Analyzing scRNAseq data with RCA

Huipeng Li

20/03/2017

Introduction

In this tutorial, RCA will be applied to analyze two single cell RNAseq datasets from a recent publication (http://dx.doi.org/ 10.1038/ng.3818). The input data can be downloaded from Gene Expression Omnibus (GEO) with accession GSE81861. The corresponding raw sequencing data can be accessible on European Genome Phenome Archive (EGA) under accession EGAS00001001945.

Section 0: Preparing environment and data

A few libraries need to be loaded before running RCA.

```
### loading libraries
library(WGCNA)
```

```
## ======================================================================
## *
## *  Package WGCNA 1.34 loaded.
## *
## *    Important note: It appears that your system supports multi-threading,
## *    but it is not enabled within WGCNA in R.
## *    To allow multi-threading within WGCNA with all available cores, use
## *
## *          allowWGCNAThreads()
## *
## *    within R. Use disableWGCNAThreads() to disable threading if necessary.
## *    Alternatively, set the following environment variable on your system:
## *
## *          ALLOW_WGCNA_THREADS=<number_of_processors>
## *
## *    for example
## *
## *          ALLOW_WGCNA_THREADS=8
## *
## *    To set the environment variable in linux bash shell, type
## *
## *           export ALLOW_WGCNA_THREADS=8
## *
## *     before running R. Other operating systems or shells will
## *     have a similar command to achieve the same aim.
## *
## ======================================================================
```

```
library(flashClust)
library(gplots)
library(preprocessCore)
library(RCA)
```

```
### setting options
options(stringsAsFactors = FALSE)
```

Download the expression data from GEO and save them in the working directory. Then read the csv files into R as data frames.

```
data_cellline = read.csv("../data/GSE81861_Cell_Line_FPKM.csv",row.names=1);
# data_NM_all = read.csv("../data/GSE81861_CRC_NM_all_cells_FPKM.csv",row.names=1);
# data_tumor_all = read.csv("../data/GSE81861_CRC_tumor_all_cells_FPKM.csv",row.names=1);
# data_NM_epi = read.csv("../data/GSE81861_CRC_NM_epithelial_cells_FPKM.csv",row.names=1);
# data_tumor_epi = read.csv("../data/GSE81861_CRC_tumor_epithelial_cells_FPKM.csv",row.names=1);
```

Section 1: Clustering a cellline dataset involving multiple cell types and multiple batches.

Define the input dataset, and extract color code from column names.

```
fpkm_data = data_cellline;
color_to_use0 = colnames(fpkm_data);
color_to_use0 <- strsplit(color_to_use0,"__");
color_to_use <- paste("",lapply(color_to_use0,"[[",3),sep="");
color_to_use=gsub("\\.","#",color_to_use);
```

Run RCA analysis.

```
### construct data object
data_obj = dataConstruct(fpkm_data);
### filt out lowly expressed genes
data_obj = geneFilt(obj_in = data_obj);
### normalize gene expression data (Note: default is no normalization).
data_obj = cellNormalize(data_obj);
### log transform the data
data_obj = dataTransform(data_obj,"log10");
### project the expression data into Reference Component space
data_obj = featureConstruct(data_obj,method = "GlobalPanel");
### generate cell clusters
data_obj = cellClust(data_obj);
```

```
##  ..cutHeight not given, setting it to 0.886  ===>  99% of the (truncated) height range in dendro.
##  ..done.
```

Plot RCA results.

```
RCAPlot(data_obj,cluster_color_labels = color_to_use);
```

```
## pdf
##   2
```

Fig1A and Fig1B show the scatter plot and heat map of RCA clusters, labeled with predefined cell types (colors).

