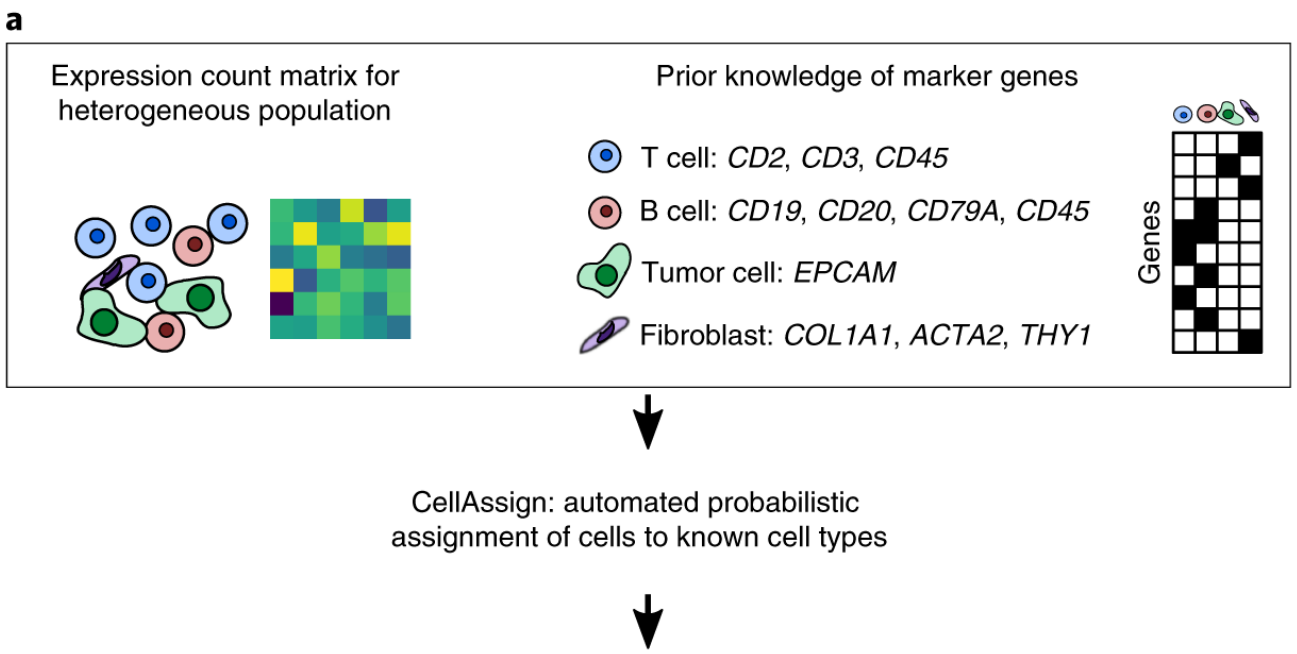
- Manual cell type annotation
  - Visualize known marker genes of major cell types to annotate the clusters
    - Hard to perform if cell types are unknown
  - Identify top differentially expressed genes for each cluster and link those with marker genes
    - Wilcoxon rank-sum test comparing cells in cluster j with (all) other cells
  - Labor intensive and no consensus annotation

