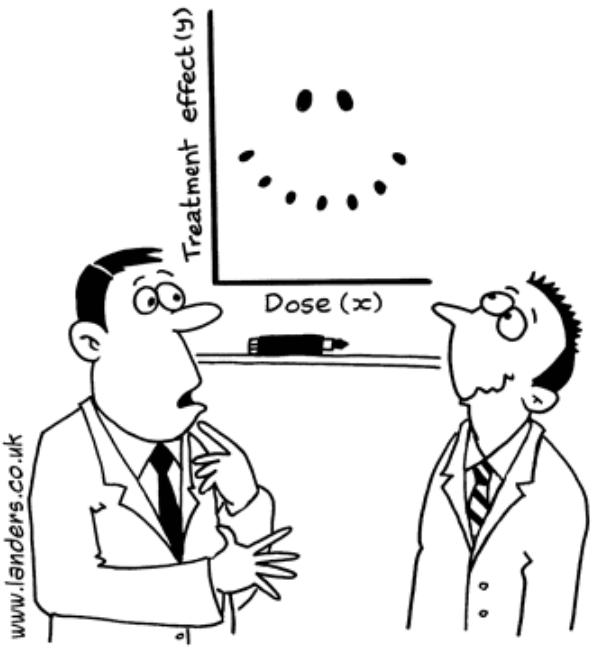
Analytics Training



"It's a non-linear pattern with outliers.....but for some reason I'm very happy with the data."

TOOLS & TECHNIQUES

Popular Applications Of Linear Regression For Businesses

By **Neha Shitut** Last updated **Nov 30, 2018**

In Statistics, Linear regression refers to a model that can show relationship betv n impact the other. In essence, it involves showing how the variation in the "dep y change in the "independent variables".

In Business, this dependent variable can also be called the predictor or the factor :, pricing, performance, risk etc. Independent variables are also called explanato e factors that influence the dependent variable along with the degree of the in g "parameter estimates" or "coefficients". These coefficients are tested for statistic e intervals around them so that the model that we are building is statistically rob e elasticity based on the coefficient can tell us the extent to which a certain fact a

negative coefficient can be interpreted to have a negative or an inverse relation with the dependent variable and positive coefficient can be said to have a positive influence. The key factor in any statistical models is the right understanding of the domain and its business application.

Linear Regression is a very powerful statistical technique and can be used to generate insights on consumer behaviour, understanding business and factors influencing profitability. Linear regressions can be used in business to evaluate trends and make estimates or forecasts. For example, if a company's sales have increased steadily every month for the past few years, by conducting a linear analysis on the sales data with monthly sales, the company could forecast sales in future months.

Linear regression can also be used to analyze the marketing effectiveness, pricing and promotions on sales of a product. For instance, if company XYZ, wants to know if the funds that they have invested in marketing a particular brand has given them substantial return on investment, they can use linear regression. The beauty of linear regression is that it enables us to capture the isolated impacts of each of the marketing campaigns along with controlling the factors that could influence the sales. In real life scenarios there are multiple advertising campaigns that run during the same time period. Supposing two campaigns are run on TV and Radio in parallel, a linear regression can capture the isolated as well as the combined impact of running this ads together.

Linear Regression can be also used to assess risk in financial services or insurance e company might conduct a linear regression to come up with a suggested pren o Insured Declared Value ratio. The risk can be assessed based on the attribut r demographics. The results of such an analysis might guide important business deci

In the credit card industry, a financial company maybe interested in minimizing the risk portfolio and wants to understand the top five factors that cause a customer to default. Based on the results the company could implement specific EMI options so as to minimize default among risky customers.

While Linear regression has limited applicability in business situations because it can work only when the dependent variable is of continuous nature, it still is a very well known technique in the situations it can be used. It assumes a linear relation between the independent and dependent variables. It must be noted that sometimes transformations can also be applied to non linear relationships to make them applicable in a linear regression model.

Related Articles:

Logistic Regression in SAS

Predictive Analytics- A Common Practice in Companies

Regression Modeling

Interested in learning about other Analytics and Big Data tools and techniques? Click on our course links and explore more.
Jigsaw's Data Science Course – click here.

