Understanding RNA-seq analysis through an example: voting behavior at the French parliament

Hector Roux de Bézieux

May 2018

Introduction

Single-cell RNA-seq (scRNA-seq) is a very potent biological tool used for many applications. A far from exhaustive list would include identifying new cell types, finding differentially expressed (DE) genes, and discovering lineages among cells. However, the usual framework might seem a little daunting for beginners and, while many well-crafted tutorials exists, they all share the same idea: use a biological dataset as an example. Here, we want to use a dataset that would be more understandable to a broader public to explain the usual steps in scRNA-seq analysis.

Context

Our dataset is the voting record of the French delegates of the 14th legislature, going from May 2012 to May 2017. Delegates are characterized by their names and surnames, their department and circonscription, their political group. For each of the 644 votes (which can be laws, amendments, choosing the prime minister, . . .), we also have the voting behavior of each delegate (voted yes, no, abstain or did not took part in the vote) and the reason of the vote.

Loading the data and building an ExpressionSet object

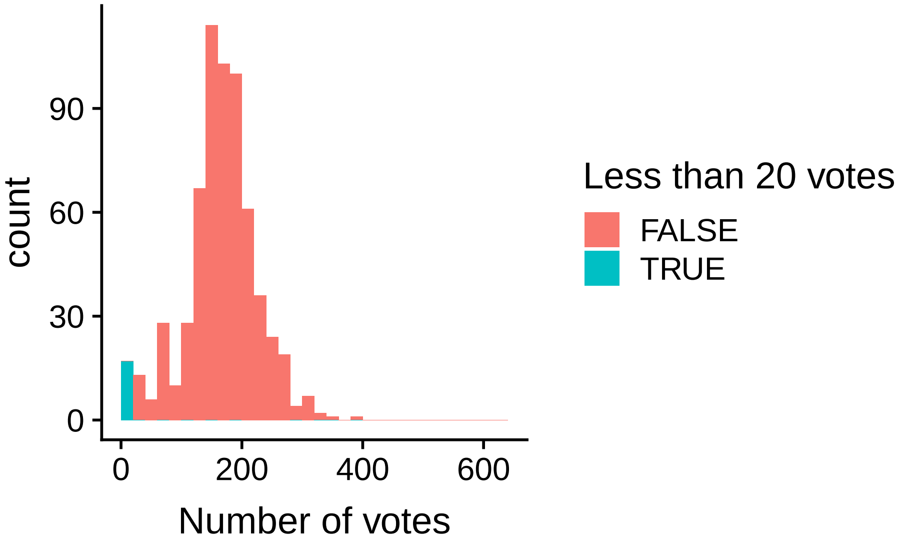
We store the data in an *ExpressionSet* object, where the *phenodata* is the information about the cells (the delegate) and the *featuredata* is the information about the genes (the votes).

```r
voting_record <- read_csv("data/voting_record.csv")
meta <- voting_record %>% select(circo, dept, name, surname, identifiant,
                                 iden, group, NbVote, NbYes, NbNo, NbAbst)
voting_record <- voting_record %>% select(-one_of(colnames(meta))) %>%
                                   t(.)
votes <- read_csv("data/votes.csv")
colnames(voting_record) <- rownames(meta) <- meta$identifiant
rownames(voting_record) <- rownames(votes) <- votes$Number
Assay <- ExpressionSet(assayData = voting_record,
                       phenoData = AnnotatedDataFrame(meta) ,
                       featureData = AnnotatedDataFrame(votes))
```

Filtering the cells (delegate)

```r
Nb_Votes <- data.frame(Nb_Votes = colSums(!is.na(exprs(Assay))),
                       delegate = rownames(phenoData(Assay)))
```

```r
ggplot(Nb_Votes, aes(x = Nb_Votes, fill = Nb_Votes <= 20)) +
                 stat_bin(breaks = seq(0, max(Nb_Votes$Nb_Votes), 20)) +
                 labs(x = "Number of votes") +
                 guides(fill = guide_legend(title = "Less than 20 votes"))
```



Thanks

Thanks to Vincent Viers for the initial inspiration of this project.

The code used for scraping the data is based on the blog post https://freakonometrics.hypotheses.org/50973.