Benchmark Brief

Highlights

- Performance results with up to two concurrent analysts using a table with over 100 million ratings.
- Four recommender methods tested.
- Private and shared SAS® LASR servers tested.

SAS® In-Memory Statistics for Hadoop

Performance Results using RECOMMEND Procedure running on a Four-node HP Cluster with 17,770 items.

This benchmark is intended to provide an understanding of expected performance and the system resources required to support building recommender systems using SAS In-Memory Statistics for Hadoop. The scenarios tested are designed to simulate an environment under heavy use. Analysts create a recommender system and define prediction methods with PROC RECOMMEND. Real-world data volumes and variables were used. SAS In-Memory Statistics for Hadoop offers analysts the option to either share an input data table that resides in-memory or use their own private table. This document covers both scenarios. PROC RECOMMEND performance characteristics are described including: method building response times, memory utilization and the system architecture required to support the workloads tested.

Scenario

The test scenario simulates multiple users creating recommender systems and executing a specific series of PROC RECOMMEND statements to create methods and generate recommendations. The first scenario tested consists of a single analyst running the designated series of PROC RECOMMEND actions. Subsequent scenarios and another analyst simultaneously running the same series of actions.

Data Description

The recommender system includes two data tables with the following characteristics:

Item Table

This table contains 17,770 items that receive ratings from users. The columns include the item ID, item name and the year it was made available. The file size is 4.9MB.

Rating Table

This table consists of four columns including an item ID, user ID, rating, and dated it was rated. The rating scale data values range from one through five. The table is 3.2GB in size with 100,480,507 rows.

Before running any test scenarios the input data used by analysts is first written to SASHDAT format within the Cloudera Hadoop Distributed File System (HDFS). The data is collocated and distributed across the same cluster where SAS In-Memory Statistics for Hadoop is installed. For test scenarios where analysts share a single table, data is lifted into memory creating a SAS LASR server that is shared by analysts. For test scenarios where each analyst has their own private data, multiple copies of the table are lifted into memory, one copy for each analyst. For example, in the two-user test where each analyst uses his own private table, one is created for each user.

System Configuration and Architecture

Software

- SAS/ACCESS® Interface to Hadoop
- SAS In-Memory Statistics for Hadoop
- SAS/STAT®
- SAS/GRAPH®
- Base SAS®

Cloudera Cluster 4.5

System Configuration

Hardware

HP DL380p Gen8 25-SFF CTO servers (four nodes: one SAS server root/name nodes, and three compute nodes.)

CPU: Intel Xeon E5-2690V2 3.0Ghz/10 core/256MB (20-cores per node)

Memory: 384GB RAM at 1866MHz (per node), total 1536GB (1152GB available for processing on three compute/data nodes).

Storage: 14 HP 1TB 6G SAS 7.2K 2.5in SC MDLH (per node),

- Two local disks allocated to operating system.
- There are 12 local disks configured as JBOD RAW storage for Hadoop.

Test Execution

Each test begins by executing PROC LASR statements to lift data from HDFS to create a SAS LASR server. Multiple servers are created for scenarios where each analyst uses their own data. Each SAS LASR server is created with the default nthreads option. This allows IMSTAT actions to use processors on all compute nodes in the cluster.

Once the LASR table is created, each analyst runs the following series of PROC RECOMMEND statements in the order shown. The series of statements are assembled, submitted and executed as a SAS program. Each analyst submits a program containing the following steps:

ADD: Adds the customer rating table to a new recommender system.

ADDTABLE: Adds the item profile table to the recommender system.

METHOD SLOPEONE: Creates a slopeone method, a simple regression-based method.

METHOD KNN1: Creates a KNN method labeled KNN1 with SIMILARITY=PC.

METHOD KNN2: Creates a KNN method labeled KNN2 with SIMILARITY=COSINE

METHOD KNN3: Creates a KNN method labeled KNN3 with SIMILARITY=ADJCOS

METHOD ENSEMBLE_KNN: Creates an ensemble method using KNN1, KNN2, and KNN03, with maxiter=100, maxfeval=5000.

METHOD SVD_als: Creates an SVD method with technique = als, fconv = 1e-3, gconv = 1e-3, maxiter = 100, seed = 2000, MAXFEVAL = 5000, function=L2 lamda = 0.2.

METHOD SVD_lbfgs: Creates an SVD method with technique =lbfgs, fconv = 1e-3 gconv = 1e-3 maxiter = 100 seed = 2000 MAXFEVAL = 5000 function=L2 lamda = 0.2 .

METHOD ENSEMBLE_SVD: Creates an ensemble method using SVD_als and SVD_lbfgs, with maxiter = 100 MAXFEVAL = 5000.

PREDICT: Creates two recommendations per user ID.

Performance Results

Table 1 shows the maximum elapsed time required to complete each METHOD statement. The PROC LASR create option is included to show the amount of time required to lift the data to a SAS LASR server before executing statements. All timings are collected from the SAS logs generated by each test scenario..

