Hands-On Big Data

IASSIST Workshop, Minneapolis, MN
June 2, 2015

Ryan Womack

Data Librarian, Rutgers University, http://ryanwomack.com

Introduction

What this workshop IS:

- Fun, hopefully
- Introduction to the big data landscape
- Reviews major technologies
- Will familiarize you with some of the main players in the ecosystem
- Uses real interactions and environments to give a feel for working with Big Data

Introduction

What this workshop is NOT:

- a complete guide to big data
- an in-depth programming tutorial
- Does not provide "instant" big data expertise
- Does not give thorough theoretical background

Introduction

What you can hope to gain:

- Exposure to a wide-range of big data packages
- A sampling of tools and methods
- Understanding of the power, potential, and limitations of big data
- May help you decide how far and in which direction you want to go with big data
- Pathways for further learning

Introduction

What you can hope to gain, part 2:

- Hadoop, HDFS, MapReduce, MongoDB

Introduction

What you can hope to gain, part 2:

- Hadoop, HDFS, MapReduce, MongoDB
- HBase, Pig, Pig Latin, Hive, Spark, Scala, Sqoop

Introduction

What you can hope to gain, part 2:

- Hadoop, HDFS, MapReduce, MongoDB
- HBase, Pig, Pig Latin, Hive, Spark, Scala, Sqoop
- Oozie, ZooKeeper, Flume, Ambari, Hue

Introduction

What you can hope to gain, part 2:

- Hadoop, HDFS, MapReduce, MongoDB
- HBase, Pig, Pig Latin, Hive, Spark, Scala, Sqoop
- Oozie, ZooKeeper, Flume, Ambari, Hue
- Hortonworks, Cloudera, Tessera, RHadoop

Introduction

What you can hope to gain, part 2:

- Hadoop, HDFS, MapReduce, MongoDB
- HBase, Pig, Pig Latin, Hive, Spark, Scala, Sqoop
- Oozie, ZooKeeper, Flume, Ambari, Hue
- Hortonworks, Cloudera, Tessera, RHadoop
- AWS, EMR, EC2, Azure

Introduction

What you can hope to gain, part 2:

- Hadoop, HDFS, MapReduce, MongoDB
- HBase, Pig, Pig Latin, Hive, Spark, Scala, Sqoop
- Oozie, ZooKeeper, Flume, Ambari, Hue
- Hortonworks, Cloudera, Tessera, RHadoop
- AWS, EMR, EC2, Azure
- Tessera, Trelliscope, Lasso, PCA, SVD

Setup

- Workshop materials, including scripts and data, are available for download from http://github.com/ryandata/bigdata
- The script files contain working demonstrations of the concepts mentioned here.
- You should have some familiarity with the workings of your OS to complete some of the steps involved.

About Me

- I am also new to big data, and wanted to get beyond the slogan
- If I can do it, you can too!
- Taking the data librarian perspective, not the computer science perspective
- What can we learn about how Big Data really works?
- Enabling ongoing learning beyond this workshop

What is Big Data?

A buzzword...
A concept...
A set of practices...
An ecosystem...
Image of Big Data
Reality of Big Data
Big Data Landscape

Big Data is ...

Big Data is typically defined by the 3 V's

1. Velocity
2. Variety
3. Volume
4. to which a 4th is often added, Veracity

Some say Value is the 4th V
Computing definition: Big Data is data that is too big for a single computing instance to handle.
Personal Definition: Big Data is any data bigger than you know what to do with.
4 V's of Big Data

What to do with big data

One way to handle big data is with more powerful hardware.
Some problems are amenable to this.
Structured data, well-formulated modelling, interactions
E.g, a bank's central transaction database, some modeling problems
High-performance computing, parallel computing, and large scale databases
Processor is the bottleneck, not the data
This is expensive, not new (mainframe)
This kind of big data is not our focus here.

What's new in big data

Internet-scale activity produces large volumes of dispersed, unstructured data.
Log files, search records, user activity.
Facebook, Yahoo, Google, and others have pioneered this work.
"The Cloud"
How Big Data relates to Data Science
This workshop focuses on techniques developed to deal with this kind of data.