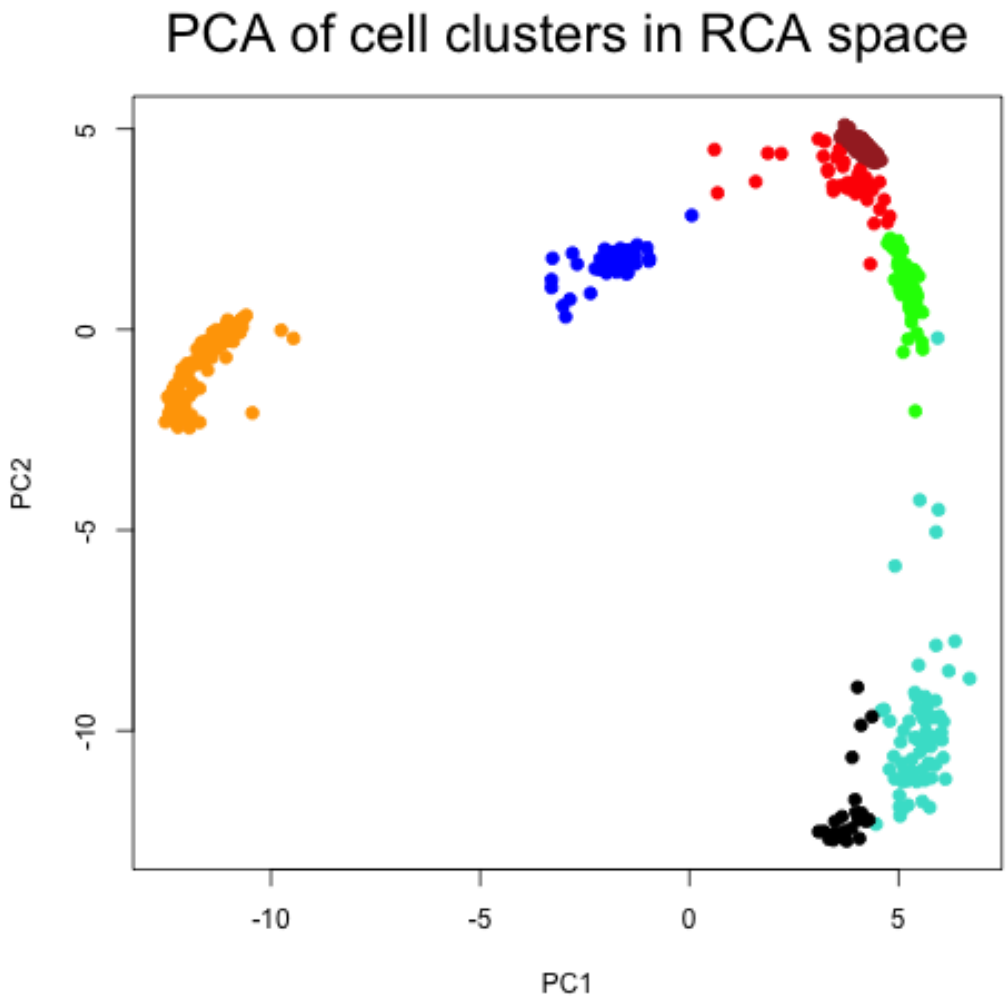

Figure 1: Fig 1A: RCA clusters of cell line data in PCA space. Each color represents one cell type.

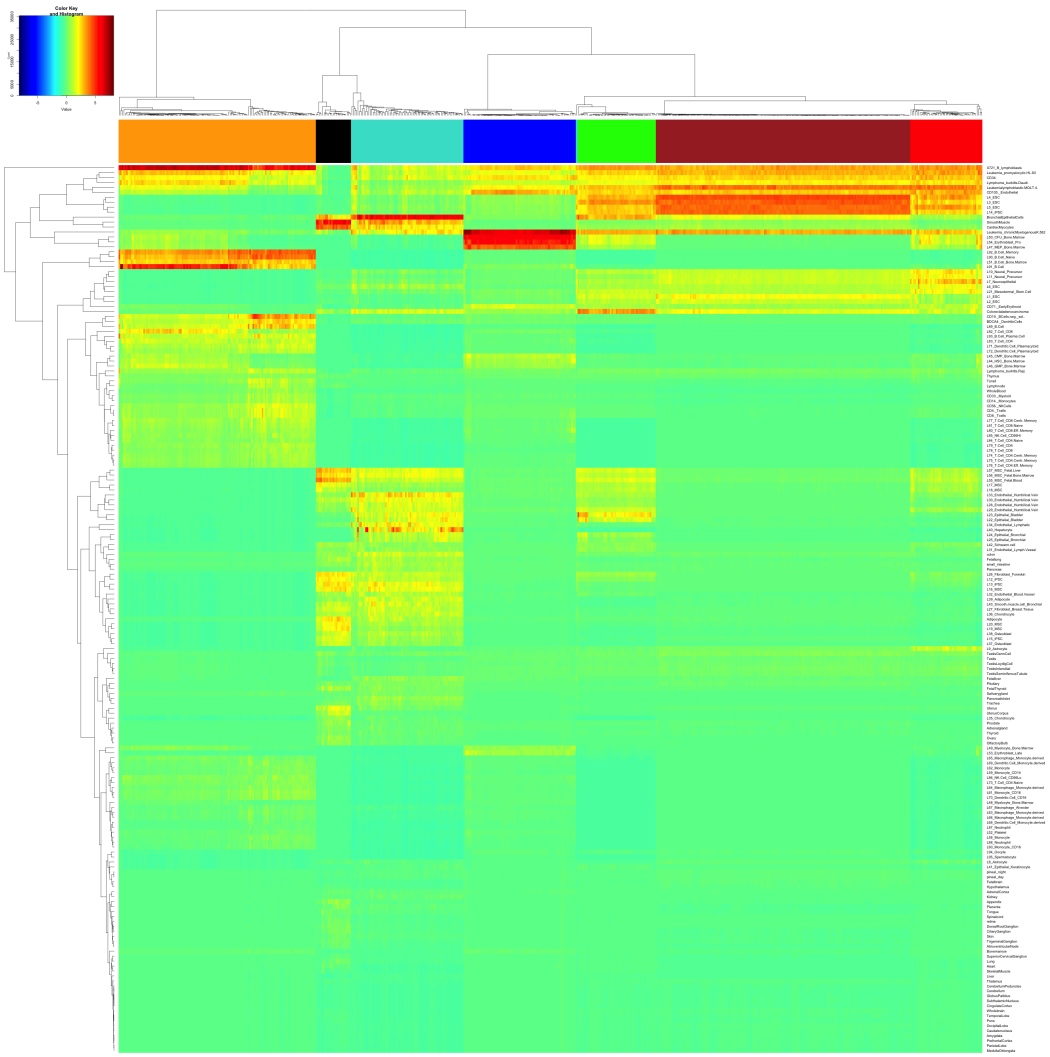

Figure 2: Fig 1B: Heat map showing reference component scores, with rows representing reference profiels and columns representing single cells.