Data Science Workflows Using R and Spark

E. James (Jim) Harner

Department of Statistics
Department of Management Information Systems
West Virginia University

Octobrt 1, 2018

Outline

Fundamentals

Data Sources

The Components of this Tutorial

This tutorial covers data science workflows using R as both an analysis and graphics engine and as an interface to databases, Hadoop, Spark, etc.

The following are the required components of this tutorial:

- the DSAA tutorial slides (these slides)
- the tutorial notebook content
- the `rspark` computational environment
- the `rspark-docker` images

These components are discussed in the next few slides.

The R Computational Environment

The primary language used in this tutorial is R, but a working knowledge of bash scripts and SQL is useful. You should download and install R if you have not already done so. Go to: https://www.r-project.org

Likewise, you should install the RStudio IDE, which is found here: https://www.rstudio.com

The DSAA Tutorial Beamer Slides

The tutorial slides provide an overview of the tutorial content. The GitHub repo for the slides can be found here:
https://github.com/jharner/DSAA2018rspark-tutorial

Go to the site and click on the green button to clone the repo to your computer. Alternatively, clone this repo by issuing the following command from a terminal:

```
git clone https://github.com/jharner/DSAA2018rspark-tutorial.git
```

These slides are not the main source of content for this tutorial
The R markdown notebook documents in the next slide provide the detailed, execuatable content.

The Tutorial Notebook Content

The interactive, executable content of this tutorial is available in a GitHub repo called rspark-tutorial found at:
`https://github.com/jharner/rspark-tutorial`

As before, go to the site and click on the green button to clone the repo to your computer. You can also clone this repo with the following command:

```
git clone
https://github.com/jharner/rspark-tutorial.git
```

The `rspark-tutorial` consists of executable R markdown documents organized into modules containing sections. These tutorial documents are executed within `rspark`.

The rspark Computational Environment

This `rspark-tutorial` local repo can then be imported into the computional environment used in this tutorial, which is called rspark.

The `rspark` computational environment is available in several GitHub repos depending on whether you want to built the environment from scratch or you you want to download pre-built images. Assuming the latter, go to the rspark-docker repo:
`https://github.com/jharner/rspark-docker`

Go to the site and click on the green button to clone the repo to your computer. Alternatively, clone this repo by issuing the following command from a terminal:

```
git clone
https://github.com/jharner/rspark-docker.git.
```

Running rspark

In order to execute the content in the rspark-tutorial, the rspark computational environment must be run as a web applicaton. Follow the directions in the rspark-docker to launch rspark.

Once rspark is running, import the rspark-tutorial by clicking on the Files tab in RStudio server. The rspark-tutorial directory must be zipped before being imported. Click on the Upload option under the Files tab and navigate to the rspark-tutorial.zip file and upload it.

You can now execute the R markdown documents, i.e., those with the .Rmd suffix. These files can be executed interatively as notebooks or knitted to html, pdf, or Word documents.

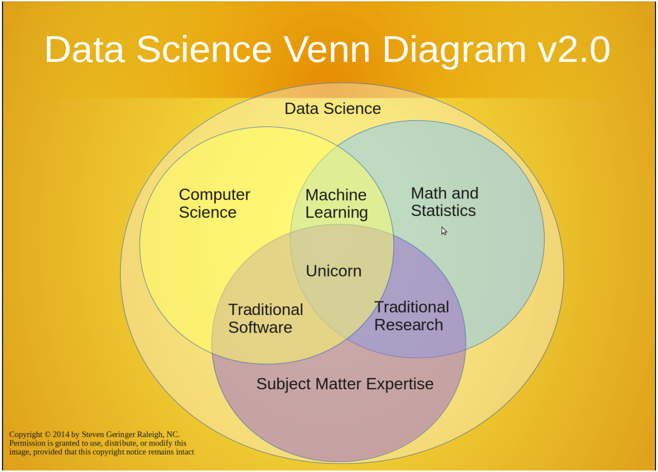
Running the DSAA Tutorial Slides

The slides cannot be executed in rspark's rstudio docker container since the full publishing capability of RStudio is not supported. In particular the LATEXBeamer package (and other components of LATEX) was not installed to keep the container size reasonable.

As a result, the DSAA2018rspark-tutorial repo must be executeds within a local version of RStudio.

What is Data Science?

Data science combines elements of statistics and computer science to develop methodologies to analyze large, complex data and streaming data within various subject-matter areas.



Data Science Workflows

This tutorial spans the entire data science workflow or process.

The notes for this section are here.

RStudio Server

RStudio is a powerful integrated development environment (IDE) primarily used for developing code for R projects, including R packages. RStudio supports the integration of R, Python, bash, and SQL code chunks into Rmarkdown documents (and notebooks), among other languages.

The notes for this section are here.

Linux

Data science requires underlying tools and the most basic of these is the operating system (OS). Linux is most commonly used since it is open source and has advanced features, e.g., its kernel and file system, which make handling big data feasible.

The notes for this section are here.

ML Basics

An *algorithm* is a procedure specifying a set of steps to accomplish a task.

Efficient algorithms that work sequentially or in parallel are the basis of pipelines to prepare, process, and analyze data.

The notes for this section are here.

Plain Text

Plain text files are the simplest way to store data. Generally, the data is stored in rows representing observations (or records) with columns representing variables (or fields). The beginning of the file may contain **metadata**, i.e., information about the data. It is sometimes called the **header**, which may represent the variable names.

The notes for this section are here.

JSON

JavaScript Object Notation (JSON) is a text format for the serialization of structured data. The design of JSON is a simple and concise text-based format, particularly when compared to XML.

The notes for this section are here.

Spreadsheets

Spreadsheets are widely used as a storage option. Microsoft Excel is commonly used spreadsheet software. The ‘readxl‘ package is part of the ‘tidyverse‘. It is used for importing tabular Excel files (‘.xls‘ and ‘.xlsx‘) into R as tibbles.

The notes for this section are here.

Databases

This section introduces relational data base management systems and NoSQL databases. The relational model is essential for multi-user transactional data, but it does not scale for big data. NoSQL databases are often distributed across a cluster.

The notes for this section are here.

Web Servies

Web Services is a recently introduced phrase. The *Web* and the *HyperText Transfer Protocol (HTTP)* that underlies the communication of data on the Web have become a vital part of our information network and day-to-day environment. Thus, being able to access various forms of data using HTTP is an important facility in a general programming language.

The notes for this section are here.