

An Interactive Example

Gene Dan

github.com/genedan
genedan.com

February 3, 2014

CPUs

This is an interactive example where I'll be guiding the audience on how to export visual output from an IDE (like RStudio) to a pdf. We will be using the **cpus** dataset from the **MASS** package. This dataset includes performance metrics for various cpus.

As in the main presentation, we can simultaneously include code and its output within the same chunk:

```
library(MASS)
head(cpus[, c(1, 5, 8)])
```

| ## | | name | cach | perf |
|------|---------|---------|------|------|
| ## 1 | ADVISOR | 32/60 | 256 | 198 |
| ## 2 | AMDAHL | 470V/7 | 32 | 269 |
| ## 3 | AMDAHL | 470/7A | 32 | 220 |
| ## 4 | AMDAHL | 470V/7B | 32 | 172 |
| ## 5 | AMDAHL | 470V/7C | 32 | 132 |
| ## 6 | AMDAHL | 470V/8 | 64 | 318 |

Suppressing Output

However, sometimes when writing a technical document, you want to suppress the output while presenting the code. This gives you a little bit of breathing space to explain yourself before showing the output. We can achieve this by setting the **eval** option to **FALSE**:

```
library(MASS)
head(cpus[, c(1, 5, 8)])
```

You can see that we have the same two input lines as in the previous example, but with no output this time.

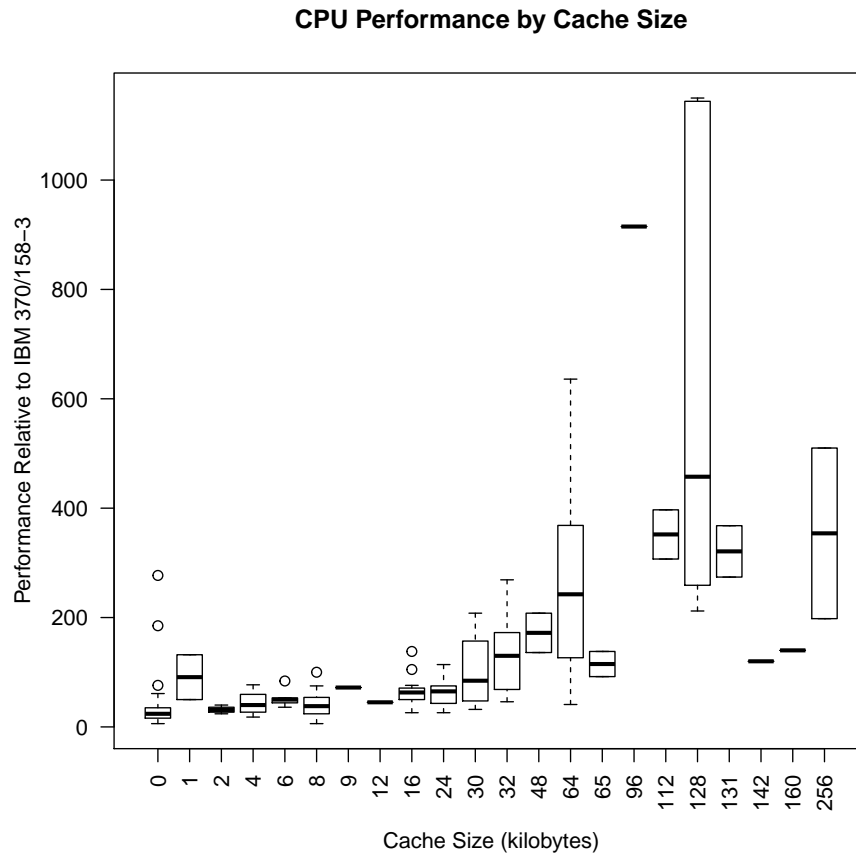
Likewise, sometimes we want to show the output without expressing the code that was used to generate it. We can do so by setting the **echo** option to **FALSE**:

```
##           name cach perf
## 1  ADVISOR 32/60  256  198
## 2  AMDAHL 470V/7   32  269
## 3  AMDAHL 470/7A   32  220
## 4  AMDAHL 470V/7B   32  172
## 5  AMDAHL 470V/7C   32  132
## 6  AMDAHL 470V/8   64  318
```