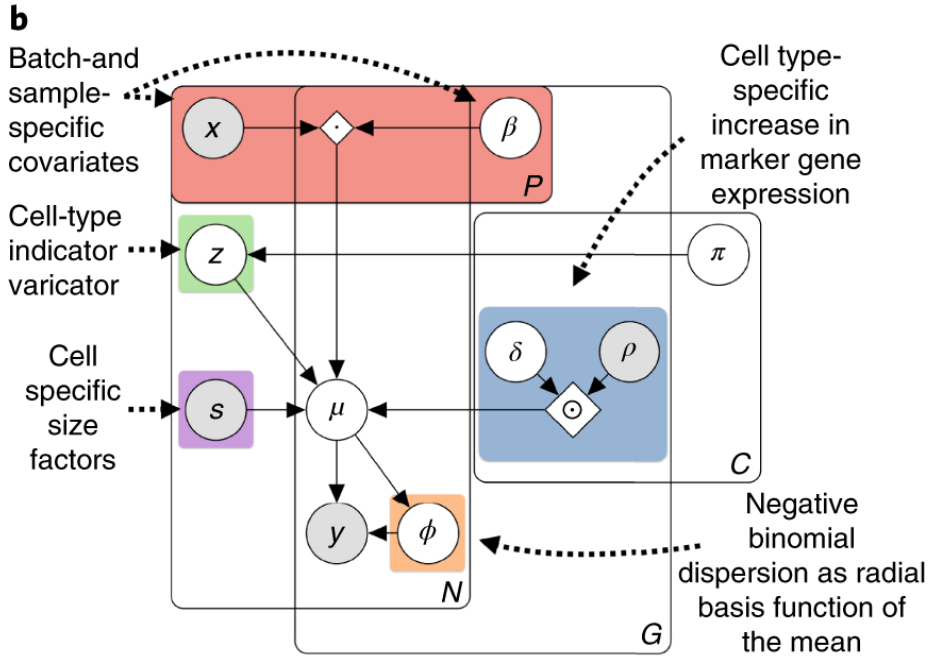
- Automatic cell type annotation
  - Use pre-defined sets of markers
  - Use GPT-4

  - Use pre-existing annotated scRNA-seq data (later lectures)
    - Traditional methods like linear regression and SVM
    - Transfer learning using deep learning

CellAssign (Zhang et. al., Nature Methods 2019)

- Core idea:
  - Define cell markers
  - Build a hierarchical model with latent cell type variables
  - Calculate posterior probabilities that a cell belong to a specific cell type
- One drawback: not using the clustering result





CellAssign (Pliner et. al., Nature Methods 2019)

- The hierarchical model
  - Latent categorical indicator

$$z_n = c \text{ if cell } n \text{ of type } c$$

  - Mixture model :
    - define $\rho_{gc} = 1$ if gene g is a marker for cell

$$\mathbb{E}\left[y_{ng} | z_n = c\right] = \mu_{ngc}$$

where

$$\overbrace{\log \mu_{ngc}}^{\text{Log mean expression}} = \underbrace{\log s_n}_{\text{Cell size factor}} + \overbrace{\delta_{gc}\rho_{gc}}^{\text{Cell type specific}} + \underbrace{\beta_{g0}}_{\text{Base expression}} + \overbrace{\sum_{p=1}^{P} \beta_{gp} x_{pn}}^{\text{Other covariates (incl.batch)}}$$

with the constraint that $\delta_{gc} > 0$.

- Priors for parameters

$$\delta_{gc} \sim \log - \text{normal}\left(\bar{\delta}, \sigma^2\right)$$
$$p\left(z_n = c\right) = \pi_c$$
$$\left(\pi_1, \ldots, \pi_C\right) \approx \text{Dirichlet}\left(\alpha, \ldots, \alpha\right)$$

- Noise model

$$y_{ng} | z_n = c \approx \mathcal{NB}\left(\mu_{ngc}, \bar{\phi}_{ngc}\right)$$

- EM algorithm to solve the model (not easy)

ScType (Lanevski et. al., Nature Communications 2022)

- Largest marker gene database for mouse and human cell types
  - Integrate two existing databases: CellMarker and PanglaoDB

- Core idea:
  - Define cell-type specificity of markers for each tissue t and a specific marker i

$$S_i^t = 1 - \frac{|M_i|_t - min(|M|_t)}{max(|M|_t) - min(|M|_t)}$$

  - Perform clustering and assign a cell type to a cluster
    - Transform data into z-scores and multiply by the cell-type specificity
    - Calculate marker-enrichment-score for a specific cell type and a specific cell: an average contrast between marker genes and negative marker genes for each cell and each cell type

$$\mathbf{x}'_c = \frac{\sum_{i=1}^{j} x'_i}{\sqrt{j}} - \frac{\sum_{k=1}^{l} x'_k}{\sqrt{l}}$$

    - Sum up the contrast across all cells in the cell cluster, and assign the cluster with the cell type that has the largest score

GPTCelltype (Hou and Ji, Nature Methods 2024)

b

Identify cell types of human prostate cells using the following markers. Identify one cell type for each row. Only provide the cell type name.

