SAS Analytical Suite

Jonathan W. Crawford

4/18/2019

jwc17@my.fsu.edu

https://youtu.be/JTtnOA8npzU

Abstract

As the digital revolution expands into every facet of our connected lives more and more data will become available to companies, governments, and researchers. Making use of this data has become the new frontier in computer science and delivering a tool to harness that power is of paramount importance. One such company at the forefront of innovation in this market is SAS, based in Raleigh, North Carolina. This paper aims to explore the history, capabilities, and validity of SAS as a tool for data analysts of varying skill levels to provide valuable insight into large datasets that house enormous potential. SAS is a powerful tool, one that is limited only by the utility in which it is applied, and the creativity of the user.

Keywords: Statistical Analysis Software, Business Intelligence, Graphical User Interface

Contents

SAS Analytical Suite

If it were possible to accurately predict future events, in a situation of life or death, or perhaps where precision carried the utmost financial implications. The ability to base that decision on the largest amount of detail possible and the highest level of mathematical certainty increases the confidence in those findings exponentially. That is the objective of predictive modeling and Big Data research. SAS Analytical Suite for statistical analysis provides the capability to effectively categorize, calculate, and present those findings in an understandable way. This approach of unifying operations under one program, eases adoption and creates value to the end user in terms of economical simplicity.

Birth of SAS

Like many states North Carolina has several universities, with one of them bearing a higher concentration of agricultural research than the others as a land grant university. That institution is North Carolina State University in the state capital of Raleigh. In the 1960's massive amounts of agricultural data flowed into the research departments of this university due to the nature of it possessing the most powerful mainframe computer available within the state at that time. The halls of academia are rife with unbounded creativity and collaboration free of restrictive financial interests and the climate at NC State in this era was no different.

There at NC State in 1966, SAS was imagined as a way to harness this agricultural data, via a graduate team led by two faculty members of the university named Anthony Barr, and James Goodnight. The language was based on the original PL/I IBM procedural language of the day. Over the course of the next decade they would develop an albeit rudimentary tool for statistical analysis, yet one the world had not realized how much it had needed. They outgrew the university by 1976, as Goodnight (2014) himself notes in an interview with Forbes,

> Of course, the university, at the time, was not supposed to be a business anyway so they thought it was a good idea that we move off-campus as well – so we moved across the street and that's how SAS was founded. It had a long development history at NC State before we left – I think we left with about 300,000 lines of code; it's probably 10 million lines of code now. (para. 7)

SAS then became an actual company and not just a research project. Over the subsequent decades and through software iterations it has maintained a powerful market share in the analytics industry. Fast forward to today and SAS fourth generation is utilized by every major industry for implementations as varied as human genome health research in clinical studies to population or revenue forecasting by governments and businesses like American Express.

The SAS Advantage

SAS is a fourth-generation language, which means it provides a sufficient amount of abstraction to allow for support of database manipulation, mathematical functions, a GUI, and report generation exporting. This means it has a much more refined and specific purpose than third-generation general purpose languages like C++, Python, or Java. Developers often decry tools of specificity due to their restrictive nature, but SAS provides functionality that translates to value for casual BI users and highly skilled computer science professionals alike.

Front and Center

The reason that SAS has been able to achieve such success in the BI data science segment is that it utilizes built-in macros for simple tasks like loading data and richly detailed graphing functions. There are stored libraries of mathematical formulas that can be applied to unstructured data quicker and more effectively than reinventing the wheel. This is where it sees extensive use from BI professionals as a huge step up from providing statistical representations in Excel. This

is important because this segment often includes analysts whose primary purpose is in areas other than programming, like doctors, scientists, or executives. Yet despite placing significant emphasis on graphics SAS remains speedy and powerful, as Jalanila (2016) points out,

> "while SAS® Text Miner has better ease of use with point and click functionality, Python was able to process the algorithm as fast as SAS® Text Miner. SAS® Text Miner has better visualizations which help the users to better interpret the results where as in Python and R, the user has to code manually" (para.10)

Same speed, same result, but with better graphics, and much easier data input and manipulation, this is the essence of the SAS advantage over counterparts.

Methods

SAS programmers and analysts have access to an entire suite of built-in statistical methods and these functions are easy to call with simple naming once data has been loaded. This includes frequencies, regressions, correlations, deviations, and everywhere in between. Calling these functions is as simple as "proc ttest data = brands;" for a paired t-test with an optionally customized graph showing the correlation between brand names for some type of data entered. This is powerful in that it does not require mathematical knowledge to employ, yet any amount of additional customization commands can be typed by a user depending on their skill set. These functions are what makes SAS usable "out of the box" unlike other data science offerings.

```
title 'Distribution of Mileage';
proc sgplot data=sashelp.cars(where=(type ne 'Hybrid'));
  histogram mpg_city;
  density mpg_city / lineattrs=(pattern=solid);
  density mpg_city / type=kernel lineattrs=(pattern=solid);
  keylegend / location=inside position=topright across=1;
  yaxis offsetmin=0 grid;
run;
```

