Mathematica Vs R

author: Anton Antonov date: 2015-10-21 width: 1920 height: 1200 transition: none transition-speed: fast

Talk plan

How to make a comparison breakdown

- . . . that might take too long . . .

The quick list of great R features

How to quickly start work with R

. . . If you are an experienced *Mathematica* user.

1. Data structures
2. Libraries to use
3. How to approach problems
4. How to structure files

Examples

- In three groups:
  - on par
  - R is worse
  - R is better
- This list is probably too long. . .

How to make the comparison?

Language design features

- Is it functional, lazy, based on objects?
- What data structures?
- Is it easy to learn?

Documentation writing capabilities

- Does it facilitate writing technical papers?
- Presentations?
- Automatic reports?

Projects on machine learning and data mining (since we talk about R)

- How the codes compare?
- What observations we would abstract into rules?
- What the best practices?

Performance

- How it can perform faster?
- How easy/well is to parallelize the computations?

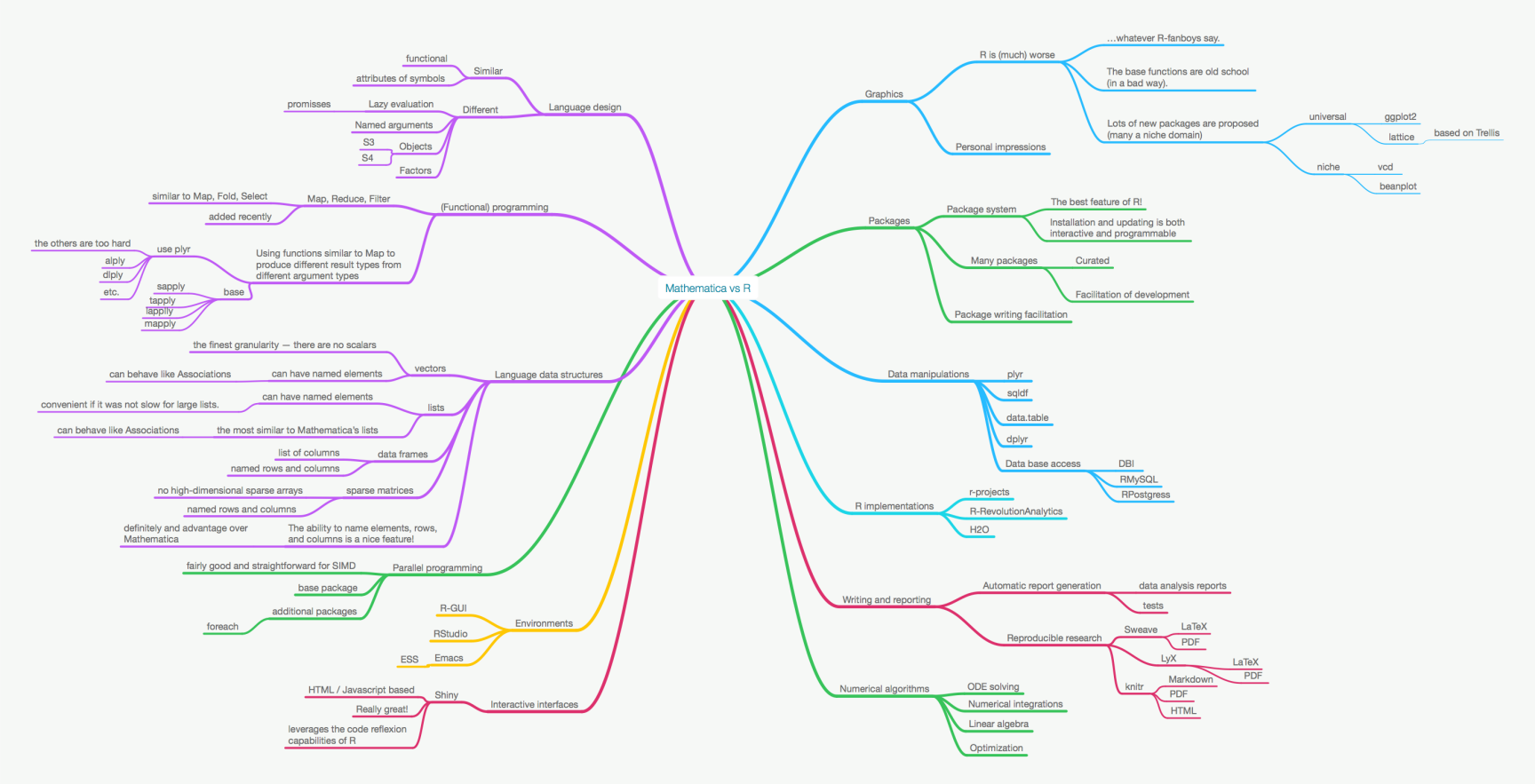
Extensibility

More detailed breakdown: Mind-map slide .