Jigsaw's Big Data Course – click here.

analytic techniques | blogs on analytics | Online Analytics Training

Neha Shitut

Neha is a Senior Faculty at Jigsaw, holds a Masters in Econometrics and Financial Economics and Bachelors in Economics and Applied Statistics from University of Mumbai. With 9 years of professional analytics experience under her belt, 6 of which were in Retail Analytics at Genpact LLC, she has a thorough understanding of Statistical Predictive Modelling techniques and their application including Marketing Mix, Store Group Analysis and Pricing and Promotion studies. She is well equipped in SAS, R, Applied Statistics, Applied Econometrics, Regression, Clustering, Decision Tree, SAS Macros, SQL and has over 500+ hours of teaching to her credit. She believes herself to be a lifelong learner and takes a special interest in yog
Ananda Sangha, an NGO based on spiritual teachings where she manages
webinars on several topics on integrating spirituality into daily life.

Leave a comment

TOOLS & TECHNIQUES | BIG DATA ANALYTICS

Using Flume Beyond Ingesting Data Streams Into Hadoop

By **Mathivanan** Last updated **Dec 3, 2018**

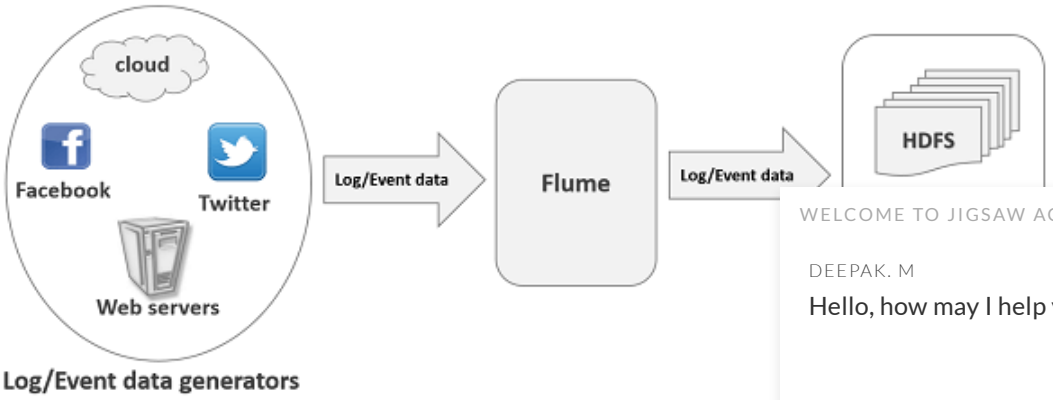
Apache Flume and Streaming Data:

Apache Flume, as its website mentions – is a distributed, reliable, and available system for efficiently collecting, aggregating and moving large amounts of log data from many different sources to a centralized data store such as Hadoop HDFS. The application of Apache Flume is restricted not only to log data aggregation. It can be used to transport large volumes data streams from any possible data source.

In an organization, we would have a number of data sources that generate a variety of streaming data at high velocity and high volume. Processing streaming data has become very important and esser

- In the banking/financial services domain, monitoring credit card transactions continuously from various sources like transactions at points-of-sale, ecomm detect and flag credit card frauds
- Purchase behaviour from the above type of data streams can also be used by sales/discount campaigns to clients as part of their up-selling or cross-selling
- In the IT sector, several network or mail servers of data centres keep generati helpful in identifying slow moving traffic, bottlenecks and locations where ad ease or handle the traffic.
- Capturing web-click streams by ecommerce sites to identify browsing patter send promotion campaigns that are pinpointed to their needs based on this d

With Flume, we can not only capture and ingest the streaming data into a data store like Hadoop HDFS but we can also do certain amount of in-flight processing of this data. We will discuss how we can use Apache Flume with a specific attention to on this feature.



Components of Flume:

To process a data stream, we run a Flume agent, which is a long running process. A
event. A flume agent hosts the components through which events flow from an ext
(sink).

To set up a flume agent, we need to write a configuration file specifying the properties of the source, channel and the sink. We can also specify properties of *interceptors* which gives the ability to modify and/or drop events in-flight.

Once the configuration is written, we can run a Flume agent using the command flume-ng which comes with Flume installation with the syntax given below.

flume-ng agent -n <agent_name> -c conf -f <flume-conf.properties>

Following are the details of each component.

Agent: The Agent receives events from clients or any other agents. Flume-Agent is an independent daemon process, which is installed on each node to collect the events. Any Java Virtual Machine (JVM) that runs flume, consists of sources, sinks, channels and other important components through which events get transferred from one place to another.

