Triangular sums and transformations of matrices

Xavier R. Hoffmann

August 29, 2021

---

1 Shift vector

Helper function that shifts a vector up or down. Given a vector v of length N it returns the vector w applying the transformation:

$$w(i) = \begin{cases} v(i+k) & i+k \in [1, N] \\ 0 & \text{else} \end{cases} \tag{1}$$

With $k > 0$ the items are moved upwards and with $k < 0$ the positions move down. Null elements (index positions that don't exist) are replaced by 0. The vectors v and w have the same length.

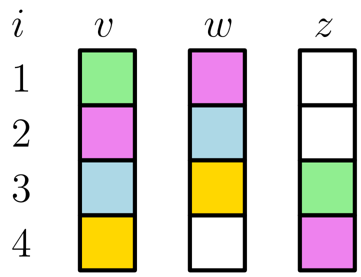

Figure 1: w is obtained shifting the original vector with $k = 1$. z is the result of applying a shift of $k = -2$. Blank cells represent zeros.

2 Shift matrix rows

Helper function that shifts the rows of a matrix up or down. Given a matrix A of shape $N \times M$ it returns the matrix B applying the transformation:

$$B(i, j) = \begin{cases} A(i+k, j) & i+k \in [1, N] \\ 0 & \text{else} \end{cases} \tag{2}$$

With $k > 0$ the rows are moved upwards and with $k < 0$ they move down. All elements of null rows (index rows that don't exist) are replaced by 0. The matrices A and B have the same dimensions.

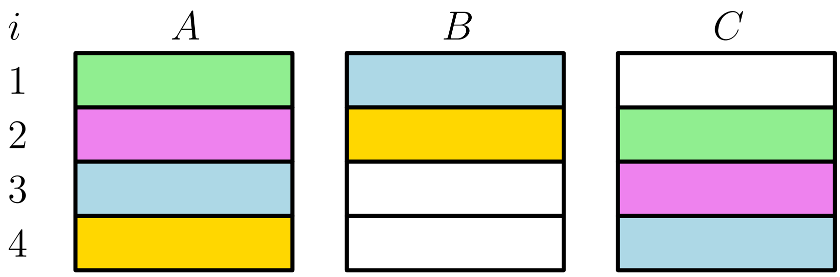

Figure 2: B is obtained shifting the original matrix with $k = 2$. C is the result of applying a shift of $k = -1$. Blank cells represent rows where all positions are zeros.

3 Reverse cumsum

Computes the reverse cumulative sum of a vector, possibly shifted some positions up or down. Given a vector v of length N it returns the vector w applying the transformation:

$$w(i) = \begin{cases} \sum_{j=i+k}^{N} v(j) & i + k \in [1, N] \\ 0 & \text{else} \end{cases} \tag{3}$$

With $k > 0$ the items of the resulting vector are moved upwards and with $k < 0$ the positions move down. Whenever a sum includes an index that doesn't exist, the whole sum is replaced by 0. The vectors v and w have the same length.

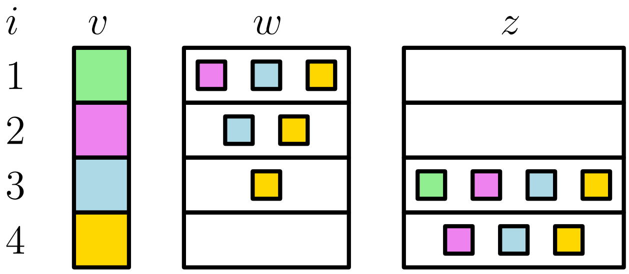

Figure 3: w is obtained with a shift of $k = 1$. z is the result of applying a shift of $k = -2$. Blank cells represent zeros.

4 Triangular sum rows

Computes the sum along rows of upper-triangular terms, optionally shifting the resulting vector up or down. Given a matrix A of shape $N \times M$ it returns the vector w applying the transformation:

$$w(i) = \begin{cases} \sum_{j=i+k}^{N} A(i + k, j) & i + k \in [1, N] \\ 0 & \text{else} \end{cases} \tag{4}$$

With $k > 0$ the items of the resulting vector are moved upwards and with $k < 0$ the positions move down. Whenever a sum includes an index that doesn't exist, the whole sum is replaced by 0. The vector w has length M.

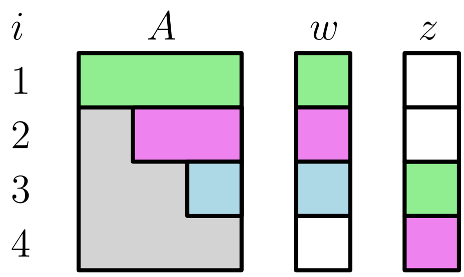

Figure 4: The lower-diagonal elements of A are not included in the sum. w is obtained with a shift of $k = 0$. z is the result of applying a shift of $k = -2$. Blank cells represent zeros.