Mind-map

id: MindMap

Here is a more detailed view breakdown for a comparison:



Mind map link

The quick list

Programming and working with R has four great features:

1. Great IDE's support
    - RStudio

    - ESS for Emacs.
    - r4intellij of IntelliJ Idea.
2. Great package system
    - The package system itself, and
    - the amount of available packages (at CRAN),
3. Documentation integration with LaTeX, Markdown, and HTML
    - Example of HTML report automatically generated using knitr
4. Interactive interfaces building and deployment (Shiny)
    - ODE with seasonal term solving intercative interface

R as a language

What is R?

- The R Project for Statistical Computing
- What is R?
- Disclaimer on my bias
- I have used Mathematica for 22 years, and R for 2 years.
- But look at the R packages I have written.

Language origins

- LISP / Scheme refitted to look and feel like S.
- Too many designers and too many of them are statisticians.
- See the anti-pattern (Design by a comittee).

R as a language 2

Books (having *Mathematica* programmers in mind)

1. The R Inferno by Patrick Burns
2. Advanced R by Hadley Wickham

Articles

3. Ihaka, Ross (2010). R: Lessons Learned, Directions for the Future (PDF). Joint Statistical Meetings 2010, Statistical Computing Section.
4. Ihaka, Ross; Temple Lang, Duncan (25 August 2008). Back to the Future: Lisp as a Base for a Statistical Computing System (PDF). Compstat 2008.
5. Morandat, Frances; Hill, Brandon (2012). "Evaluating the design of the R language: objects and functions for data analysis"

Blogs

6. Eric Blair, "R is slow" (2006)

Data structures

data frame

- A list of lists that make full array

data table

- Kind of like Dataset

list

- Very similar to Mathematica's associations

vector

- There are no scalars

factor

- For handling categorical variables. I avoid using it.

sparse matrix

- Really nice!

formula

- Not a data structure but object to know and use.

environment

- Similar to Context‘ but also Association. It is used to handle scoping (variable values, bidings, etc.)

R tips for a Mathematica programmer

Data structures

- Learn data frames manipulations.
- Use named elements in vectors and lists.
- Use named rows and columns for matrices and data frames.

Functions over data

- Study and use the packages ‘plyr’ and ‘dplyr’.
- The articles of Hadley Wickham are insightful for R’s culture and data manipulation in general.

Map, Reduce, Filter

- Those correspond to *Mathematica* Map, Fold, Select.

In complicated data massaging use sqldf

- The package sqldf lets you treat your data frames as SQL tables.
- Reasonbly fast, great for project and task transition.

R tips for a Mathematica programmer 2

The Extract operator

- Similar to Part
- Used in a fashion similar to Pick

Intros to R for programmers

- See look for books/blogs like this: R language for programmers

Object-oriented programming

S3

- Functional polymorphism.
- Rudimentary, can be trivially implemented with Mathematica's pattern matching of signatures.

S4

- Very nice, based on CLOS.

Graphics

R has three distinct graphics systems: 1. the "traditional" graphics system, 2. the grid graphics system (the lattice package based on the Trellis graphics), and 3. ggplot2.

Overview presentation from the author of "R Graphics"

[Murrell "R Graphics"] (https://www.stat.auckland.ac.nz/~paul/Talks/CSIRO2011/rgraphics.pdf)

- Comments
- Interesting and insightful (for an R user), but the plots are not impressive compared to *Mathematica*.

Links

- CRAN Task View: Graphic Displays & Dynamic Graphics & Graphic Devices & Visualization
- Lattice project
- ggplot2

(Demo Plots3D.R for 3D graphics.)

