Lightning Talk

Introduction to

data.table

(in 6 mins)

Matthew Dowle

What is data.table?

- Think `data.frame`, inherits from it
- `data.table()` and `?data.table`

Goals:

- Reduce programming time

  fewer function calls, less variable name repetition

- Reduce compute time

  fast aggregation, update by reference

- In-memory only, 64bit and 8GB+ routine
- Useful in finance but wider use in mind, too

  e.g. genomics

Reducing programming time

```
trades[
   filledShares < orderedShares,
   sum( (orderedShares-filledShares)
        * orderPrice / fx ),
   by = "date,region,algo"
]
```

```
R   :   i          j              by
SQL :   WHERE      SELECT         GROUP BY
```

Reducing compute time

e.g. 10 million rows x 3 columns x,y,v 230MB

```
DF[DF$x=="R" & DF$y==123,]   # 8     s
DT[.("R",123)]               # 0.008s

tapply(DF$v,DF$x,sum)        # 22    s
DT[,sum(v),by=x]             #  0.83s
```

See above in timings vignette (copy and paste)

Fast and friendly file reading

e.g. 50MB .csv, 1 million rows x 6 columns

```
read.csv("test.csv")              # 30-60s

read.table("test.csv", colClasses=,
nrows=, etc...)                   #    10s

fread("test.csv")                 #     3s
```

e.g. 20GB .csv, 200m rows x 16 columns

```
fread("big.csv")                  #   450s
```

Update by reference using :=

Add new column ”sectorMCAP” by group :

```
DT[,sectorMCAP:=sum(MCAP),by=Sector]
```

Delete a column (0.00s even on 20GB table) :

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

Be explicit to really copy entire 20GB :

```
DT2 = copy(DT)
```

Why R?

Lazy evaluation enables :

- `DT[ FilledShares < OrderedShares ]`
- `j` inspected and optimized before evaluation

`is.data.frame(DT)==TRUE`. Pass `DT` to any package taking `DF` only. It works.

CRAN (cross platform release, quality control)

Thousands of statistical packages to use in `j`

Thank you!

http://datatable.r-forge.r-project.org/