Hadoop

"Doug Cutting, Cloudera's Chief Architect, helped create Apache Hadoop out of necessity as data from the web exploded, and grew far beyond the ability of traditional systems to handle it. Hadoop was initially inspired by papers published by Google outlining its approach to handling an avalanche of data, and has since become the de facto standard for storing, processing and analyzing hundreds of terabytes, and even petabytes of data.
Apache Hadoop is 100% open source, and pioneered a fundamentally new way of storing and processing data. Instead of relying on expensive, proprietary hardware and different systems to store and process data, Hadoop enables distributed parallel processing of huge amounts of data across inexpensive, industry-standard servers that both store and process the data, and can scale without limits. With Hadoop, no data is too big."
http://www.cloudera.com/content/cloudera/en/about/hadoop-and-big-data.html

Hadoop and HDFS architecture

- Hadoop is an environment that enables a cluster of computers to function as a single large scale storage unit, with redundancy in case of failure, and centralized management. Name story.
- The Hadoop environment enables other processing and analytical jobs to run across the cluster in ways that are (relatively) transparent to the user.
- HDFS (Hadoop Distributed File System) is the underlying file system for Hadoop clusters.
- Hadoop Architecture
- A lot goes into actual administration of Hadoop cluster that we will abstract away from.

MapReduce

- MapReduce is the paradigm for many operations across a Hadoop cluster
- The Map component of the operation goes out to each node and performs its task, returning a result
- The Reduce component collects these results and aggregates/summarizes them from all nodes
- MapReduce functions are often written as Java programs and require both programming expertise and familiarity with the data

MapReduce Examples

- MapReduce Illustration
- MapReduce WordCount Program
- A short intro to MapReduce (great place to start)
- A longer intro to Hadoop ecosystem

Working with Hadoop

- All elements of the Hadoop ecosystem come with their own sets of command line tools.
- E.g., to place something into the hadoop file system, the command `hadoop fs -put` can be used.
- This workshop uses shortcuts and tools to avoid some of the inevitable command line work required for a production system.
- A toy Hadoop cluster, Yahoo Hadoop cluster
- Yahoo and Facebook were early large users of these technologies.
- Google has its own file system and more proprietary approaches. See Google BigQuery.

Amazon Web Services

Amazon Web Services (AWS) provides

- a variety of hosting environments
- quick start and stop of a variety of servers
- both raw computing power and pre-configured computing

Downside:

- Metered access - cheap for a short while, expensive in the long term
- Not as flexible as a self-hosted installation

We will use AWS for the bulk of our Hadoop/Hue/Pig/Hive demo.
AWS provides a feel for many of the issues involved in self-hosting, but with time-saving steps.

AWS setup

- You must obtain an AWS account from aws.amazon.com
- Follow the registration steps, then login.
- Setting up an account requires use of a phone for two-factor authenication. Please have one available.
- Demo-only users login at https://028335041858.signin.aws.amazon.com/console
- You will be provided with login and password.
- If needed, this is a reference to setting up AWS Command Line Interface

AWS EC2, EMR

- EC2 is Amazon's Elastic Compute Cloud. This allows you to create "instances", which are actual running computers with memory and disk storage, on the fly.
- You can create the number and size of machines that you need for your job, and pay a metered rate based on the number, size, and duration of the compute resources you use.
- EMR is Elastic Map Reduce. This uses EC2 computing resources to automate the creation of a Hadoop cluster that can perform MapReduce jobs.
- It should cost around $5 to run the standard EMR cluster for 2-3 hours in this workshop.
- Please be sure to terminate your instances when you are done. Leaving computers on is expensive in AWS!

AWS setup, key pair

- To create an Amazon EC2 key pair (Help Guide)
- Sign in to the AWS Management Console and open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
- From the Amazon EC2 console, select a Region.
- In the Navigation pane, click Key Pairs.
- On the Key Pairs page, click Create Key Pair.
- In the Create Key Pair dialog box, enter a name for your key pair, such as, mykeypair.
- Click Create.
- Save the resulting PEM file in a safe location.
- Your Amazon EC2 key pair and an associated PEM file are created.

