RHadoop

Дмитрий Ковалев
Институт проблем информатики РАН
dm.kovalev@gmail.com

18 ноября 2013 г.

ВВЕДЕНИЕ

R

R

- Набор операторов для вычислений над векторами и матрицами

R

- Набор операторов для вычислений над векторами и матрицами
- Развитые библиотеки с функциями анализа данных

R

- Набор операторов для вычислений над векторами и матрицами
- Развитые библиотеки с функциями анализа данных
- Мощные средства графического вывода

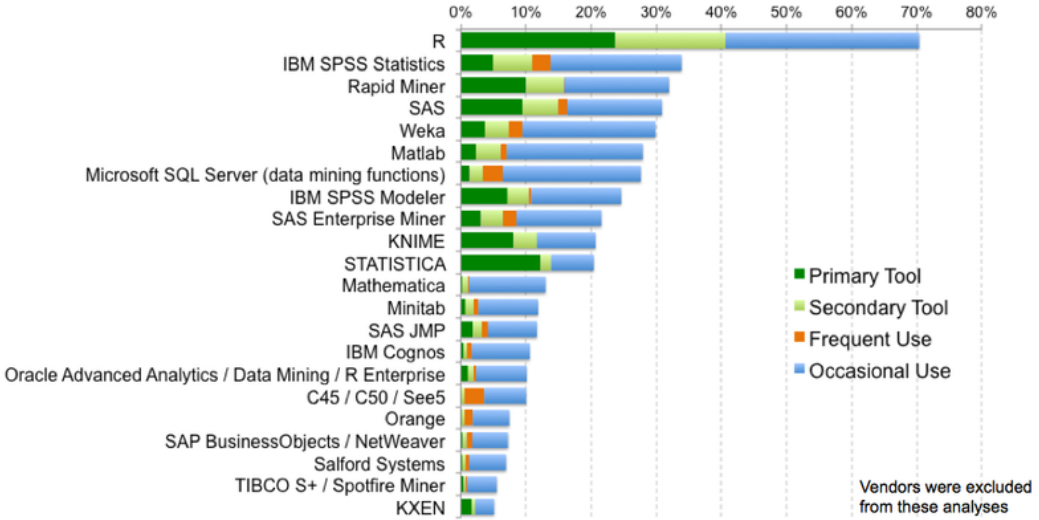
R

- Набор операторов для вычислений над векторами и матрицами
- Развитые библиотеки с функциями анализа данных
- Мощные средства графического вывода
- Простой и эффективный ЯП с условиям, циклами, рекурсивными UDF и средствами ввода/вывода

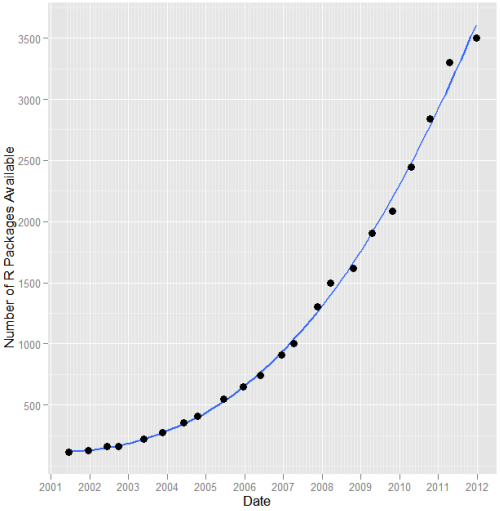
R

- Набор операторов для вычислений над векторами и матрицами
- Развитые библиотеки с функциями анализа данных
- Мощные средства графического вывода
- Простой и эффективный ЯП с условиям, циклами, рекурсивными UDF и средствами ввода/вывода
- Свободно распространяемое ПО