5 Triangular dot

Computes a matrix-vector product but only of upper-triangular terms, optionally shifting the resulting vector up or down. Given a matrix A of shape $N \times M$ and a vector v of length M it returns the vector w applying the transformation:

$$w(i) = \begin{cases} \sum_{j=i+k}^{N} A(i+k, j)v(j) & i+k \in [1, N] \\ 0 & \text{else} \end{cases} \tag{5}$$

With $k > 0$ the items of the resulting vector are moved upwards and with $k < 0$ the positions move down. Whenever a sum includes an index that doesn't exist, the whole sum is replaced by 0 The vector w has length M.

Note that this operation is almost identical to `triangular sum rows` but previously multiplying by the elements of the vector v.

6 Triangular sum columns

Computes the sum along columns of upper-triangular terms, optionally adding a positive or negative diagonal offset to include more or less terms. Given a matrix A of shape $N \times M$ it returns the vector w applying the transformation:

$$w(i) = \sum_{j=1}^{i+k} A(j, i) \tag{6}$$

With $k > 0$ below-diagonal bands are included and wth $k < 0$ above-diagonal bands are excluded. Whenever the sum includes a row index that doesn't exist, the sum adds 0. The vector w has length M.

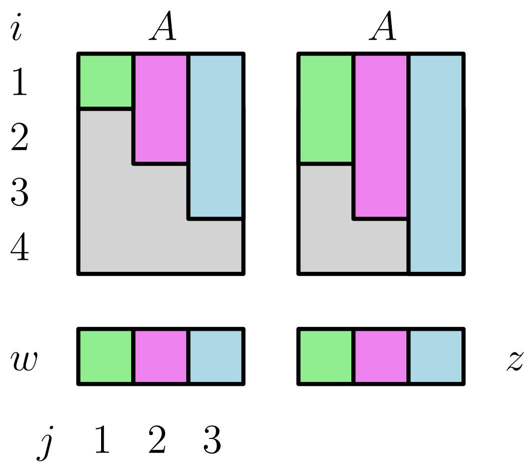

Figure 5: The lower-diagonal elements of A are not included in the sum. w is obtained with $k = 0$. z is the result of applying $k = 1$.

7 Triangular sum chunks

Computes the sum of the submatrix defined by the diagonal element and the upper right corner, optionally moving the lower right corner of the submatrix (by default the diagonal element). Given a matrix A of shape $N \times M$ it returns the vector w applying the transformation:

$$w(i) = \begin{cases} \sum_{m=1}^{i+k} \sum_{j=i+\ell}^{M} A(m, j) & i + \ell \in [1, M] \\ 0 & \text{else} \end{cases} \tag{7}$$

With k we move the lower right corner of the submatrix up ($k < 0$) or down ($k > 0$). With ℓ we move the lower right corner of the submatrix left ($\ell < 0$) or right ($\ell > 0$). Whenever the sum includes a column index that doesn't exist, the whole sum is replaced by 0. Whenever the sum includes a row index that doesn't exist, the sum adds 0. The vector w has length M.

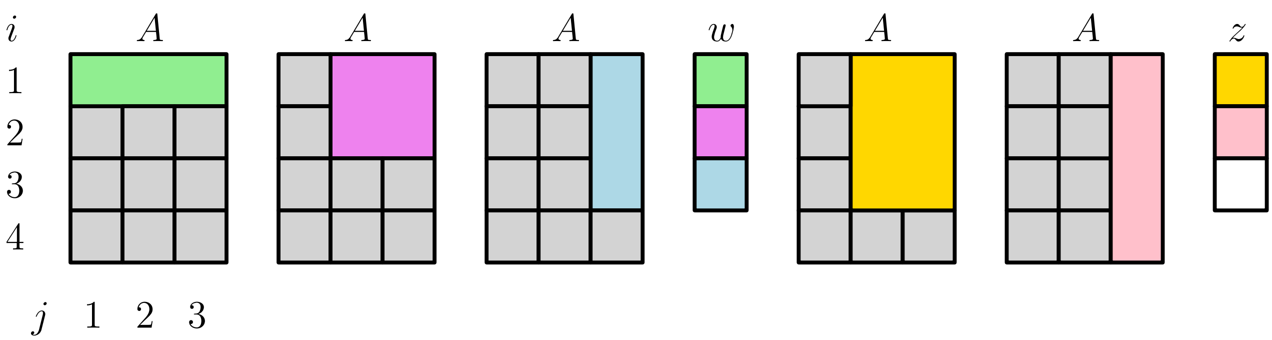

Figure 6: w is obtained with $k = 0$ and $\ell = 0$. z is obtained with $k = 2$ and $\ell = 1$. Blank cells represent zeros.