AWS setup, create EMR cluster

- From the EMR page (https://console.aws.amazon.com/elasticmapreduce), click "Create Cluster"
- Accept default options, except under "Security and Access" choose the key that you just generated.
- Under EC2 Security group, click "Create Security Group"
- click "Create Security Group" again, fill out the fields, and click "Yes, Create"
- You want to create an open security group for ease of setup, not recommended in production environment
- Edit the rules to allow all inbound traffic from all locations - set "Source" field to the security group id just created (just for convenience - again, not recommended)
- Click "Create Cluster" and wait.

AWS setup, connections

- Click on the SSH link next to Master public DNS, and follow instructions to connect.
- On Windows Systems, use PuttyGEN to convert you .pem key to a .ppk key on Windows systems. Import the .pem key, then "Save private key".
- Then use PuttyExe to SSH in.
- On Mac/Linux systems, just use ssh.
- This will enable you to access the master node on your cluster directly and run command line operations.
- Next click "Enable Web Connection".
- Follow the instructions provided.
- Now you can click on Hue and the other web management interfaces.
- Congratulations, you are running a Hadoop cluster!

Amazon S3

- Amazon provides low cost, accessible storage through S3.
- One advantage of Amazon EMR is that it can use S3 storage.
- Items in s3 buckets can be made public.
- But be sure to edit the properties of the bucket itself to make sure the contents are listable by everyone.

Pig

As mentioned earlier, writing MapReduce programs in native Java, Python or other languages requires a high-level of expertise and can be time-consuming.

- Pig is designed to remove some of this barrier.
- Pig was developed at Yahoo!
- The language of Pig is Pig Latin.
- Pig is somewhat open-ended in structure and syntax (compared to Hive).
- Pig masks the parallelization of its actions behind simpler code.

Hue

- For our Pig example, we will use the Hue administration interface for AWS EMR, which runs through the browser.
- Run a word count on the Complete Works of Shakespeare
- Default login: hue, password: 1111
- The Pig Editor is available here.
- There is also a command line interface called Grunt.
- You can see a more full-featured Hue environment at http://demo.gethue.com

Hive

- The Hive language was developed at Facebook.
- Hive is very similar to standard SQL. Here is an SQL-to-Hive cheat sheet.
- Like Pig, it masks the complexity of working with a Hadoop cluster behind simpler code. Hive can handle less structured sources as if they were relational databases.
- The log files, however, show the complexity - also apparent when things break!
- The manuals at the Hive site are the most complete reference to the language.
- Again, we use Hue to access the Beeswax editor.

Spark

- Spark is a newer project that is getting a lot of attention. Spark can use Hadoop data stores, but is its own stand-alone system.
- Can mix SQL-type queries with more programming language type functions.
- Accepts Java, Python, and Scala functions. Spark is written in the Scala programming language.
- Generally much faster than Hadoop.
- The Spark demo requires access to command line tools.
  - This is much easier in Mac/Linux environments, but if you have Cygwin on Windows, you may be able to perform the same steps.
- Introduction to Big Data with Apache Spark started June 1 on EdX.

More Spark materials

- Spark in CDH
- Stanford Spark Class
- Spark on an EMR Cluster
- See p. 234 of *Guide to High Performance Computing* for Spark linear regression code

Administration

- We have already seen Hue.
- Oozie is a workflower scheduler for Hadoop jobs. You have already seen it in action in Hue.
- Zookeeper is a coordination service to help configure and manage distributed applications.
- Ganglia is a monitoring system for Hadoop clusters.
- Ambari is a web-based interface to several different monitoring and administration tools. We will see it in more detail in the Hortonworks Sandbox on Azure.