Parallel programming

package 'parallel'

```
if ( TRUE ) {
  cat("\n\tParallel NN's generation with mcparallel:")

  cat("\n\tTotal number of cores:", detectCores() )
  cat("\n\tCores to be used:", mcCores )

  # We assume the rowIDs to be the item ID's we want to compute NN's for.
  cat("\n\t\tComputing overall similarity...")
  startTime <- Sys.time()
```

```
rowIDToIndex <- 1:length(rowIDs)
names(rowIDToIndex) <- rowIDs

rowIDToIndexList <- Slice(
  rowIDToIndex,
  ceiling( length(rowIDToIndex) / mcCores ) )

pls <-
  lapply( 1:length(rowIDToIndexList), function(i) {
    fname <- paste("./overallNNs", as.character(i), ".tsv", sep = "_")
    mcparallel( GenerateAndWriteNNs( rowIDToIndexList[[i]], rowIDs, fname ), as.character(i) )
  })
print(mccollect(pls, wait=TRUE))

endTime <- Sys.time()
cat("\n\t\tParallel SIMD computing time:", difftime( endTime, startTime, units="secs" ) )
```

Parallel programming 2

package 'foreach'

```
guessesPar <-
      foreach( parMovieInds = slicedMovieIndsList, .combine = rbind ) %dopar% {
        res <- llply( parMovieInds, function( i ) {
          mvec <- movieVecs[ i,,drop=FALSE]
          recs <- SMRRecommendationsByProfileVector( gtSMR, mvec, 30 )
          prof <- SMRProfileDF( gtSMR, itemHistory = recs[,c(1,3)] )
          prof[ gtSMR$TagTypeRanges[ nrow(gtSMR$TagTypeRanges), 1] <= prof$Index, ]
        }, .progress="time", .parallel = TRUE )
        names(res) <- movieIDs
      }
```

Efficient implementations

Lots of people (companies) are inclined to take a subset of the R language and make efficient implementations for it.

For example H_2O by 0xdata: http://h2o.ai/product/.

```
##library(h2o)
##localH2O = h2o.init()
```

Let us run a demo to see H_2O at work.

```
##demo(h2o.glm)
```

From

Get started with H2O in 3 easy steps

1. Download H2O. This is a zip file that contains everything you need to get started.
2. From your terminal, run:

```
cd ~/Downloads
unzip h2o-3.2.0.9.zip
cd h2o-3.2.0.9
java -jar h2o.jar
```

3. Point your browser to http://localhost:54321.

Examples for "on par"

ODE simulations interactive interface

Movies recommendations engines

```
- *Mathematica* interface
- [R/Shiny interface](http://127.0.0.1:7396)
```

Geo-spatial statistics and recommendations

Functional parsers

Well slideshows, reports, etc.

Examples where R is worse

Tiger shark migration analysis

- I wrote this blog post three years ago.
- I was not able to redo the experiments in R because it was very hard to get the data.
  - (. . . and I gave up).

2D and 3D quantile regression

- See this blog post.
- This would be hard to implement without the computational regions technology.

Finding local extrema in noisy data

- See this blog post.
- The approach would be hard to implement since R does not have differentiation and equation solving.
  - Yes, I looked for "automatic differentiation" in R.
  - And yes, you can solve systems of linear equations.

Examples for future projects where R would be on par with Mathematica

GitHubData plots

Time series conversational engine

Most characterizing sentence extraction

Music

Other examples at GitHub

- I plan to post example comparison projects at GitHub: MathematicaVsR.

Examples where R "has a package for it"

Mel Frequency Cepstral Coefficients

```
- tuneR
```

Frequent sets and associations rules mining

```
- arules
```

Many others...

On being bilingual

- In order to know your mountain you should climb the one next to it.
- Here are *Mathematica* packages I wrote that follow R existing functionalities:
- MosaicPlot.m
- QuantileRegression.m

- RecordsSummary in MathematicaForPredictionUtilities.m
- RSparseMatrix.m

Conclusions

- If you know *Mathematica* well I am not sure it is worth investing learning R if you do not have some project that would motivate you.
- Having R in LaTeX or org-mode is really great!
- I mostly side with the venom toward R seen in the book "The R Inferno" by Burns.
- More constructive to learning R is the book "Advanced R" by Hadley Wickham.
- And in general it is a good idea to read H. Wickham's writings.
- R is slow, but keep in mind that at least half dozen fast implementations exist for some subsets of R.
- More examples at GitHub MathematicaVsR.