SparkR: Scaling R Programs with Spark

Shivaram Venkataraman, Zongheng Yang, Davies Liu, Eric Liang, Hossein Falaki, Xiangrui Meng, Reynold Xin, Ali Ghodsi, Michael Franklin, Ion Stoica, Matei Zaharia
Why R?

Statistics
data.frame
Visualization
CRAN packages

Implementation of “S” Statistical computing language (Bell Labs 1975)
Why R?

TOP 20 SKILLS OF A DATA SCIENTIST

- data analysis
- r
- python
- data mining
- machine learning
- statistics
- sql

Source: https://rjmetrics.com/resources/reports/the-state-of-data-science/
Big Data & R

Statistics
data.frame
Visualization
CRAN packages

+ Data
Big Data & R: Challenges

Data access: HDFS, Hive, S3

Capacity: Single Machine Memory

Parallelism: Single Thread
Approach
Approach 1: Parallel R API

Features

Parallel R APIs *foreach, apply*
Run custom R code, packages

Challenges:
Efficiency, performance
Functionality ?
Approach 1: Parallel R API

# lines: list of strings
ints <- apply(lines,
    function(line) {
        as.numeric(line[2])
    })
res <- sum(collect(Reduce(ints,
    function(x, y) {
        x + y
    }))))

Convert string to integer
Add up results
Approach 2: High level API

Features:
- Wrappers over SQL / ML algorithms
- Reuse query optimization, codegen etc.
- Easy to use, develop

Challenges:
- Custom R code / packages?
Approach 2: High level API

# lines: list of strings
linesDF <- as.DataFrame(lines)

res <- select(linesDF, sum(lines$age))
SparkR User API

DataFrames + Machine Learning
SparkR DataFrames

DataSources API

Column Functions, Aggregations

Translate to Spark SQL

```r
people <- read.df(
  "people.json",
  "json")

avgAge <- select(
  df,
  avg(df$age))

head(avgAge)
```
model <- glm(
  a ~ b + c,
data = df)

summary(model)

R Formulas
Concise specification of ML problem
Response \(a\) modeled by linear predictors \(b, c\)

Model Summaries
Print coefficients, standard errors etc.
Efficient distributed computation
SparkR UDFs

**DataFrame UDFs, UDAFs**
Run R functions on *partitions*
Users specify output schema
dapply, gapply

**Partition Aggregate**
Run R functions in parallel
Parameter tuning, Model averaging
spark.lapply
Architecture

Local

Worker

Worker
Architecture

Local

R

Spark Context

R-JVM bridge

Java Spark Context

Worker

Worker
Architecture

Local

R
Spark Context
R-JVM bridge

Java Spark Context

Spark Executor

Worker

Spark Executor

Worker
Implementation: R-JVM Bridge

Layer to call JVM methods directly from R

Supported across platforms, languages
Implementation: Closure Capture

From http://obeautifulcode.com/R/How-R-Searches-And-Finds-Stuff/
Evaluation
SparkR Scalability

Data: Flight arrivals from 2009-2014, 37.27M rows and 110 columns
Queries: Top-5 destinations, Aggregation, Count-Distinct
R-JVM Bridge

Data: Flight arrivals from 2009-2014, 37.27M rows and 110 columns
Query: Top-5 destinations from JFK on 64 cores
SparkR Status

Open source, part of Apache Spark from 1.4.0

>60 contributors including
UC Berkeley, Databricks, Alteryx, Intel, IBM etc.
SparkR

Big data processing from R

High-level APIs for SQL, ML

Custom R packages with UDFs

Try it out at [http://spark.apache.org](http://spark.apache.org)!