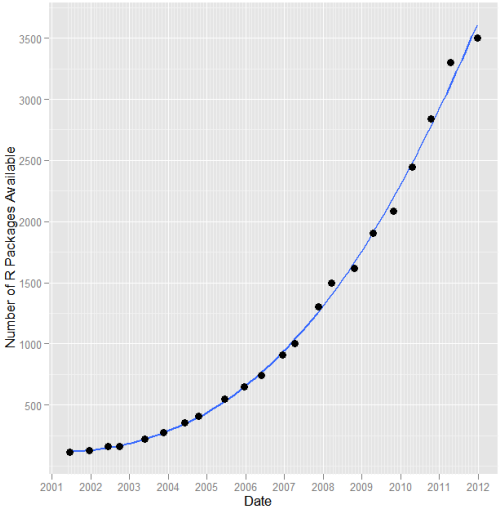
R


0%
10%
20%
30%
40%
50%
60%
70%
80%
R
IBM SPSS Statistics
Rapid Miner
SAS
Weka
Matlab
Microsoft SQL Server (data mining functions)
IBM SPSS Modeler
SAS Enterprise Miner
KNIME
STATISTICA
Mathematica
Minitab
SAS JMP
IBM Cognos
Oracle Advanced Analytics / Data Mining / R Enterprise
C45 / C50 / See5
Orange
SAP BusinessObjects / NetWeaver
Salford Systems
TIBCO S+ / Spotfire Miner
KXEN
Primary Tool
Secondary Tool
Frequent Use
Occasional Use
Vendors were excluded from these analyses

Пакеты R

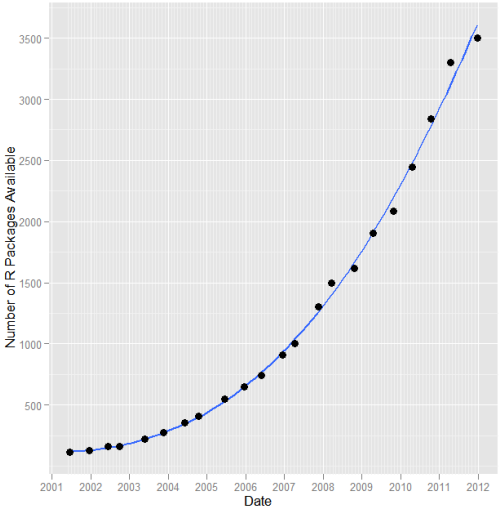

Number of R Packages Available
3500
3000
2500
2000
1500
1000
500
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
Date

Пакеты R




```
x <- available.packages()
length(unique(rownames(x)))
```

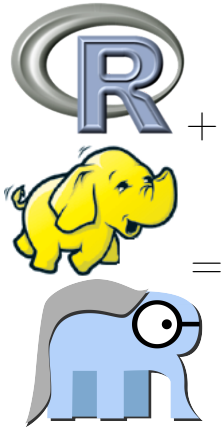
Пакеты R



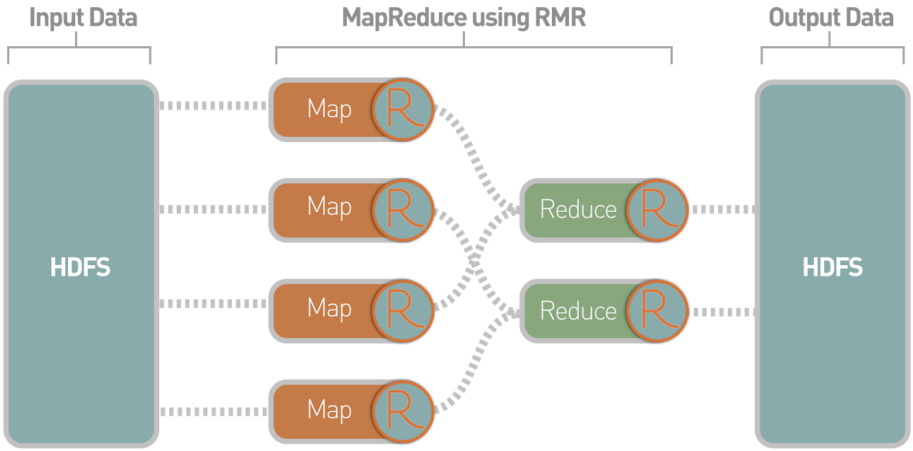

```
x <- available.packages()
length(unique(rownames(x)))
```

```
[1] 4972
```

