Analiza Transkryptomu - Zadanie 1 (Normalizacja bibliotek)

Ksenia Kvitko

3.04.2020

1. Wczytanie i weryfikacja danych z pliku

```
raw_data <- read.csv("..\\source_files\\counts.txt", sep = "\t", skip = 1)
dim(raw_data)
```

```
## [1] 37834     9
```

```
head(raw_data)
```

```
##         Geneid
## 1 LOC109343272
## 2 LOC109343320
## 3 LOC109343262
## 4 LOC109343339
## 5 LOC109343296
## 6 LOC109343328
##
## 1                                                                                                       NC_
## 2                                               NC_032009.1;NC_032009.1;NC_032009.1;NC_032009.1;NC_
## 3                                                           NC_032009.1;NC_032009.1;NC_032009.1;NC_
## 4
## 5 NC_032009.1;NC_032009.1;NC_032009.1;NC_032009.1;NC_032009.1;NC_032009.1;NC_032009.1;NC_032009.1;NC_
## 6                                                           NC_032009.1;NC_032009.1;NC_032009.1;NC_
##                                                                                    Start
## 1                                                            616;816;1102;1755;2518;3449
## 2                                         4875;5322;5533;5997;6341;6859;6989;7381;7560;7907
## 3                                     12168;13831;14462;14704;14941;15310;15729;16084;16475
## 4                                                                                  17455
## 5 18551;18946;19103;19308;19510;19755;19915;20568;20811;20965;21202;21728;21890;30866
## 6                                     23810;23970;24165;24389;24547;24896;25061;25249;25537
##                                                                                      End
## 1                                                           690;1013;1508;1862;3123;4199
## 2                                         5224;5424;5577;6095;6409;6918;7082;7455;7840;8123
## 3                                     12377;14376;14557;14773;15223;15650;15996;16356;16947
## 4                                                                                  18263
## 5 18863;19006;19197;19420;19678;19794;19979;20681;20875;21121;21321;21816;21979;30876
## 6                                     23899;24086;24289;24453;24759;24943;25168;25401;25764
##                         Strand Length bam.flower.bam bam.stem.bam bam.leaf.bam
```

```
## 1                 +;+;+;+;+;+;+   2145            41          189          410
## 2         -;-;-;-;-;-;-;-;-;-;-   1393             9           19           14
## 3           +;+;+;+;+;+;+;+;+;+   2560             0            0            0
## 4                             +    809             0            0            0
## 5 +;+;+;+;+;+;+;+;+;+;+;+;+;+;+   1502             4            0            0
## 6           +;+;+;+;+;+;+;+;+;+   1147            94            2            0
```

2. Wybranie interesujących kolumn, ustalenie nomenklatury, weryfikacja kompletności danych

```
geneLengths <- raw_data[, c(1, 6:9)]
dim(geneLengths)
```

```
## [1] 37834     5
```

3. Normalizacja metodą TPM

Krok I - normalizacja względem długości genu

```
TPM_step1 <- geneLengths
TPM_step1$bam.flower.bam <- TPM_step1$bam.flower.bam / TPM_step1$Length
TPM_step1$bam.stem.bam <- TPM_step1$bam.stem.bam / TPM_step1$Length
TPM_step1$bam.leaf.bam <- TPM_step1$bam.leaf.bam / TPM_step1$Length
```

Krok II - normalizacja względem wielkości biblioteki

```
TPM_step2 <- TPM_step1
TPM_step2$bam.flower.bam <- TPM_step2$bam.flower.bam / (sum(TPM_step2$bam.flower.bam) / 1000000)
TPM_step2$bam.stem.bam <- TPM_step2$bam.stem.bam / (sum(TPM_step2$bam.stem.bam) / 1000000)
TPM_step2$bam.leaf.bam <- TPM_step2$bam.leaf.bam / (sum(TPM_step2$bam.leaf.bam) / 1000000)
```

4. Rozwiązanie zadania

Utworzenie tabeli oraz wektorów

```
dane_TPM <- TPM_step2[,c(1,3:5)]
colnames(dane_TPM)[2:4] <- c("liść_TPM", "pęd_TPM", "kwiat_TPM")
```

Podanie liczby wymiarów - finalna tabela zawiera 4 kolumny i 37834 wierszy

(tych ostatnich zgodnie z liczbą w oryginalnej tabeli z pliku)

```r
dim(dane_TPM)
```

```
## [1] 37834     4
```