Paper 1648-2018

How Statistically Analyzing System Behaviors with SAS® Visual Analytics Revealed Unknown Data Issues

Jessica Fraley, University of North Carolina at Chapel Hill

ABSTRACT

Automatic loading, tracking, and analysis of data readiness in SAS® Visual Analytics is easy when you combine SAS® Data Integration Studio with the DATASET and LASR procedures. This paper is a follow-up to a previous paper presented at SAS Global Forum 2017 on the methodology used at the University of North Carolina at Chapel Hill to track data preparation and readiness with SAS Visual Analytics reporting. This paper covers a real-world example of how analysis and visualization methods surfaced unknown data integrity issues brought about by anomalous system behaviors. This paper also covers how we recognized the issues, and created SAS Visual Analytics visualizations to help any SAS® customer quickly identify potential data integrity issues that originate from system behaviors.

INTRODUCTION

Data is constantly changing. Not only does it grow, the form that it takes morphs as new columns are added to existing tables, new tables are added to warehouses or data lakes, and existing data is repurposed or analyzed in new ways. As data changes over time, the processes that create and modify this data become larger and more complex.

To aid in understanding both our system and our data, the Enterprise Reporting and Departmental Systems (ERDS) department of the University of North Carolina at Chapel Hill created reports that specifically analyze how our ETLs and our data intersect. These reports use both raw data reporting and Univariate statistics such as Standard Deviation and Variance to longitudinally analyze both our system and our data.

The results of these analyses are then presented in a series of graphs within SAS® Visual Analytics. These are not reports that we look at every day. The reports are as much about longitudinal knowledge as they are about what the system is doing right now.

We will not be covering the specifics of our reporting methodologies in this paper as our methodologies were covered in a paper presented at SAS Global Forum 2017 which is referenced in the References section below. The purpose of this paper is to serve as a real world case study covering 1) how a growing and unknown HDFS issue was surfaced, and 2) how having this type of analysis in place allowed us to detect the issue prior to any symptoms being reported by either standard system monitoring or our customers.

HOW WE MEASURE AND DISPLAY OUR DATA PROCESSES

The ERDS department gathers data to create 2 different types of reporting

1. Operational reports – reports designed to display the current state of the data and system
2. Projective reports – reports that show us how both the data and our system's behavior changes over time and are likely to change in the future.

SUMMARY OF THE REPORTS CREATED TO MONITOR OUR NIGHTLY JOBS AND DATA

This is a listing of the reports as described in our previous paper. We will only display images of the reports relative to the example data issue. The primary measurements are:

1. Line chart of the end times of the nightly ETL and LASR load jobs by date
2. Line chart of the growth of each the data table by date

3. Line chart of the cumulative data size (all data sources) by date as represented in Figure 1.
4. Line chart tracking the run times of the individual jobs by date
5. Line chart of the cumulative runtimes of all jobs by date
6. Dual Axis Bar-Line chart (as represented in Figure 2) showing Standard Deviation and Variance of each job's run time over a sliding 30 day period and plotted as a moving average.
7. Bubble chart of each job start time and duration by date

The two charts that helped identify the presenting issue were the "Line chart of the cumulative data size (all data sources) by date" (Figure 1), and the "Standard Deviation and Variance of each job's run time over a sliding 30 day period and plotted as a moving average in a Dual Axis Bar-Line chart" (Figure 2).

Figure 1. Line chart of the cumulative data size (all data sources) by date

As you will notice from Figure 1, our data has a uniform linear total growth rate.

