Assessment 3 : peer marking

Students provide, in a concerted manner within their group, a vector of positive weights (in %)

$$(w_1, \ldots, w_g) \qquad \text{such that } \sum_{i=1}^{g} w_i = 100,$$

where g is the total number of students in their group. Note that giving a weight equal to $W = 100/g$ for all students will result in all of them having the same mark.

Next, they receive a group mark M (out of 100) for their group report by a lecturer.

The individual mark of the ith student is given by

$$m_i = w_i M/W, \qquad i = 1, \ldots, g,$$

provided all these marks are not larger than 100. Otherwise, we modify the weights and marks as follows. Let $w_{\max}$ be the largest weight among those given by the students (giving a mark > 100) and $W_{\sup} = 100W/M$ the weight that gives a maximal mark of 100. Let $\kappa = (W_{\sup} - W)/(w_{\max} - W)$ the ratio which permits to change the weight $w_{\max}$ to get a final mark of 100 for that student. We then change all weights w_i above W as follows:

$$w_i \longleftarrow W + \kappa(w_i - W).$$

It is straightforward to see that the largest new mark becomes $(W + \kappa(w_{\max} - W))M/W = 100$.

Similarly, we change all weights w_i below W as follows:

$$w_i \longleftarrow W - \kappa(W - w_i).$$

(The sum of the new weights stay equal to 100.) We then recompute the marks with the same formula as above, namely $m_i = w_i M/W, \qquad i = 1, \ldots, g.$

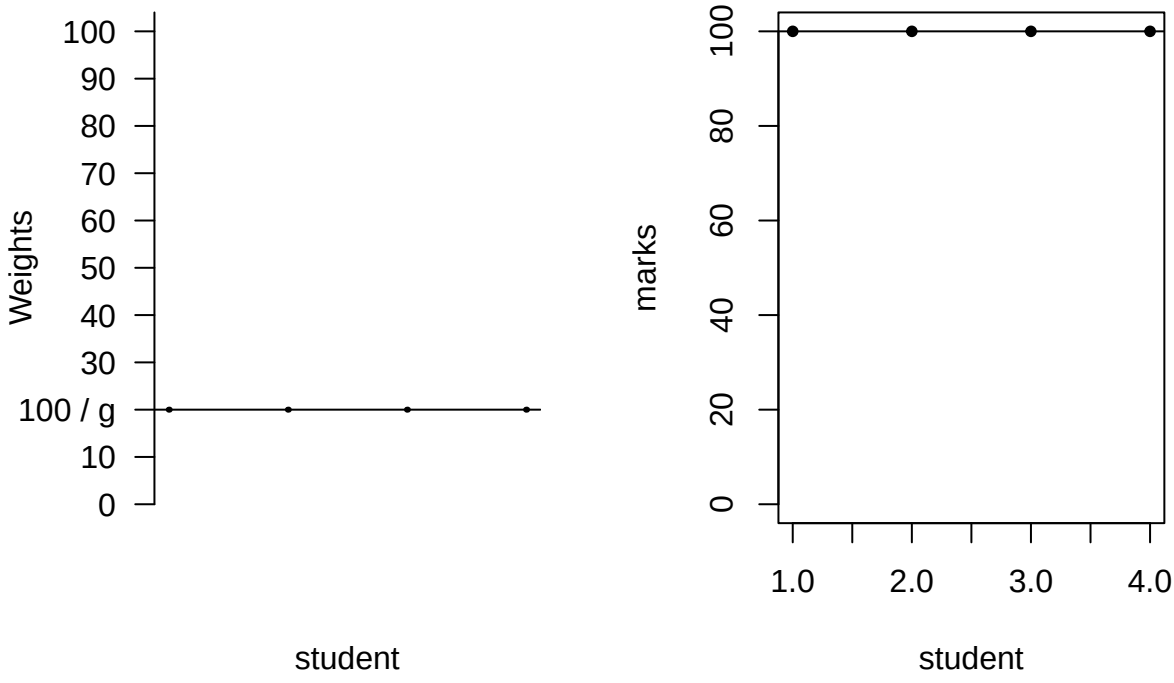
The corresponding R code is given below.

```r
compute.marks <- function(report.mark, weights) {
  # INPUT:
  #  - report.mark : mark given to the group report out of 100
  #  - weights:      vector of weights (in %) for all students in the group
  # OUTPUT:
  #  - optionnaly, vector of new weights
  #  - vector of (weighted) marks for all students in the group
  if (any(weights < 0 | weights > 100)) stop("All weights should be between 0 and 100.")
  if (abs(sum(weights) - 100) > 10 ^ (-3)) stop("The sum of weights (in %) should be 100.")
  g <- length(weights)
  W <- 100 / g
  marks <- report.mark * weights / W
  if (any(marks > 100)) {
    wmax <- weights[which.max(weights)]
    Wsup <- 100 * W / report.mark
    kappa <- (Wsup - W) / (wmax - W)
    excess.weights <- sum(weights[weights > W] - W)
```

```
    weights[weights > W] <- W + (weights[weights > W] - W) * kappa
    weights[weights < W] <- W - (W - weights[weights < W]) * kappa
    marks <- report.mark * weights / W
  }
  return(list(weights = weights, marks = marks))
}
```

Let's see an example.

```
M <- report.mark <- 100
weights <-  c(25, 25, 25, 25)
g <- length(weights)
W <- 100 / g
weights2 <- weights
weights2[weights >= W] <- weights2[weights >= W] - W
weights2[weights < W] <- -W + weights2[weights < W]
par(mfrow = c(1, 2))
plot(weights2, type = "h", lwd = 3, axes = FALSE,
     ylab = "Weights", xlab = "student", ylim = c(-20, 80))
abline(h = 0)
at <- c(-20, -10, 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100)
atplus <- at + 20
atplus[3] <- "100 / g"
axis(2, at = at , labels = atplus, las = 1)
marks <- g * M * weights / 100
plot(1:g, marks, ylim = c(0, max(marks)), xlab = "student", pch = 20) ; abline(h = 100)
```



```r
M <- report.mark <- 100
weights <- c(25, 25, 25, 25)
( res <- compute.marks(report.mark, weights) )
```

```
## $weights
## [1] 25 25 25 25
##
## $marks
## [1] 100 100 100 100
```

```r
sum(res$weights)
```

```
## [1] 100
```

```r
weights <- c(25, 25, 25, 25)
report.mark <- 100
( res <- compute.marks(report.mark, weights) )
```

```
## $weights
## [1] 25 25 25 25
##
## $marks
## [1] 100 100 100 100
```

```r
sum(res$weights)
```

```
## [1] 100
```