Data manipulation and filtering with data.table

By using data.table to structure our cytometry data, we can manipulate and filter the datasets extremely quickly. Subsetting rows or columns of a data.table can be achieved using the following structure:

x	[	ROWS	,	COLUMNS	,	BY (optional)	]

Filtering approaches:

- Rows can be selected by the row number or values in a specified column
- Columns can be specified as the column numbers or the column names
- BY is used when grouping data together, but this is not demonstrated on this page

Here is a quick guide to some common operations, but a more comprehensive guide can be found on the data.table home page.

Task	Command (with minimum variable inputs)
Access a 'demo' dataset stored within Spectre, and assign it to 'cell.dat'.	`cell.dat <- Spectre::demo.clustered`

Subsetting data using **column** numbers	`## Select the first column of the data.table`
	`sub <- cell.dat[ ,1]`
	`sub`
	
	`## Select columns 1, 3, and 5`
	`sub <- cell.dat[ ,c(1,3,5)]`
	`sub`
	
	`## Select columns 1 to 10, and 15`
	`sub <- cell.dat[ ,c(1:10,15)]`
	`sub`

Subsetting data using **column** names	`## View a list of column names`
	`as.matrix(names(cell.dat))`
	
	`     [,1]`
	` [1,] "FileName"`
	` [2,] "NK11"`
	` [3,] "CD3"`
	` [4,] "CD45"`
	` [5,] "Ly6G"`
	` [6,] "CD11b"`
	` [7,] "B220"`
	` [8,] "CD8a"`
	` [9,] "Ly6C"`
	`[10,] "CD4"`
	`[11,] "NK11_asinh"`
	`[12,] "CD3_asinh"`
	`[13,] "CD45_asinh"`

Task | **Command (with minimum variable inputs)**

```
        [14,] "Ly6G_asinh"
        [15,] "CD11b_asinh"
        [16,] "B220_asinh"
        [17,] "CD8a_asinh"
        [18,] "Ly6C_asinh"
        [19,] "CD4_asinh"
        [20,] "Sample"
        [21,] "Group"
        [22,] "Batch"
        [23,] "FlowSOM_cluster"
        [24,] "FlowSOM_metacluster"
        [25,] "Population"
        [26,] "UMAP_X"
        [27,] "UMAP_Y"
```

```
## Select columns names 11 to 19
cols <- names(cell.dat)[c(11:19)]
cols
```

```
[1] "NK11_asinh"  "CD3_asinh"   "CD45_asinh"  "Ly6G_asinh"  "CD11b_asinh"
"B220_asinh"  "CD8a_asinh"  "Ly6C_asinh"  "CD4_asinh"
```

To select columns based on **column name**, either '..' needs to go before the vector of column names, or ', with = FALSE' needs to be added to the end of the data.table filtering arguments.

```
## OPTION 1 - Select columns using '..'
sub <- cell.dat[ ,..cols]
sub

## OPTION 2 - Select columns using 'with = FALSE'
sub <- cell.dat[ ,cols, with = FALSE]
sub
```

Subsetting data using **row** numbers

```
## Select the first row of the data.table
sub <- cell.dat[1, ]
sub

## Select rows 1, 3, and 5
sub <- cell.dat[c(1,3,5), ]
sub

## Select rows 1 to 10, and 15
sub <- cell.dat[c(1:10,5), ]
sub
```

Subsetting data using **row** values in a selected column

Subsetting rows using data.table can be performed with the following structure:

cell.dat	[	CONDITIONS	,		]

The **conditions** is essentially a filtering operation to determine which rows have a value in a specific column that is equal to, higher, or lower than a specified value. This can be performed using something like this:

<table>
<tr><th>Task</th><th>Command (with minimum variable inputs)</th></tr>
<tr><td></td><td>

```
## Creates a TRUE/FALSE results for which rows contain "Ly6C_asinh" values above 2
cell.dat[["Ly6C_asinh"]] > 2
```

We can use this conditional TRUE/FALSE results to select which rows to include from our data.table.

```
## Select rows (cells) where 'Ly6C_asinh' is above 2
sub <- cell.dat[cell.dat[["Ly6C_asinh"]] > 2, ]
sub

## Select rows (cells) where 'CD45_asinh' is below 3
sub <- cell.dat[cell.dat[["CD45_asinh"]] < 3, ]
sub

## Select rows (cells) where 'FlowSOM_metacluster' is above 2
sub <- cell.dat[cell.dat[["FlowSOM_metacluster"]] == 5,]
sub

## Select rows (cells) where 'Population' is 'Infil Macrophages'
sub <- cell.dat[cell.dat[["Population"]] == 'Infil Macrophages',]
sub
```

</td></tr>
<tr><td>Subsetting data using multiple <b>row</b> values in multiple columns</td><td>

Multiple 'or' arguments an be added by using '|', for example: **cell.dat[A | B | C,].** This is an 'OR' operation, so all cells that are Ly6C_asinh > 2, in addition to all cells that are 'Infil Macrophages' will be included, rather than only cells satisfying both conditions (which would be an 'AND' operation).

```
## Select rows (cells) where
    # 'Ly6C_asinh' is above 2 OR
    # 'Population' is 'Infil Macrophages'

sub <- cell.dat[cell.dat[["Ly6C_asinh"]] > 2 |
                cell.dat[["Population"]] == 'Infil Macrophages'
                ,]
sub
```

</td></tr>
</table>

Extracting text before or after a symbol

Let's say we have a character vector, with a symbol dividing it into two:

```
x <- c("everything before|everything after")
```

We can use the following to extract everything **before** '|'.

```
before <- sub("\\|.*", "", x)
before
```

```
"everything before"
```

And we can use the following to extract everything **after** '|'.

```
after <- sub(".*\\|", "", x)
after
```

```
"everything after"
```