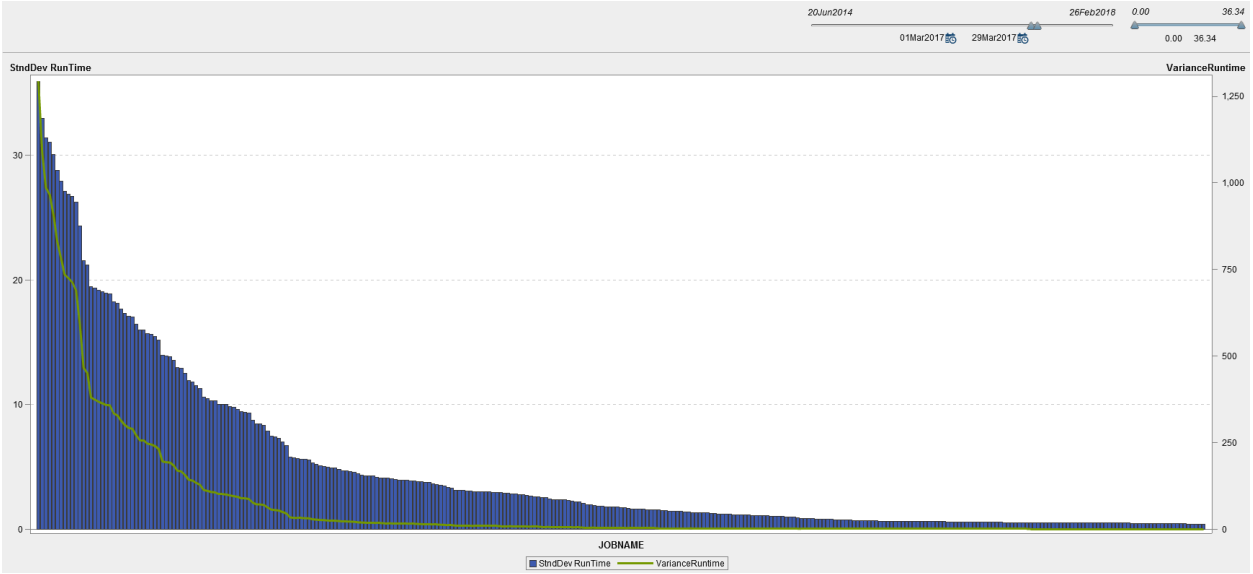

Figure 2. DUAL AXIS BAR-LINE plot of Standard Deviation and Variance of each job's run time over a 30 day period

In figure 2 we visually display the standard deviation and variance of each ETL, DataQuery and load job runtime over a moving 30 day period. Taken together, these measures provide a look at the variability of job runtimes. Jobs with high runtime variability are harder to predict as far as how long they will take to complete, but more importantly if a job's runtime variability changes appreciably over time then something has happened to affect that job. This knowledge is key to managing runtimes.

Additionally, within the Standard Deviation and Variance chart, the user can highlight specific jobs (Figure 3). This gives us the ability to identify specific jobs based on their variability.

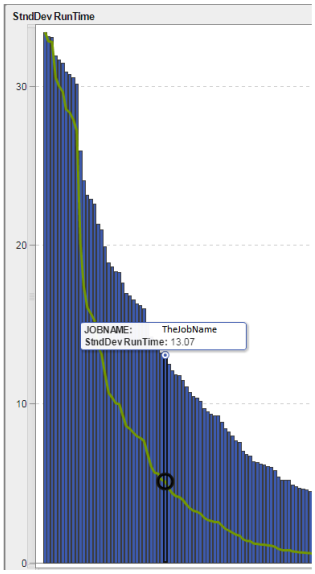

Figure 3. Standard Deviation and Variance chart displaying individual job data.

The ability to view job runtime variability changes can serve as an early warning system for data processing problems that are just beginning but have not grown to the point of providing behavioral evidence.

HOW THESE REPORTS SURFACED UNKNOWN DATA ISSUES RELATED TO OUR DATA PROCESSING

THE SYMPTOMS

To reiterate, when these reports revealed a developing issue neither the system nor the data was presenting visible behaviors which would indicate an anomaly. We had not heard feedback from our customers indicating an issue existed. From our perspective and from our customer's perspective, the data and system were acting normally.

I realize it is rude to reveal the ending before it arrives, but in this case it's worth revealing. The core issue at the heart of our case study resided within HDFS. The HDFS issue resulted in partial data loads. Most of the data would be successfully loaded into HDFS each morning, but a small amount would not be loaded. At the next day's load the previous missing data would be loaded and other data may or may not be loaded. No error messages were thrown while this was occurring. The solution was a patch for HDFS.