CR2, CD24, FAS, CXCR3, CD1c
KLK3, KRT8, KLK2, MSMB, ACPP, KLK1, KLK4
MMRN1, FLT4, RELN, CCL21, PROXl, LYVE1
TPSAB1, FCER1A, TPSB2, KIT, CD69, HDC
ACTA2, MY01B, ACTA2, ANPEP, DES, MCAM, PDGFRB, CSPG4

1. Dendritic cells
2. Luminal epithelial cells
3. Lymphatic endothelial cells
4. Mast cells
5. Pericytes

Be more specific

1. Myeloid dendritic cells (mDCs)
2. Prostate secretory luminal epithelial cells
3. Prostate lymphatic endothelial cells
4 . Prostate tissue-resident mast cells
5. Prostate vascular pericytes

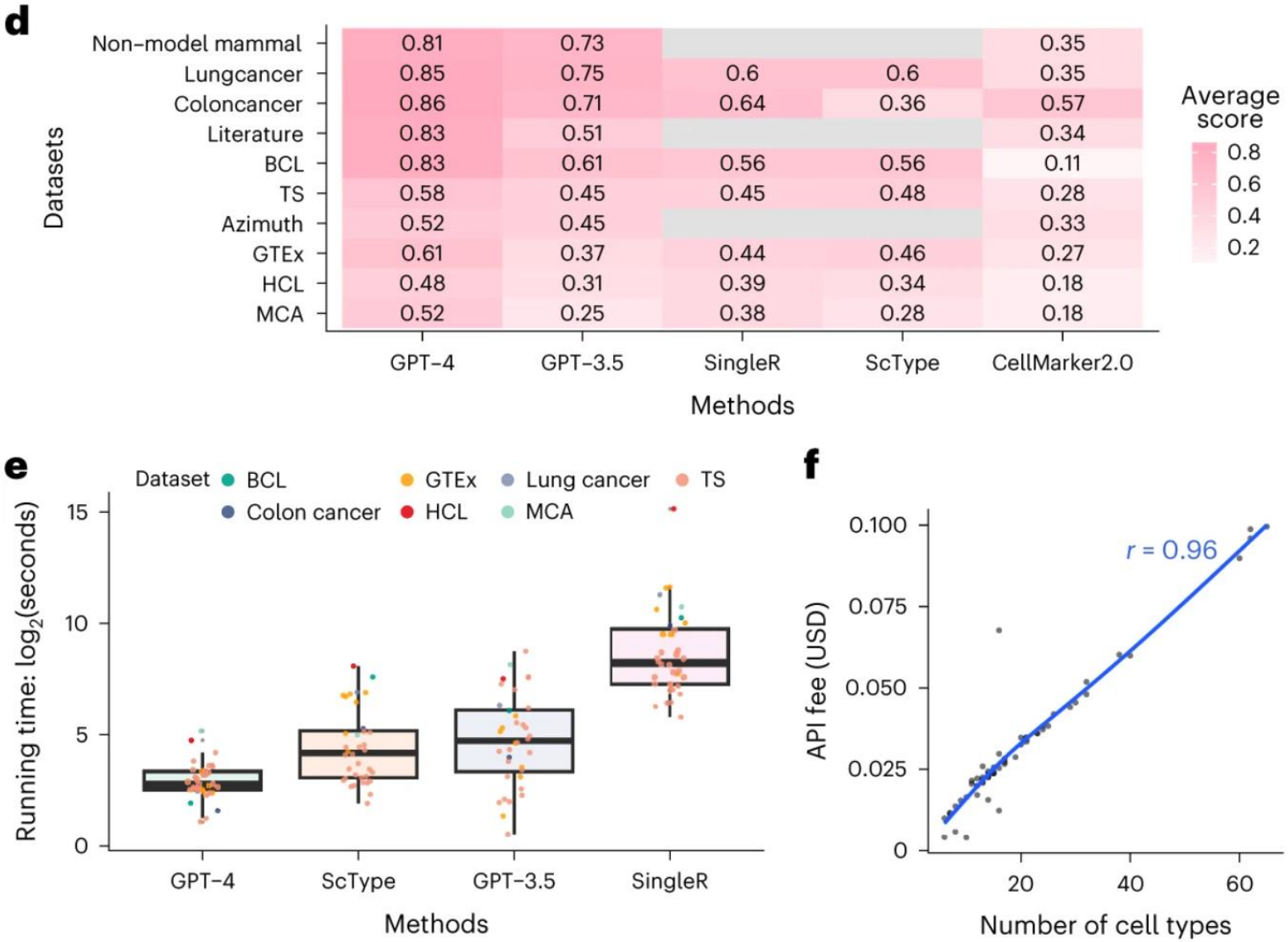
Identify cell types of human prostate cells using the following markers. Identify one cell type for each row. Only provide the cell type name. Some could be a mixture of multiple cell types. Some could be unknown cell types.