Source: The source is the component of an Agent which receives data from the data generators and transfers it to one or more channels in the form of Flume events. The major sources are:

- NetCat Source
- Spooling Directory Source
- Syslog Source

Flume also allows us to specify advanced sources of data streams such as Twitter, Kafka and so on.

Example – Properties of a source in a typical Flume configuration:

 # Name the components on this agent

agent1.sources = r1

 # Describe/configure the source – SpoolDir

agent1.sources.r1.type = spooldir

agent1.sources.r1.spoolDir = /home/hduser/Desktop/flume/input

Channel: Channels are communication bridges between sources and sinks within a
various channels, like:

- Memory Channel
- JDBC Channel
- Kafka Channel
- File Channel

Example – Properties of a channel in a typical Flume configuration:

agent1.channels = c1

 # Use a channel which buffers events in file

agent1.channels.c1.type = file

agent1.channels.c1.capacity = 1000

agent1.channels.c1.transactionCapacity = 100

agent1.channels.c1.checkpointDir = /home/hduser/Desktop/flume/checkpoint/

agent1.channels.c1.dataDirs = /home/hduser/Desktop/flume/dataDir/

agent1.channels.file.writeFormat = Text

Sinks: Events are transferred to HDFS. Sinks supports multiple file formats like Text, AVRO, JSON, Compressed files. There are multiple sinks available that deliver data to a wide range of destinations, such as :

- HDFS Sink
- HBase Sink
- Logger Sink

Example – Properties of a sink in a typical Flume configuration:

 # Describe the HDFS sink

agent1.channels = c1

agent1.sinks = k1

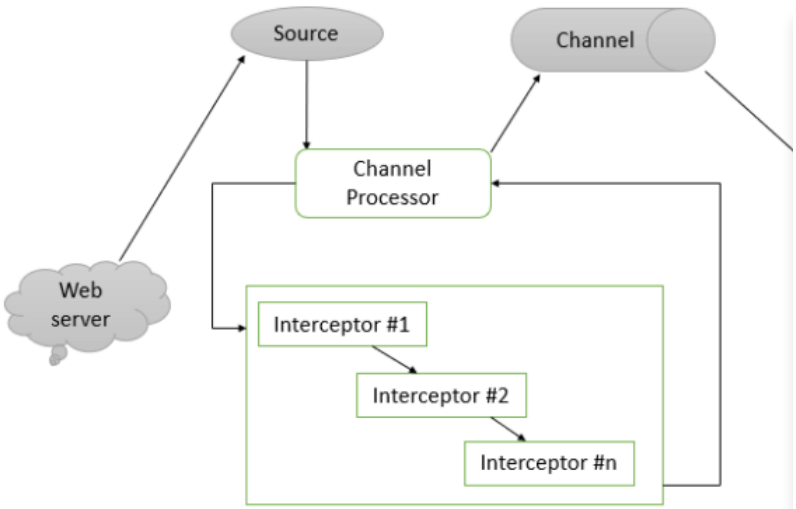
agent1.sinks.k1.type = hdfs

agent1.sinks.k1.channel = c1

agent1.sinks.k1.hdfs.path = /user/hduser/regexpfilter

agent1.sinks.k1.hdfs.fileType= DataStream

agent1.sinks.k1.hdfs.writeFormat = Text



Flume Interceptors:

The events received by sources can be transformed or dropped by interceptors before they are written to the corresponding channels. Interceptors are simple components that sit between a source and the channels. Interceptors are commonly used to analyse events, we can do any filter, aggregation operations in Interceptors.

Types of Interceptors:

Timestamp Interceptor:

It inserts the timestamp into the Flume event headers with the timestamp key. The timestamp key is the header that the HDFS Sink uses for bucketing. If the timestamp header is already present, this interceptor will replace it unless the preserve Existing parameter is set to false. To add a timestamp interceptor, use the alias timestamp.

The configuration parameters are:

agent1.sources.source_name.interceptors = Interceptor_timestamp

agent1.sources.source_name.interceptors. Interceptor_timestamp.type = timestamp

agent1.sources.source_name.interceptors. Interceptor_timestamp.preserveExisting = false

Host Interceptor

The host interceptor inserts the hostname or IP address of the server on which the agent is running into the Flume event headers. The key to be used in the headers is configurable using the host Header parameter, but defaults to host. We can insert the hostname instead of the IP address, set useIP to false.