THE REPORTS

The runtime variability in the Standard Deviation and Variance report gave us a clue that something was not right. For some unexplained reason we saw an increase in runtime variance as displayed in Figure 4.

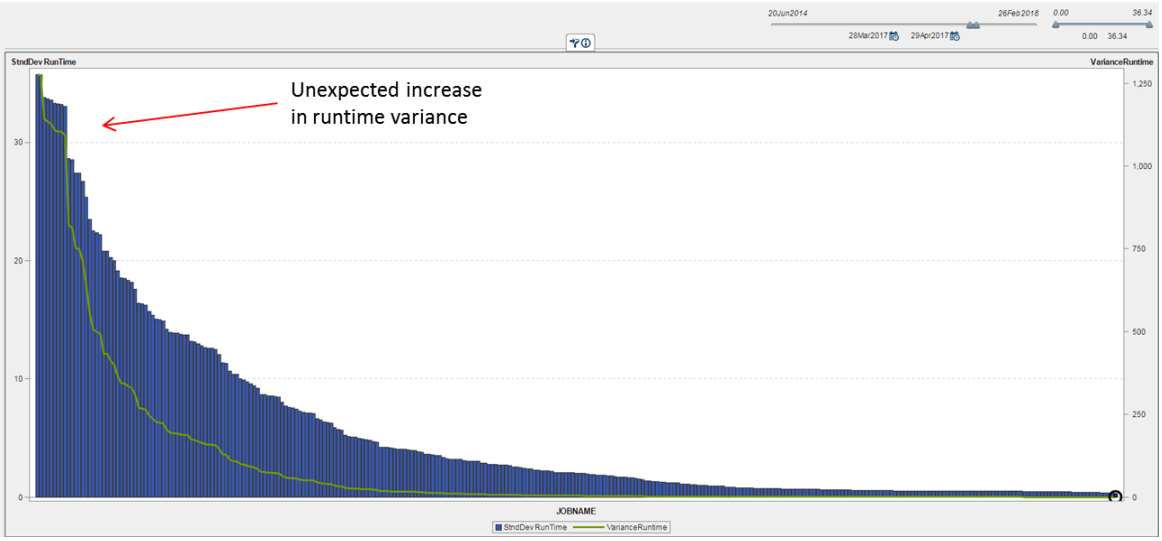

Figure 4. Standard Deviation and Variance of each job's run time over a 30 day period during which a problem was developing.

In the side by side comparison of Figure 2 and Figure 4 (see Figure 5 below), there is an obvious significant increase in the area under the standard deviation curve as well as the area under the variance curve. The area under the curve represents a cumulative measure of the variability of the run times for all of the jobs. As the red arrow points out, some of the increases were quite large.

This was somewhat puzzling as all of the nightly jobs were completing in the allotted time and no issues were being reported within the data. Something had changed which caused an increase in the number of jobs with large runtime standard deviations.

