Putting it all together: Creating a Big Data Analytic Workflow with Spotfire

Authors: David Katz and Mike Alperin, TIBCO Data Science Team

In a previous blog, we showed how ultra-fast visualization of big data is achieved with in-memory, in-data source and on-demand data access and aggregation using out-of-the-box Spotfire data connectors. In addition to exploiting these native data source capabilities, Spotfire can also leverage in-datasource advanced analytics through its integration with distributed computing frameworks like MapReduce, SparkR, H2O and Fuzzy Logix. Figure 1 summarizes these capabilities.

Notes
1. MapReduce, Spark and H2O support distributed computing in Hadoop. Fuzzy Logix supports Teradata, Aster and Netezza
2. TERR can be deployed as the advanced analytics engine in Hadoop nodes driven by MapReduce or Spark. TERR can also be called on Teradata nodes.

Fig 1 Big Data Architecture

One of the strengths of Spotfire is the ability to integrate these different data access and aggregation capabilities to facilitate a single analytic workflow with seamless visualization as a unifying force. In this blog, we will walk through a practical example built in a Spotfire template. It leverages a data connector as well as a distributed computing framework and mature R packages that utilize advanced analytics algorithms not available through the connector. R packages can be run in Spotfire with the embedded high performance Tibco Enterprise Runtime R (TERR) instance. Everything is integrated with point-and-click visualizations and interfaces so that the analyst works in a low friction environment to focus on solving the business problem at hand.
Let’s look at a manufacturing data set. Our task is to find the causes of a rise in the failure rate or yield loss. The items being manufactured go through a number of processes in varying sequences. At the end they are tested, in batches of different sizes, for adequate functionality. When yield loss goes up, we need to find out why – and fast!

The analysis proceeds in four steps:

1. Visualization of batch summary trend using in-datasource Data Connector capability to select a date range with a concentration of high yield loss batches
2. Identification of the most important yield loss predictors using the Distributed Computing framework
3. Importing row-level detail data into Spotfire for date range and predictors of interest
4. In-memory machine learning analysis of imported data to visualize detailed effects of predictors and their interactions on yield loss

**Summary Aggregation with Data Connector**

First, we select the data for the analysis. The left side of Figure 2 shows the interface for choosing the schema, date column and yield loss class. The data source can be any Hadoop or structured big data source that can be accessed via SQL.

Once the data is selected, the **Refresh Plots** button initiates a process in the data source that creates the trend and funnel plots shown on the right side of Figure 2. The plots show aggregates for each batch, which are quickly computed in the underlying database, via the in-datasource data connector capability. The connector then returns only the summary data needed to generate the plots. These calculations are
fast because they are computed in parallel, close to where the big data resides; only the much smaller results travel over the network back into Spotfire.

We have shown the data in two visualizations. The top visualization, known as a funnel plot, highlights outlier batches with excessive yield loss significantly greater than the general population. Spotfire combines the aggregated data points, provided by the connector, with the outlier limit line generated locally using the TERR (TIBCO Enterprise Runtime for R) engine embedded in Spotfire.

The lower plot of figure 2 is a time series view of the same data. Here, Spotfire has again helpfully colored the real outliers in red. We see that most of the groups with higher yield loss are clustered together. We have marked data in a time period which has a concentration of significant activity. Marking the summary data here will configure a future to step to use the corresponding detail data in the analysis.

**Identification of most important Predictors with Distributed Computing Framework**

The next step is to determine which potential predictors have the greatest power to explain the yield loss we are seeing. This is done using the distributed computing framework to run a random forest advanced analytics algorithm in the datasource. It is not uncommon for there to be hundreds or even thousands of potential predictors. The predictors may be measurements made on the product during processing, readings from sensors on processing equipment or attributes of the equipment. There may simply be too many predictors and too much data to bring it all back into the Spotfire client memory for rapid analysis. Hence the need to run this ‘variable reduction’ analysis in-datasource.

Figure 3 shows the interface for running the random forest in-datasource analysis. The first button initiates a tool provided by H2O for random forest regression. This is done in the datasource using a stored procedure. This tool distributes the random forest computations across all the nodes in the datasource and runs them there in parallel to improve performance. In addition, working in-datasource, it is not constrained by available client memory. This analysis creates a model representation in the database, as well as returning the predictor importance measures into Spotfire. Results are shown in the middle panel of Figure 3. The most significant predictors are shown marked by the user for configuring the next step of the analysis.
Importing detail data into Spotfire
Next, we want to import the relevant part of the data for analysis in-memory. The random forest analysis has provided a subset of predictor columns that are of interest, and the time series trend has identified the rows (time periods) of interest. This reduces the data in size by many orders of magnitude so that it can fit into our client memory space and be subjected to additional analysis. The **Import Selected Data** button is used to import this row-level detailed data into Spotfire client memory with the on-demand data connector capability.

In-memory Machine Learning analysis of imported data
Finally, we use the data subset imported into Spotfire client memory and run an ensemble-tree machine learning algorithm using the high performance runtime for R (TERR) instance embedded in the Spotfire client. The R package gbm (gradient boosting machine) supplies the machine learning algorithm, similar to the random forest algorithm we used in-datasource. This is a mature package with many options for tuning the model and interpreting the results. We focus all of Spotfire’s in-memory advanced analytics capabilities on the problem in this final step of the analysis.
Figure 4 shows the simple interface the business analyst uses to configure, run and evaluate the model.

Figure 5 shows the results of the analysis. The bar chart in the top left corner shows the predictor importance ranked by the impact each has on yield loss. Marking a predictor bar in this chart produces a drill-down chart below it showing the detailed effect of that predictor, across its full range, on yield loss. The table next to the bar chart shows the most significant 2-way interactions between individual predictors. Marking a row in this table produces a detailed drill-down Heatmap of the interaction. Our customers find that they can identify important signals and relationships with this analysis that they have not been able to detect using other methods.
Our goal in Spotfire is to provide the business end-user an intuitive, visual, easy-to-use interface to access the full spectrum of big data technology. In this blog, we have shown how a data connector - for fast visualization, a distributed computing framework - for in-datasource advanced analytics, and TERR - for in-memory advanced analytics, are all integrated into a reusable analytic workflow that any business user can navigate simply by interacting with visualizations. This particular example involved understanding manufacturing yield loss. Using this same technology, we have deployed many other analytic workflows such as: micro-segmentation of markets to predict customer behavior, fraud detection and risk assessment, price optimization, predicting performance of equipment and processes with sensor data and optimization of transportation routes and schedules.