Simple JSON RPC with OpenCPU

The `jsonlite` package is used by `OpenCPU` to convert between `JSON` data and `R` objects. Thereby clients can retrieve `R` objects, or remotely call `R` functions using `JSON` where the function arguments as well as function return value are `JSON` objects. For example to download the `Boston` data from the `MASS` package:

Command in R	Example URL on OpenCPU
`toJSON(Boston, digits=4)`	`https://cran.ocpu.io/MASS/data/Boston/json?digits=4`
`toJSON(Boston, dataframe="col")`	`https://cran.ocpu.io/MASS/data/Boston/json?dataframe=col`
`toJSON(Boston, pretty=FALSE)`	`https://cran.ocpu.io/MASS/data/Boston/json?pretty=false`

To calculate the variance of some the numbers `1:9` in the command line using using `curl`:

```
curl https://cran.ocpu.io/stats/R/var/json -d "x=[1,2,3,4,5,6,7,8,9]"
```

Or equivalently post the entire body in `JSON` format:

```
curl https://cran.ocpu.io/stats/R/var/json -H "Content-Type: application/json" \
-d "{\"x\":[1,2,3,4,5,6,7,8,9]}"
```

Below an example where we call the `melt` function from the `reshape2` package using some example rows from the `airquality` data. Here both input and output consist of a data frame.

```
curl https://cran.ocpu.io/reshape2/R/melt/json -d 'id=["Month", "Day"]&data=[
  { "Ozone" : 41, "Solar.R" : 190, "Wind" : 7.4, "Temp" : 67, "Month" : 5, "Day" : 1 },
  { "Ozone" : 36, "Solar.R" : 118, "Wind" : 8, "Temp" : 72, "Month" : 5, "Day" : 2 } ]'
```

Or equivalently:

```
curl https://cran.ocpu.io/reshape2/R/melt/json -H "Content-Type: application/json" \
  -d '{"id" : ["Month", "Day"], "data" : [
    { "Ozone" : 41, "Solar.R" : 190, "Wind" : 7.4, "Temp" : 67, "Month" : 5, "Day" : 1 },
    { "Ozone" : 36, "Solar.R" : 118, "Wind" : 8, "Temp" : 72, "Month" : 5, "Day" : 2 }
  ] }'
```

This request basically executes the following `R` code:

```
mydata <- airquality[1:2,]
y <- reshape2::melt(data = mydata, id = c("Month", "Day"))
toJSON(y)
```