Hadoop



R+Hadoop


Input Data
MapReduce using RMR
Output Data
HDFS
Map
Map
Map
Map
Reduce
Reduce
HDFS

Пакеты RHadoop

Hadoop	**RHadoop**
hdfs	rhdfs
hbase	rhbase
mapreduce	**rmr2**, plymr

RMR2

Локально:

Локально:

```
sapply(1:10, function(x) x^2)
```

Локально:

```
sapply(1:10, function(x) x^2)
```

```
[1]  1  4  9 16 25 36 49 64 81 100
```

Локально:

```
sapply(1:10, function(x) x^2)
```

```
[1]  1  4  9 16 25 36 49 64 81 100
```

На кластере:

Локально:

```
sapply(1:10, function(x) x^2)
```

```
[1]  1  4  9 16 25 36 49 64 81 100
```

На кластере:

```
numbers = "path/to/data"
```

Локально:

```
sapply(1:10, function(x) x^2)
```

```
[1]  1  4  9 16 25 36 49 64 81 100
```

На кластере:

```
numbers = "path/to/data"
```

```
out = mapreduce(numbers, map = function(k,v) v^2)
from.dfs(out)
```

Локально:

```
sapply(1:10, function(x) x^2)
```

```
[1]  1  4  9 16 25 36 49 64 81 100
```

На кластере:

```
numbers = "path/to/data"
```

```
out = mapreduce(numbers, map = function(k,v) v^2)
from.dfs(out)
```

```
$key
NULL

$val
 [1]   1   4   9  16  25  36  49  64  81 100
```

Wordcount

```
wordcount =
  function(
      input,
          output = NULL,
              pattern = " ") {
```

Wordcount

```
wordcount =
  function(
      input,
          output = NULL,
              pattern = " ") {
```

```
mapreduce(
      input = input,
            output = output,
                  map = wc.map,
                        reduce = wc.reduce,
                              combine = T)}
```

Wordcount

```
wc.map =
    function(., lines) {
        keyval(
            unlist(
                strsplit(
                    x = lines,
                    split = pattern)),
            1)}
```

Wordcount

```
wc.map =
    function(., lines) {
        keyval(
            unlist(
                strsplit(
                    x = lines,
                    split = pattern)),
            1)}
```

```
wc.reduce =
    function(word, counts ) {
        keyval(word, sum(counts))}
```

Набор данных:

```
mtcars
```

Набор данных:

```
mtcars
```

```
                   mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Duster 360        14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
...
```

Локально:

Локально:

```
aggregate(mtcars$mpg, list(mtcars$cyl), mean)
```

Локально:

```
aggregate(mtcars$mpg, list(mtcars$cyl), mean)
```

```
  Group.1        x
1       4 26.66364
2       6 19.74286
3       8 15.10000
```

Локально:

```
aggregate(mtcars$mpg, list(mtcars$cyl), mean)
```

```
  Group.1        x
1       4 26.66364
2       6 19.74286
3       8 15.10000
```

На кластере:

Локально:

```
aggregate(mtcars$mpg, list(mtcars$cyl), mean)
```

```
  Group.1        x
1       4 26.66364
2       6 19.74286
3       8 15.10000
```

На кластере:

```
input = "/path/to/data"
```

Локально:

```
aggregate(mtcars$mpg, list(mtcars$cyl), mean)
```

```
  Group.1        x
  1       4 26.66364
  2       6 19.74286
  3       8 15.10000
```