Plotting a Figure

We can plot the relative performance of cpus v.s. cache size using R's plotting capabilities. Set the option **fig** to **TRUE** in order to do this:

```
plot(factor(cpus$cach), cpus$perf, las=2,
     main="CPU Performance by Cache Size",
     xlab="Cache Size (kilobytes)",
     ylab="Performance Relative to IBM 370/158-3")
```

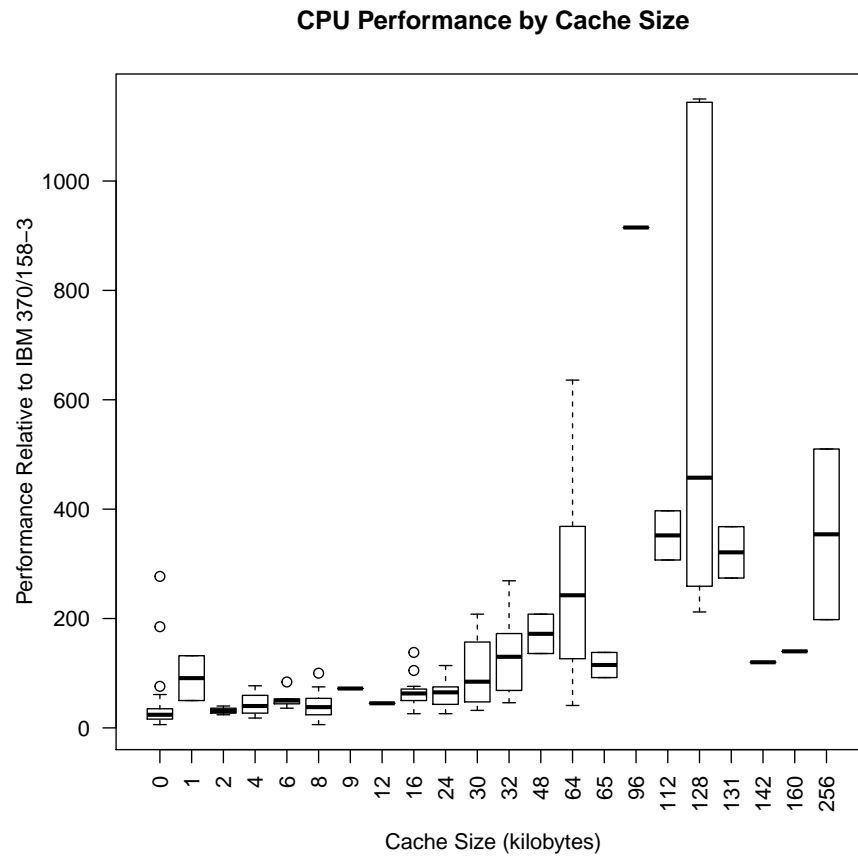


As before, we can suppress the output:

```
plot(factor(cpus$cach), cpus$perf, las=2,  
      main="CPU Performance by Cache Size",  
      xlab="Cache Size (kilobytes)",  
      ylab="Performance Relative to IBM 370/158-3")
```

Suppressing Code for Publication

And likewise we can suppress the code used to generate the plot. Sometimes, this is necessary when you are presenting to a non-technical audience (such as clients):