KLK3, KRT8, KLK2, MSMB, ACPP, KLK1, KLK4
MMRN1, FLT4, RELN, CCL21, PROX1, LYVE1
CD69, IL7R,, CD3D, CD3E, CD3G,ACTA2, MYO1B, ACTA2, ANPEP, PDGFRB, CSPG4
DDX49,LOC105371196,MTND1P30,LOC105373682,TAGLN2,ZNF836,ZNF677,COILP1

1. Prostate epithelial cells
2. Lymphatic endothelial cells
3. T cell and smooth muscle cell mixture
4. Unknown cell type

- The package automatically generate prompt message
- GPT-4 is able to identify unknown cell types
- Systematic benchmarking in the paper

GPTCelltype (Hou and Ji, Nature Methods 2024)



Related papers

- Blondel, V. D., Guillaume, J. L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. Journal of statistical mechanics: theory and experiment, 2008(10), P10008.
- Reichardt, J., & Bornholdt, S. (2006). Statistical mechanics of community detection. Physical review E, 74(1), 016110.
- Traag, V. A., Waltman, L., & Van Eck, N. J. (2019). From Louvain to Leiden: guaranteeing well-connected communities. Scientific reports, 9(1), 5233.
- Duò, A., Robinson, M. D., & Soneson, C. (2018). A systematic performance evaluation of clustering methods for single-cell RNA-seq data. F1000Research, 7.
- Kiselev, V. Y., Kirschner, K., Schaub, M. T., Andrews, T., Yiu, A., Chandra, T., ... & Hemberg, M. (2017). SC3: consensus clustering of single-cell RNA-seq data. Nature methods, 14(5), 483-486.
- Dong, R., & Yuan, G. C. (2020). GiniClust3: a fast and memory-efficient tool for rare cell type identification. BMC bioinformatics, 21, 1-7.
- Grün, D., Lyubimova, A., Kester, L., Wiebrands, K., Basak, O., Sasaki, N., ... & Van Oudenaarden, A. (2015). Single-cell messenger RNA sequencing reveals rare intestinal cell types. Nature, 525(7568), 251-255.

- Zhang, A. W., O'Flanagan, C., Chavez, E. A., Lim, J. L., Ceglia, N., McPherson, A., ... & Shah, S. P. (2019). Probabilistic cell-type assignment of single-cell RNA-seq for tumor microenvironment profiling. Nature methods, 16(10), 1007-1015.
- Ianevski, A., Giri, A. K., & Aittokallio, T. (2022). Fully-automated and ultra-fast cell-type identification using specific marker combinations from single-cell transcriptomic data. Nature communications, 13(1), 1246.
- Hou, W., & Ji, Z. (2024). Assessing GPT-4 for cell type annotation in single-cell RNA-seq analysis. Nature Methods, 1-4.