На кластере:

```
input = "/path/to/data"
```

```
out = mapreduce(input, map = function(k,v) keyval(v$cyl, v$mpg), reduce=
    function(k,v) keyval(k, mean(v)))
```

Локально:

```
aggregate(mtcars$mpg, list(mtcars$cyl), mean)
```

```
  Group.1        x
  1       4 26.66364
  2       6 19.74286
  3       8 15.10000
```

На кластере:

```
input = "/path/to/data"
```

```
out = mapreduce(input, map = function(k,v) keyval(v$cyl, v$mpg), reduce=
    function(k,v) keyval(k, mean(v)))
```

```
from.dfs(out)
```

Локально:

```
aggregate(mtcars$mpg, list(mtcars$cyl), mean)
```

```
  Group.1        x
  1       4 26.66364
  2       6 19.74286
  3       8 15.10000
```

На кластере:

```
input = "/path/to/data"
```

```
out = mapreduce(input, map = function(k,v) keyval(v$cyl, v$mpg), reduce=
    function(k,v) keyval(k, mean(v)))
```

```
from.dfs(out)
```

```
$key
[1] 4 6 8

$val
[1] 26.66364 19.74286 15.10000
```

Композиция MapReduce

```
y = mapreduce(input, map = function(k,v) v[v$cyl > 4 ,])
x = mapreduce(y, map = function(k,v) keyval(v$cyl, v$mpg), reduce=function(k,v
    ) keyval(k, mean(v)))
```

Композиция MapReduce

```
y = mapreduce(input, map = function(k,v) v[v$cyl > 4 ,])
x = mapreduce(y, map = function(k,v) keyval(v$cyl, v$mpg), reduce=function(k,v
    ) keyval(k, mean(v)))
```

```
from.dfs(x)
```

Композиция MapReduce

```
y = mapreduce(input, map = function(k,v) v[v$cyl > 4 ,])
x = mapreduce(y, map = function(k,v) keyval(v$cyl, v$mpg), reduce=function(k,v
    ) keyval(k, mean(v)))
```

```
from.dfs(x)
```

```
$key
[1] 6 8

$val
[1] 19.74286 15.10000
```

МЕТОД НАИМЕНЬШИХ КВАДРАТОВ

Метод наименьших квадратов

$$\mathbf{X}\boldsymbol{\beta} = \mathbf{y}$$

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{bmatrix}, \boldsymbol{\beta} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix}, \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Метод наименьших квадратов

$$\left(\mathbf{X^T X}\right)\hat{\boldsymbol{\beta}} = \mathbf{X^T y}$$

Метод наименьших квадратов

$$\left(\mathbf{X^T X}\right)\hat{\boldsymbol{\beta}} = \mathbf{X^T y}$$

```
solve(t(X)%*%X, t(X)%*%y)
```

Метод наименьших квадратов

```
Sum =
  function(., YY)
      keyval(1, list(Reduce('+', YY)))
```

Метод наименьших квадратов

```
XtX =
    values(
        from.dfs(
            mapreduce(
                input = X.index,
                map =
                    function(., Xi) {
                        Xi = Xi[,-1]
                        keyval(1, list(t(Xi) %*% Xi))},
                reduce = Sum,
                combine = TRUE)))[[1]]
```

Метод наименьших квадратов

```
Xty =
    values(
        from.dfs(
            mapreduce(
                input = X.index,
                map = function(., Xi) {
                    yi = y[Xi[,1],]
                    Xi = Xi[,-1]
                    keyval(1, list(t(Xi) %*% yi))},
                reduce = Sum,
                combine = TRUE)))[[1]]
```

Метод наименьших квадратов

```
solve(XtX, Xty)
```

Метод наименьших квадратов

```
solve(XtX, Xty)
```

```
X = matrix(rnorm(2000), ncol = 10)
X.index = to.dfs(cbind(1:nrow(X), X))
y = as.matrix(rnorm(200))
```

ДЗ