Table 1: RECOMMEND statement timings are shown in hh:mm:ss format. For shared SAS LASR server test scenarios, the table was created once and shared by all users.

	Private SAS® LASR™ servers		Shared SAS® LASR™ servers	
	One	Two	One	Two
CREATE RS	00:00:26	00:00:27	00:00:26	00:00:28
KNN ensemble	02:22:24	03:47:57	02:22:43	03:50:54
KNN1	00:19:18	00:30:20	00:19:14	00:31:35
KNN2	00:20:04	00:28:47	00:19:42	00:30:24
KNN3	00:19:57	00:27:06	00:19:39	00:30:55
Slopeone	00:01:59	00:03:07	00:02:02	00:03:07
SVD als	00:03:04	00:07:23	00:03:18	00:04:39
SVD ensemble	00:00:22	00:00:28	00:00:21	00:00:31
SVD lbfgs	00:08:50	00:15:39	00:09:22	00:12:57

Findings

The test results show that a four-node cluster can support multiple users simultaneously creating methods using PROC RECOMMEND. The elapsed time for each PROC RECOMMEND method increases as the number of users increase. The PROC RECOMMEND statements display a generally linear increase in run time as the number of concurrent jobs is increased from one to two users. For example, as shown in Table 1, using a shared SAS LASR server, the maximum elapsed time for KNN ensemble method increases from 2 hours 22 minutes for one user up to 3 hours 50 minutes for two users. A similar relationship exists for the private SAS LASR server.

Table 2: The elapsed time generating two recommendations per user ID for the following methods

Method	Elapsed Time
Slopeone	4:22:45
SVD als	0:21:43
SVD lbfgs	0:22:26
SVD ensemble	0:24:24

Figure 1: Shows the maximum active memory used per compute node for each PROC RECOMMEND statement. There were a total of three compute nodes used in the test environment. The memory usage was collected from NMON log files recorded during test execution for a single user.

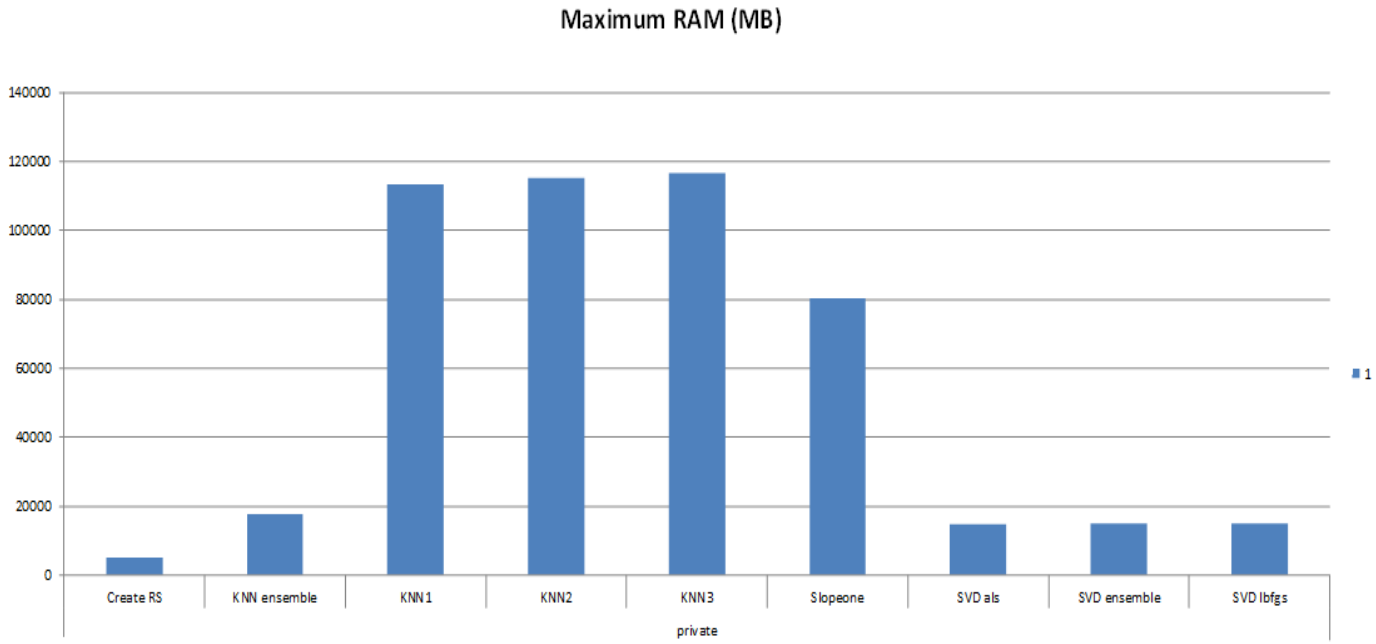

Table 3: Average CPU Utilization for Each Completed Scenario

Private SAS® LASR™ servers		Shared SAS® LASR™ servers	
One	Two	One	Two
37.2	50.9	38.2	47.6

To contact your local SAS office, please visit: sas.com/offices