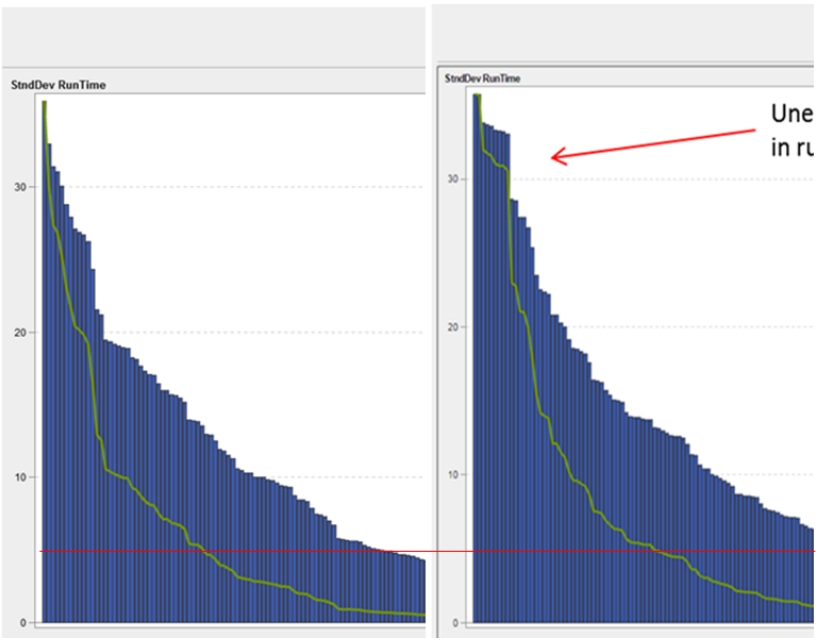

Figure 5. Side by side comparison of Standard Deviation and Variance of job runtimes before and after the onset of the anomalous behavior.

This sudden increase in job variability led us to explore the other reports. All of them appeared normal except for the chart representing our data growth when plotted over time – Figure 5.

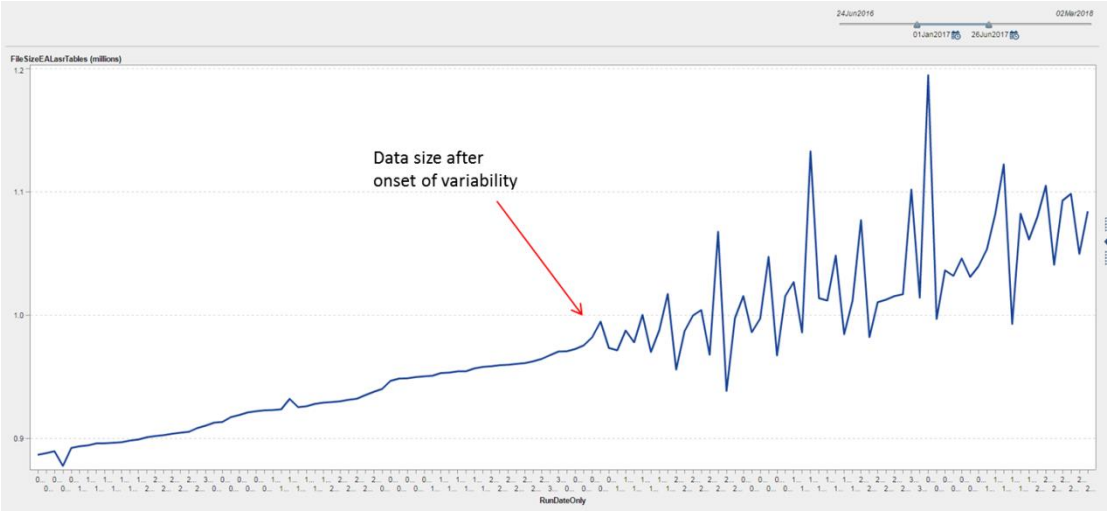

Figure 6. Longitudinal plot of cumulative data growth in MB showing increased data size variability

Figure 6 demonstrates cumulative data size growth rate by date. Prior to April 1, 2017 the data had a smooth linear growth rate. Around April 1st something changed and the size of the data being loaded into LASR began changing drastically.