Importing a Table of Values

Although we can show R output directly on the page, sometimes the default monospace font isn't appropriate for our audience (especially if we are dealing with non-technical people):

```
cpus[1:20, c(1, 2, 3, 4, 5, 8)]
```

| ## | | name | syct | mmin | mmax | cach | perf |
|-------|-----------|----------|------|-------|-------|------|------|
| ## 1 | ADVISOR | 32/60 | 125 | 256 | 6000 | 256 | 198 |
| ## 2 | AMDAHL | 470V/7 | 29 | 8000 | 32000 | 32 | 269 |
| ## 3 | AMDAHL | 470/7A | 29 | 8000 | 32000 | 32 | 220 |
| ## 4 | AMDAHL | 470V/7B | 29 | 8000 | 32000 | 32 | 172 |
| ## 5 | AMDAHL | 470V/7C | 29 | 8000 | 16000 | 32 | 132 |
| ## 6 | AMDAHL | 470V/8 | 26 | 8000 | 32000 | 64 | 318 |
| ## 7 | AMDAHL | 580-5840 | 23 | 16000 | 32000 | 64 | 367 |
| ## 8 | AMDAHL | 580-5850 | 23 | 16000 | 32000 | 64 | 489 |
| ## 9 | AMDAHL | 580-5860 | 23 | 16000 | 64000 | 64 | 636 |
| ## 10 | AMDAHL | 580 5880 | 23 | 32000 | 64000 | 128 | 1144 |
| ## 11 | APOLLO | DN320 | 400 | 1000 | 3000 | 0 | 38 |
| ## 12 | APOLLO | DN420 | 400 | 512 | 3500 | 4 | 40 |
| ## 13 | BASF | 7/65 | 60 | 2000 | 8000 | 65 | 92 |
| ## 14 | BASF | 7/68 | 50 | 4000 | 16000 | 65 | 138 |
| ## 15 | BTI | 5000 | 350 | 64 | 64 | 0 | 10 |
| ## 16 | BTI | 8000 | 200 | 512 | 16000 | 0 | 35 |
| ## 17 | BURROUGHS | B1955 | 167 | 524 | 2000 | 8 | 19 |
| ## 18 | BURROUGHS | B2900 | 143 | 512 | 5000 | 0 | 28 |
| ## 19 | BURROUGHS | B2925 | 143 | 1000 | 2000 | 0 | 31 |
| ## 20 | BURROUGHS | B4955 | 110 | 5000 | 5000 | 142 | 120 |

We can remedy this situation by using the `xtable` package that converts R output into a \LaTeX table. We can suppress the code to produce a well-formatted table for professional publication:

| name | syst | mmin | mmax | cach | perf |
|-----------------|------|-------|-------|------|------|
| ADVISOR 32/60 | 125 | 256 | 6000 | 256 | 198 |
| AMDAHL 470V/7 | 29 | 8000 | 32000 | 32 | 269 |
| AMDAHL 470/7A | 29 | 8000 | 32000 | 32 | 220 |
| AMDAHL 470V/7B | 29 | 8000 | 32000 | 32 | 172 |
| AMDAHL 470V/7C | 29 | 8000 | 16000 | 32 | 132 |
| AMDAHL 470V/8 | 26 | 8000 | 32000 | 64 | 318 |
| AMDAHL 580-5840 | 23 | 16000 | 32000 | 64 | 367 |
| AMDAHL 580-5850 | 23 | 16000 | 32000 | 64 | 489 |
| AMDAHL 580-5860 | 23 | 16000 | 64000 | 64 | 636 |
| AMDAHL 580 5880 | 23 | 32000 | 64000 | 128 | 1144 |
| APOLLO DN320 | 400 | 1000 | 3000 | 0 | 38 |
| APOLLO DN420 | 400 | 512 | 3500 | 4 | 40 |
| BASF 7/65 | 60 | 2000 | 8000 | 65 | 92 |
| BASF 7/68 | 50 | 4000 | 16000 | 65 | 138 |
| BTI 5000 | 350 | 64 | 64 | 0 | 10 |
| BTI 8000 | 200 | 512 | 16000 | 0 | 35 |
| BURROUGHS B1955 | 167 | 524 | 2000 | 8 | 19 |
| BURROUGHS B2900 | 143 | 512 | 5000 | 0 | 28 |
| BURROUGHS B2925 | 143 | 1000 | 2000 | 0 | 31 |
| BURROUGHS B4955 | 110 | 5000 | 5000 | 142 | 120 |

Table 1: CPU Performance Relative to IBM 370/158-3