Large-Scale Predictive Modeling with R and Apache Hive: from Modeling to Production

Alex Zolotovitski\(^1\)* and Yakov Keselman\(^1\)

1. Medio Systems Inc  www.medio.com
*Contact author: alex@zolot.us

**Keywords:** HPC, hive, hadoop streaming, deployment of R models in production

Some suggestions:

Recently, we have been building predictive models on large (e.g., 200M users generating 50B events) HDFS data sets related to user behavior. We have experimented with several existing frameworks (such as Apache Mahout) and \(R\) packages (see, e.g., those listed in \(Hadoop\) section of HPC task view on CRAN \([1]\)) that are able to build models based on full data sets. We have observed that while the final models were quite accurate, the time to iterate on model building and model validation was exceedingly high.

To significantly reduce the time it takes to build and validate a predictive model, we experimented with an alternative approach that uses local \(R\) on researcher’s laptop as a client for hive server pulling samples of data for fast and rich graphics, descriptive statistics etc. and for building nearly as accurate models on samples of the full data set. Moreover, the resulting \(R\) code, with minimal changes (e.g., redirecting graphical and diagnostic output, read data from the standard input, output results to standard output), is suitable for processing the full data set via \(Hadoop/Hive\) streaming (after ensuring that all data for the same user is chunked together and ordered by time) and \(R\)script. Hence, validation of the best locally-developed models on the full data set is fast as well.

In our presentation, we illustrate the full cycle of predictive modeling on large mobile gaming data sets. First, Hive is used for sampling (stratified, if necessary) of user data and for defining simple cumulative attributes on the data. Second, \(R\) is used for fast iterative predictive model building and validation on the sample and for defining additional attributes, if needed. Next, the resulting model is executed on the full data set by streaming user data (one user at a time) through the \(R\) scripts that were produced during the modeling step. Finally, performance of the model is monitored by executing additional \(R\) scripts on a Hive sample of the full data set. Our experience shows that such arrangement takes full advantage of both tools, resulting in accurate models that scale to hundreds millions of users.

**References**