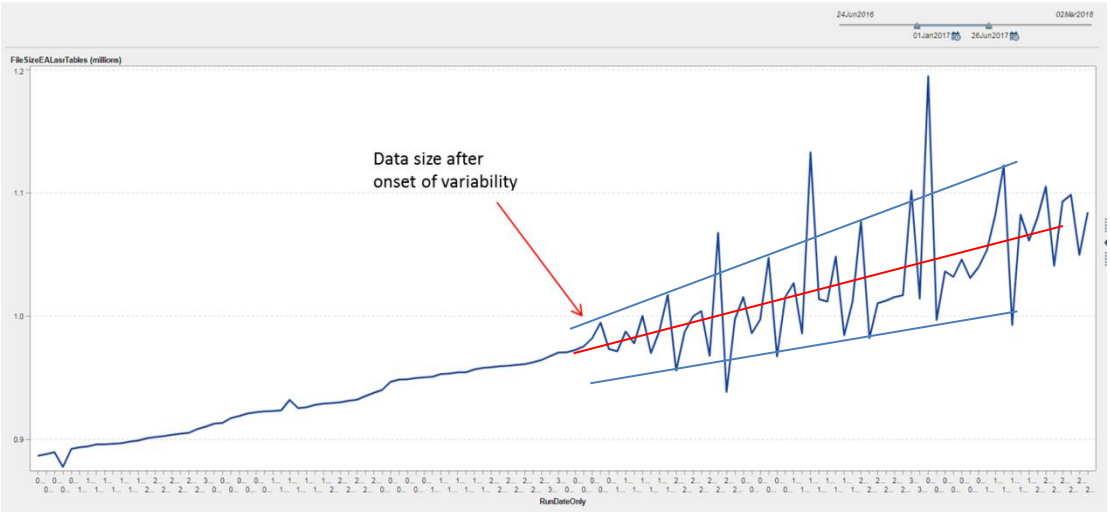

Figure 7. Longitudinal plot of cumulative data growth in MB showing increased data size

variability with channel lines.

In Figure 7 it's pretty easy to visualize the growth line continuing into and through the suddenly noisy data of the daily data size (red line). However the spiking behavior and changes in size above and below the projected data growth line not only indicate an issue, they also indicate the nature of the issue. If you were to draw a line which continues the linear growth rate, the swing in data size is approximately equal on both sides of that line. Also, with the exception of a few outliers the peaks of the data size swings appear to be growing in size (as represented by the blue channel lines). It is fairly easy to see that the presenting issue is growing over time as the channel lines continue to widen.

RESULTS

Both charts provided essential information in identifying HDFS as the source of the issue in that we knew specifically which jobs were having increased runtime variability as well as providing a measure of the issue's effects. More importantly, it let us know that an issue was developing before it became a serious problem.

This reporting also allowed us to eliminate many potential causes at the start. For instance by comparing the data size of the loaded data to the source data on extreme variability days, we were able to eliminate source data as an issue. This saved the work that it would have taken to set up manual testing of individual components within the jobs and data. In the end, the answer was a patch to the HDFS system. It turned out the system itself was affecting the data. But the need for a patch would never have been surfaced if we had simply measured data output, or individual process timings, or consumed system resources as these provide only a partial picture of what our system is doing.

As you will see in Figure 8, the variability of our job's runtimes returned to normal after the patch. It's important to note that Figure 8 is not truly valid for comparison against Figure 4, as during the timeframe of this case study, our ETL programmers were also working on efficiency changes for their code which resulted in a smaller output dataset.

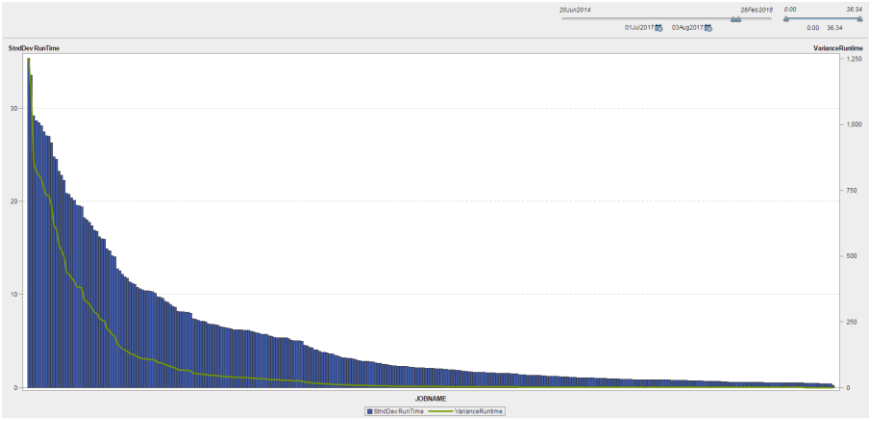

Figure 8. Standard Deviation and Variance of each job's run time over a 30 day period following the application of an HDFS patch.

Figure 9 shows the reduction in data size as accomplished by our ETL developers, but also the elimination of the extreme volatility and a return to a linear data growth rate as a result of the HDFS patch.