The configuration parameters are:

agent1.sources.source_name.interceptors = Interceptor_host

agent1.sources.source_name.interceptors.Interceptor_host.type = host

agent1.sources.source_name.interceptors.Interceptor_host.useIP = false

agent1.sources.source_name.interceptors.Interceptor_host.preserveExisting = tru

UUID Interceptor:

UUID abbreviated as Universally Unique Identifier is used to set unique identifier intercepted and assigned an UUID of 128-bit value. This enables deduplicating do duplicated because of replication and redelivery in a Flume network that is design performance.

The configuration parameters are:

agent1.sources.source_name.interceptors = Interceptor_uuid

agent1.sources.source_name.interceptors. Interceptor_uuid.headerName = event

agent1.sources.source_name.interceptors. Interceptor_uuid.prefix = usingFlume

agent1.sources.source_name.interceptors. Interceptor_uuid.preserveExising = false

Regex filtering interceptor:

The previous discussed Interceptors push the entire events to sinks, which may result in huge data in sink. The Regex filtering interceptors can be used to make sure only important events are passed through Flume agents to reduce the volume of data being pushed into HDFS.

The configuration parameters are:

agent1.sources.source_name.interceptors = include exclude

agent1.sources.source_name.interceptors.include.type = regex_filter

agent1.sources.source_name.interceptors.include.regex = .*Pattern1.*

agent1.sources.source_name.interceptors.include.excludeEvents = false

agent1.sources.source_name.interceptors.exclude.type = regex_filter

agent1.sources.source_name.interceptors.exclude.regex = .* Pattern2.*

agent1.sources.source_name.interceptors.exclude.excludeEvents = true

Use Case:

An American Internet Service Provider (ISP) offers internet service to its clients across the globe. It has servers in every region and a huge network traffic in their servers at some point of time, which leads their servers brings down. They want to analyse the traffic congestion on their network using the log data generated by the servers. So that in future they can increase the number of servers in the regions.

The Log files generate the details of IP address, time and date, the web URL, the port number and the browser details.

123.223.223.123 – – [13/May/2016:00:23:50 -0400] “GET /wallpapers/flowers.jpeg HTTP/1.0” 200 1031 “http://www.LinkedIn.com/” “Chrome/68.0 (Macintosh; I; PPC)”

123.123.123.101 – – [13/May/2016:00:23:50 -0400] “GET /wallpapers/flowers.jpeg HTTP/1.0” 200 1031 “http://www.LinkedIn.com/” “Chrome/68.0 (Macintosh; I; PPC)”

123.223.223.123 – – [13/May/2016:00:23:50 -0400] “GET /wallpapers/flowers.jpeg HTTP/1.0” 200 1031 “http://www.LinkedIn.com/” “Chrome/68.0 (Macintosh; I; PPC)”

123.123.123.100 – – [13/May/2016:00:23:48 -0400] “GET /pics/wpaper.gif HTTP “http://www.LinkedIn.com/” “Chrome/68.0 (Macintosh; I; PPC)”

123.123.123.112 – – [13/May/2016:00:23:47 -0400] “GET /asctortf/ HTTP/1.0” 2 “http://search.netscape.com/Computers/Data_Formats/Document/Text/RTF” “Ch

123.223.223.123 – – [13/May/2016:00:23:48 -0400] “GET /downloads/sample.gif “http://www.nfl.com/news” “Chrome/68.0 (Macintosh; I; PPC)”

How to run the Agents:

Step-1: Start the flume agent.

flume-ng agent –conf conf –conf-file spool_dir.conf –name agent1 -flume.root.logger=INFO,console

Step-2: Simulate the input data stream.

Run the one of the following shell scripts in order to simulate data stream source relevant to each flume agent, from the Linux shell prompt.

To simulate data stream for data source spooldir

$ ./spool_files.sh log.txt

To simulate data stream for data source netcat

$ ./socket_stream.sh log.txt | telnet localhost 7777

To simulate data stream for data source exec for generating a continuously growing file.

$ ./tail_file.sh log.txt

Step-3: To check the output files in the HDFS location

Hadoop fs -ls /user/hduser/regexpfilter

Appendix

Attached are the following files.

Flume Downloads

Apache Flume | big data | big data tools | data streaming

Dhananjay Basrur

Dhananjay has over four years of content and copywriting experience, having also been involved with customer support and product design teams. He has worked in a number of roles, including UX writing, social media management, content curation, and more. In his spare time, he's almost al

Leave a comment