

Pair-Score Heuristic

Let n be the number of questions in the survey. Consider one pair of candidates. For this pair, the pair score is given by the sum of the scores obtained in each of the questions.

$$pairScore = \sum_{i=1}^n score(q_i)$$

Where, $score(q_i)$ is the score obtained in question- i of the survey. Each question is given criteria which are mapped to the numeric value in the following way:

- SIMILARITY -> 1
- DISSIMILARITY -> 2

Score of the given question is calculated based on the following parameters:

- *expected*
- *observed*
- *pivotAnswer*
- *partnerAnswer*

expected is the criterion expected for the question. *observed* is the actual observation from the answers. *pivotAnswer* is the answer of the pivot candidate (the first candidate in the pair). *partnerAnswer* is the answer of the compared candidate (the second candidate in the pair). Then, the score obtained in the question is calculated as follows:

$$score(q_i) = weight(q_i) * gain(q_i)$$

Where, $weight(q_i)$ is the weightage given to the question by the course admin, and $gain(q_i)$ is the gain for that question, which is calculated as follows:

```
switch(questionType){
```

```
case MCQ_CHOOSE_ONE: gain = (1 - |expected - observed|)
```

```
case NUMERIC: if (expected == SIMILARITY and observed == SIMILARITY)
```

```
{
```

```
    gain = 1
```

```
}
```

```
else
```

```
{
```

```
    gain = (r * |pivotAnswer - partnerAnswer|)(-1)expected
```

```
}
```

```
case MCQ_CHOOSE_MANY:
```

```
if (expected == SIMILARITY) {
```

```
    gain = n(pivotAnswer ∩ partnerAnswer)
```

```
}
```

```
else {
```

```
    gain = n(pivotAnswer ∪ partnerAnswer) - n(pivotAnswer ∩ partnerAnswer)
```

```
}
```

```

case FREE_TEXT:
if (expected == SIMILARITY) {
    if (isAMatch == true) {
        gain = 1.0;
    } else {
        gain = 0.0;
    }
}
else if (expected == DISSIMILARITY) {
    if (isAMatch == false) {
        gain = 1.0;
    } else {
        gain = 0.0;
    }
}
}

```

Greedy Algorithm

The greedy grouping algorithm first chooses the pivot candidates from all the candidates either arbitrarily or based on the bounded criteria ('*atleast greater than*' or '*atleast lesser than*'). For each pivot, it gets the pair scores with remaining candidates. Once it receives the pair scores for all the pairs with the given pivot, it sorts the candidates in descending order of pair score with pivot. Then, the pivot and the first $(g - 1)$ students are made into one group, where g is the size of the group. This group's candidates are removed from the universal set of all the candidates. The remaining candidates undergo the same process with the next pivot in the pivots list.