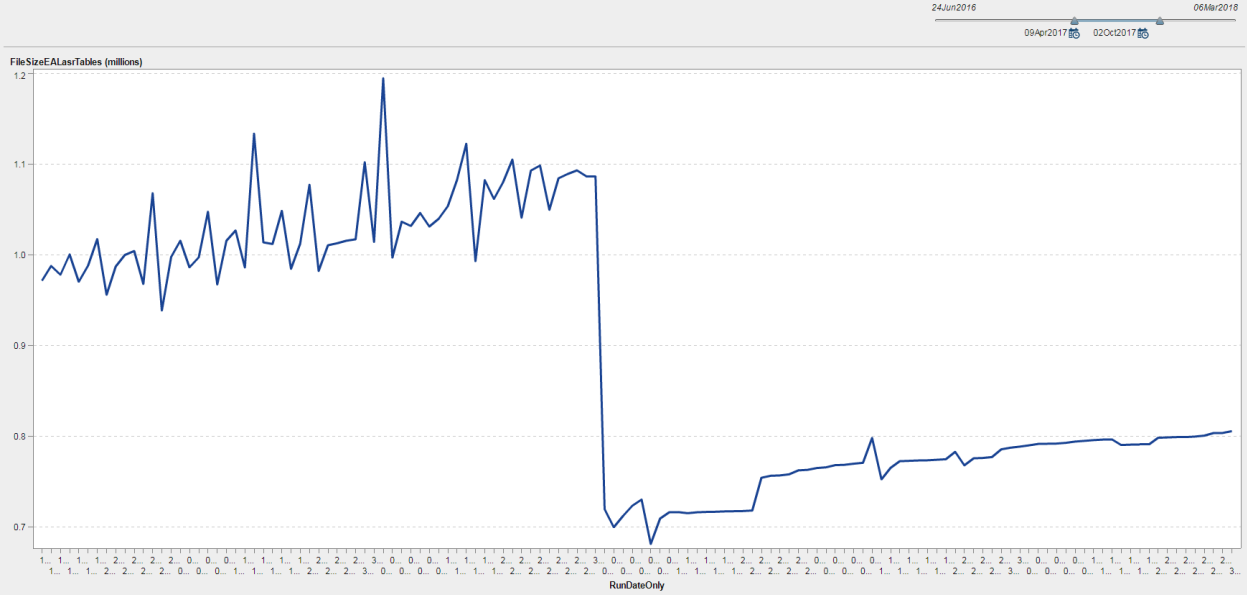

Figure 9. Longitudinal plot of cumulative data growth in MB showing the decreased data size volatility post HDFS patch.

CONCLUSION

As an occurrence, the facts of this case study are of little interest. What is of primary importance is that this HDFS issue serves as an excellent example of how analysis that encompasses both the data and the system allowed us to view hidden interactions between the data and the system. It also allowed us to understand our data at a deeper level than normal monitoring would have ever allowed.

Having these reporting capabilities allowed us to spot the issue as it was developing versus waiting until a customer reported data issues or system performance impacts, giving us the ability to be proactive in our approach and act as good shepherds of our data.

REFERENCES

Fraley, Jessica. 2017. “A Custom Method To Auto-Load SAS LASR Tables and Longitudinally Report on ETL, DQ, and LASR Timings.” *Proceedings of the SAS Global Forum 2017 Conference*. Available at http://support.sas.com/resources/papers/proceedings17/0898-2017.pdf

ACKNOWLEDGMENTS

I would like to acknowledge and thank my fellow team mates for all of their help and contributions on this, and many other projects: Prakash Balakrishnan, Maribel Carrion, Ryan Fulcher, Dean Huff, Keith Jones, Dan Kelo, Sally Lakomiak, Bob Poliachik, Efrain Santiago and Rachel Serrano.

I would especially like to thank Rachel Serrano and Mimi Bennett for being excellent editors.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Jessica Fraley
University of North Carolina at Chapel Hill
ITS Manning, Suite 2800 CB #2808

211 Manning Drive
Chapel Hill, NC 27599
Jessica_Fraley@unc.edu