Other Data frameworks

- Cassandra is a large-scale database with a focus on column performance.
- HBase is a large-scale data store modeled on Google's BigTable for storing sparse data.
- MongoDB is a NoSQL database designed for large-scale operation.
- Sqoop is a tool for transferring data between Hadoop and traditional relational databases.
- Flume is a tool for automating the flow of data (such as server logs) from live systems into Hadoop data stores.
- The presence of so many tools and techniques (Impala, Mahout) for almost any big data task is what has made "Hadoop" a go-to solution for big data needs.

An aside on ACID vs. BASE

"Eventually consistent services are often classified as providing **BASE** (Basically Available, Soft state, Eventual consistency) semantics, in contrast to traditional **ACID** (Atomicity, Consistency, Isolation, Durability) guarantees. Eventual consistency is a consistency model used in distributed computing that informally guarantees that, if no new updates are made to a given data item, eventually all accesses to that item will return the last updated value. Eventual consistency is widely deployed in distributed systems "
R for Cloud Computing, p. 210

Microsoft Azure, Google Cloud

Other cloud services are available to support big data.

- Microsoft Azure provides free trial access to Hortonworks, albeit with highly aggressive scripting. See also Portal.azure.com.
- Google Cloud can also be used to provision Hadoop clusters.
- These services are newer and (perhaps) less robust than Amazon, with fewer help resources and more hoops to jump through.
- Azure has more Microsoft server options than Linux options.

Cloudera

- Cloudera (http://www.cloudera.com)
- A leading provider of Hadoop-based big data solutions. Founded 2009.
- Distributes CDH (Cloudera Distribution with Hadoop). This means rolling together a number of projects in a coherent "stack", distributing, and providing support. Contributes to HBase and other Apache projects.
- Cloudera Live Click "Try the Demo" or directly to http://demo.gethue.com
- Hue is one management interface for Hadoop.

Hortonworks

- Hortonworks (http://www.hortonworks.com)
- Founded in 2011 by 24 Yahoo! engineers from the original Hadoop team. Like Cloudera, Hortonworks manages a "stack" of Hadoop applications and provides support. In some ways, like a linux distribution. Hortonworks is a leading contributor to Hive and other Apache projects.
- Hortonworks is available in Sandbox mode for Virtual Machines and in the Azure cloud (with free one month trial).

Ambari revisited

Using Hortonworks, either via downloaded Virtual Machine, or with the Azure cloud trial, try out Ambari.
Launch the instance, then navigate to the browser interface.

- 127.0.0.1:8888 provides starter interface on a VM (use URL provide if on Azure)
- 127.0.0.1:8000 provides script interface
- 127.0.0.1:8080 provides Ambari interface (login: admin, password: admin)

R and Big Data

- R is the leading open source statistical programming platform, and so provides a natural complement to open source big data software.
- Some R projects and extensions are devoted to providing large file access, parallel computation, and other aspects of HPC.
- There are many of these (listing on Task Views), not the focus here.
- We will discuss two projects that fit in with Hadoop-style big data on clusters:
  - RHadoop
  - Tessera

RHadoop

- RHadoop is a collection of five packages developed by Revolution Analytics:
  - `ravro`, `rhbase`, `rhdfs` are packages to write and read data for their respective formats.
  - `rmr` provides an interface to the map reduce framework through R
  - `plymr` provides a higher-level set of functions that reduces the programming requirements of rmr, "plyr meets MapReduce"
- Other packages include SparkR, RHive, RCassandra, and more
- See p. 203 and following in *R for Cloud Computing*

R in the Cloud

- One way to run R in the cloud is with an Amazon Machine Image (AMI).
- These pre-built installations make it easy to get R up and running in seconds in AWS.
- See Louis Aslett's RStudio Server versions for a particularly useful version.
- Here are some more instructions (not personally tested).
- RHadoop in EC2 instructions (not personally tested).

Tessera

Tessera is developed by Purdue, Pacific Northwest National Laboratory, and Mozilla. Launched in November 2014, this project holds a lot of promise.

