MANIPULATING DATA.TABLES

Olivia Johnson

Adelaide Code Club

25 August 2022

OUTLINE FOR TODAY

- Recap of last week
- Adding columns
- Converting columns
- Extracting summaries
- Time to practise!

RECAP

- data.table considered the fasted R package for data manipulation
- Create/read in data
  - `fread(path to data or webpage)`
  - `setDT(data frames and lists)`
  - `as.data.table(for other structures)`
  - `data.table(values)`

RECAP

- Basic syntax

```
DT[i, j, by]
```

- Use square brackets
- **i** subsets rows
- **j** subsets columns
- Group by **by**

RECAP

- Can apply functions to the data
  - DT[, **sum**(column)]
  - DT[, **mean**(col2),by="col1"]
  - **.N** - returns the number of rows, goes in j

```
> mtcars[, .N, by="cyl"]
   cyl  N
1:   6  7
2:   4 11
3:   8 14
```

ADDING COLUMNS (:=)

- Data.table allows you to compute values and add them to the table in a new column
- Can use other columns to compute values

```
DT[, col_name:=function or equation]
```

- ie. `iris[, sepal_ratio := Sepal.Width/Sepal.Length]`
- `iris[, species_ratio := Sepal.Width/Sepal.Length, by="species"]`
  - this will assign the same value to all observations in each grouping

ADDING MULTIPLE COLUMNS

- Can compute multiple columns at once

```
DT[, ':=' (col_1=value, col_2=value)]
```

- ie. `iris[, ':='`

```
(Sepal_ratio=Sepal.Width/Sepal.Length,
Petal_ratio=Petal.Width/Petal.Length)]
```

CONVERTING COLUMNS

- Can also use := to covert the data type of column values

```
DT[, b := as.integer(b)]
```

- And delete columns

```
DT[, b := NULL ]
```

EXTRACTING SUMMARIES

- You can also calculate values from a data.table and output them in another data.table

```
dt[, .(mean=mean(Petal.Length),
median=median(Petal.Length),
min=min(Petal.Length), max=max(Petal.Length)),
by='Species']
```

	Species	mean	median	min	max
1:	setosa	1.462	1.50	1.0	1.9
2:	versicolor	4.260	4.35	3.0	5.1
3:	virginica	5.552	5.55	4.5	6.9

EXERCISE TIME!