Single-cell and spatial transcriptomics analysis in R

Prof. Dr. Michael Altenbuchinger

December the 18th, 2024

Learning objectives: learn to analyze single-cell data in R, get a first glimpse on spatial transcriptomics

(1) Familiarize yourself with Seurat

Different analysis pipelines are available for the analysis of single-cell data, of which Seurat is among the most popular ones. It offers rich functionality and comprehensive tutorials to get used to it. In this first exercise, use the available Seurat vignettes and tutorials and do the following analysis (see `https://satijalab.org/seurat/articles/pbmc3k_tutorial`). Your can directly use the example data provided as "data_Seurat.zip". You should familiarize yourself with the following analysis steps:

- Install and load the R packages "Seurat" and "dplyr".
- Load and inspect the data.
- Perform quality controls. Retrieve the proportion of mitochondrial RNA. What does it tell you?
- Filter your data: (1) filter cells that have unique feature counts over 2,500 or less than 200, and (2) filter cells that have $> 5\%$ mitochondrial counts.
- Normalize your data (normalization + log-transformation). How are these steps performed?
- Reduce your data set to the 2000 most variable genes.
- Scale your data. How is this done?
- Familiarize yourself with dimension reduction and clustering.
- Screen for differentially expressed features (cluster biomarkers).
- Assign clusters with cell identities.

(2) Do your first own single-cell analysis

R Tutorials give you a first idea how to use packages. However, when doing your own analysis, you typically encounter unexpected issues. Next, perform a comprehensive single-cell analysis in line with exercise 1 but now for the data provided as "data_new.zip". An additional difficulty which arises is that the two different data sources are to be integrated.

(a) Read in both data sets and perform quality controls; filter cells with $> 10\%$ mitochondrial RNAs. To read "*.h5" files, you can use the library "hdf5r". You might need the following commands: `Read10X_h5()`, `CreateSeuratObject()`, `PercentageFeatureSet(), subset()`

(b) Normalize your data (`NormalizeData()`).

(c) Next, we need to integrate both data resources. To do so, we will use so-called anchors. Relevant commands are: `FindIntegrationAnchors()`, `IntegrateData()`

(d) Visualize the number of expressed genes per cell, cell counts and the percentage of mitochondrial RNAs per sample (`VlnPlot()`).

(e) Identify highly variable genes and plot gene variances vs. means (`FindVariableFeatures()`, `VariableFeaturePlot()`), `LabelPoints()`)

(f) Scale your data and perform dimension reduction by PCA (`ScaleData()`,`RunPCA()`). Plot the first two principal components (`DimPlot`). Visualize the loadings (`VizDimLoadings()`). What do they mean?

(g) Determining an optimal number of PCs to be included into the downstream analysis can be time intense (`JackStraw()`, `JackStrawPlot()`, `ElbowPlot`). Alternatively, simply use the top 30.

(h) Cluster your cells using UMAP (`FindNeighbors()`, `FindClusters()`, `RunUMAP()`). Verify that your data sets were correctly merged!!! This can be done visually (`DimPlot()`).

(i) Retrieve genes representing your clusters (`FindAllMarkers()`).

(j) Visualize the clustering (`RenameIdents()`, `DimPlot()`). Also visualize important cell markers (`FeaturePlot()`): "MS4A1", "PECAM1", "EPCAM", "PDGFRB", "CD68", "JCHAIN", "MKI67", "CD3D". To which cell types do they correspond?

(k) It can be tough to label all clusters and it usually requires a lot of expert knowledge. However, resources such as `www.proteinatlas.org` can help you. Try to label your clusters.

(l) You usually do not stop at this point, but try to infer additional sub-populations. If time permits, choose a cluster and try make the labeling more precise. For this, your need to repeat your clustering on the sub-population!

(m) Infer single-cell trajectories. Different packages are available with different pros and cons. Example are the libraries "monocle3" and "destiny". Select your method of choice and infer single-cell trajectories. Illustrate your results! Hint: it might make sense to use a data set represented by the leading principal components to reduce computational burden.

(3) A first glimpse at spatial transcriptomics

Spatial transcriptomics (ST) facilitates studies of spatial RNA distributions. This further complicates data analysis. However, also for ST data, there are first pipelines available. Since you already gained first experience with using Seurat, explore its capabilities for ST data analysis. To do so, you can screen the package vignettes or search the internet. Example ST data are provided as "data_new_2.rds". Load, inspect and visualize the data and perform first quality controls (`readRDS()`, `str()`, `typeof()`, `VlnPlot()`, `SpatialFeaturePlot()`).