- Running in the R environment, Tessera provides its own commands that execute across a cluster, easing the burden of analysis in this environment.
- The datadr package "divides and recombines" in a manner similar to MapReduce, providing a simplified interface to Hadoop.
- `RHIPE` is the package that interacts directly with Hadoop.
- Tessera also provides a visualization interface, Trelliscope, that can handle views across many variables and observations. Described in this paper.
- Tessera's Bootcamp is a good introduction, or try the quickstart.

High-dimensional data

- Many data analysis problems exhibit high dimensionality.
- When the number of variables is greater than the number of observations, this is high-dimensional data ($p > n$).
- A typical problem - trying to determine which of 10,000 gene SNPs (single nucleotide polymorphisms) helps to explain 100 occurrences of a rare cancer.
- Principal Component Analysis (PCA) is a classic, early method of selecting the most relevant elements in the dataset to explain variation, and is still widely used [`prcomp` in R, SAS PROC FACTOR].

High-dimensional data, cont.

- The Lasso is another popular method that performs variable reduction and selection simultaneously.

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

- see "Absolute Penalty Estimation", *International Encyclopedia of Statistical Science*.
- We (artificially) penalize having large numbers of variables, so the model will shrink to include only variables with significant influence on the outcome.
- Accurate prediction and computational feasibility make Lasso and variants popular [`lars` in R, SAS GLMSELECT].
- Many, many other variant methods have been introduced, e.g. ℓ_1/ℓ_2 penalty procedures. See *Modern Multivariate Statistical Techniques* and *Statistics for High-Dimensional Data.*

Sparse data

"In numerical analysis, a sparse matrix is a matrix in which most of the elements are zero. By contrast, if most of the elements are nonzero, then the matrix is considered dense. The fraction of zero elements over the total number of elements in a matrix is called the sparsity (density)" [Wikipedia]

- An example, Netflix movie ratings data. Only a small percentage of all movies are rated by any one user.
- Squeeze the zeros out of the matrix and represent it more compactly [`Matrix` or `sparseMatrix` packages in R].
- Singular value decomposition (SVD) and other mathematical/computational techniques can then be applied to solve the problem [`svd` or `irlba` in R].
- An active area of research.
- There is a sparse matrix collection on AWS if you would like examples to explore.

Big Data in Practice

We will run one quick example of querying an actual big dataset.

- Over 500 TB of web crawl data are available through the Common Crawl project, also available via Amazon S3.
- This is a quick way to use the index. The Support Library provides more extensive access.
- Some more tutorials here.
- Need boto (execute `pip install boto`) for this, per this link.

Big Data in Practice, cont.

- AWS provides other public datasets such as the 1000 Genomes project, which has a tutorial and a browser search interface too.
- Building a cluster would allow you to use MapReduce type functions to analyze the data.
- Another AWS example on using R on EC2 to analyze global weather.

Conclusion

- Today's examples are meant to give exposure to and a feeling for big data computing.
- To actually use these technologies to good effect requires greater immersion and training.
- Setting up a large data store takes time and expertise.
- Developing analytics on top of that data store takes expertise and time.
- Preferably a committed *team* effort.

Termination

- Remember to *terminate* any AWS instances you have created!
- Otherwise you will continue to be billed for usage, $30/day and up.

References

1. Peter Bühlmann and Sara van de Geer. *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer, 2011.
2. Michael Frampton. *Big Data Made Easy: A Working Guide to the Complete Hadoop Toolset*. Apress, 2015.
3. Thilina Gunarathne. *Hadoop v2 MapReduce Cookbook*. Second Edition. Packt, 2015.
4. Richard Hill, Laurie Hirsch, Peter Lake, and Siavash Moshiri. *Guide to Cloud Computing: Principles and Practice*. Computer Communications and Networks. Springer, 2013.
5. Alan J. Izenman. *Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning*. Springer, 2008.
6. A. Ohri. *R for Cloud Computing: an Approach for Data Scientists*. Computer Communications and Networks. Springer, 2014.
7. K. G. Srinivasa and Anil Kumar Muppalla. *Guide to High Performance Distributed Computing: Case Studies with Hadoop, Scalding, and Spark*. Computer Communications and Networks. Springer, 2015.