

Stability Score Algorithm

Luca Alessandri'

October 17, 2018

1 Introduction

Stability score will be detected for each cells. To evaluate this score the following algorithm is provided in rCASC.

Stability score Algorithm Be C the count matrix with NXM genes and cells.

$$C = \begin{bmatrix} c_{11} & c_{12} & \dots \\ \vdots & \ddots & \\ c_{N1} & & c_{NM} \end{bmatrix}$$

Be Cl_p the result vector of the clustering algorithm in the permutation p .

$\begin{cases} p = 0 \text{ then } C \text{ is complete} \\ p \neq 0 \text{ then } C \text{ is a subset of } C. \text{A specific percentage of cell is removed.} \end{cases}$

Be R_p the relation simmetric matrix MXM generated from Cl_p

$$R_p = \begin{bmatrix} Rp_{11} & Rp_{12} & \dots \\ \vdots & \ddots & \\ Rp_{M1} & & Rp_{MM} \end{bmatrix}$$

$$R_p(m_x, m_y) = \begin{cases} 1 & \text{if } Cl_p(m_x) = Cl_p(m_y) \\ 0 & \text{otherwise} \end{cases}$$

Be $R_p C = R_0 + R_p$, S the threshold of the score percentage, be $\text{dim}()$ the function that detect how many elements are in a vector, v_{p_m} each row of $R_p C$, k an interger number and be $\text{Length}()$ the function

$$\text{length}(v, k) = \sum_1^{\text{dim}(v)} \begin{cases} 1 & \text{if } R_p C_v = k \\ 0 & \text{otherwise} \end{cases}$$

$$\text{score}T_{p_m} = \frac{\text{length}(v_{p_m}, 2)}{\text{length}(v_{p_m}, 2) + \text{length}(v_{p_m}, 1)} \quad \forall m \wedge \forall p$$

$$\text{score}TT_{p_m} = \begin{cases} 1 & \text{if } \text{score}T_{p_m} \geq S \\ 0 & \text{otherwise} \end{cases}$$

Be P the total number of permutation

$$\text{Score}_m = \sum_{p=1}^P \frac{\text{score}TT_{p_m}}{P}$$

2 Example

$$\text{Be } Cl(p, m) = \begin{Bmatrix} 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 1 & 1 & 2 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 \end{Bmatrix}$$

$\forall p$ R_p is calculated,

$$R_0 = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

$$R_1 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix} \dots R_p$$

$$\forall p > 0 \text{ } R_p C \text{ is calculated } R_1 C = \begin{bmatrix} 2 & 0 & 1 & 2 & 0 \\ 0 & 2 & 1 & 0 & 2 \\ 1 & 1 & 2 & 1 & 1 \\ 2 & 0 & 1 & 2 & 0 \\ 0 & 2 & 1 & 0 & 2 \end{bmatrix} \dots R_p C$$

$\forall R_p C$ scoreT is evaluated with $S = 0.6$

$$scoreTT_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

This means that in permutation 1 the third cell is unstable "jumping" from cluster number 1 in P_0 to cluster number 2 in P_1 .

$Score_m$ is evaluated then for each p

$$Score_{mp} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix} \quad Score_m = \begin{bmatrix} 0.6 \\ 0.6 \\ 0.25 \\ 0.6 \\ 0.6 \end{bmatrix}$$

In this example number of cell in P_0 is the same as all the other permutation. What happen in reality is that P_p has a % of cell less then the P_0 . The cells that are not presents in P_p are then removed from P_0 and will be added with a 0 score in the specific position of the matrix ScoreTT.