Figure 1: SAS code example for a vehicle MPG Histogram with optional customizations

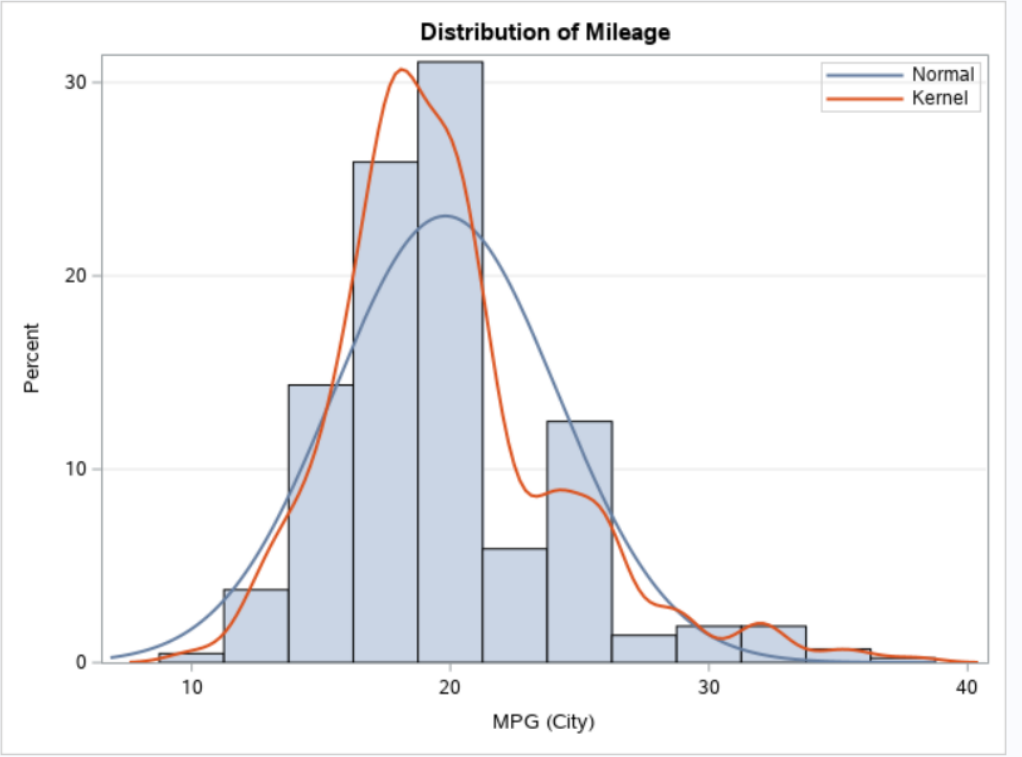

Figure 2: The resulting Histogram Plot from the example SAS code

Visualization

The fine-tuned graphing functions available in SAS provide the capability for analysts to aggregate data directly from a database, calculate, and author presentation quality reports under one roof. This is made possible with color customizable histograms, box plots, bar/pie charts, and scatterplots contained in the graphical library. This is a clear advantage for BI users in contrast to cobbling together programs, data, and services to create a usable analysis capable of being understood by an audience of potentially varying technical aptitude.

Underneath

While it is not difficult to envision SAS graphics gracing a fortune 500 boardroom filled with executives, SAS also invariably shines when placed in the hands of skilled developers. Developers with a strong knowledge of SQL can adeptly integrate custom data in dynamic fashion for evolving test runs. The SAS language is unrestrictive, statements can begin or end

anywhere, but they must end in a semicolon. Unlike other finicky languages SAS keywords are not case-sensitive. A loop in SAS is as simple as, “DO (conditions) (statements) END;”. While variables default to numerical values, they can be declared and referenced in a similar fashion to SQL, that is why SAS is often easier for SQL developers to adopt. However, the real meat of this language is the power developers have to program custom mathematical and data handling functions into the libraries that specifically pertain to their own data science pursuits.

```
data onewayanovadata;
   call streaminit(112358);
   drop n;
   do n = 1 to 100;
      treatment = rand( 'TABLE', .2, .4, .4);
      if treatment = 1 then response = rand( 'NORMAL', 10, 0.8 );
      else
      if treatment = 2 then response = rand( 'NORMAL', 11, 0.8 );
      else
      if treatment = 3 then response = rand( 'NORMAL', 15, 0.8 );
      output;
   end;
run;

proc print;
run;
```

Figure 3: An example of SAS Code with a loop and if/else Statements that produces an ANOVA Test

Once loaded and stored within the custom library, a programmer can then call their own functions seamlessly with the aforementioned built-in methods. Solving a problem with a method is the hallmark of what a computer scientist does, and this capability is what makes SAS a full-blown language using parenthetical notation for calls. Creating a custom SAS solution for their projects, and repeatable value for their organization. SAS also runs in virtualization, and therefore it is platform independent. The capability of SAS to blend the simple along with the complex in such a polished suite is what sets it apart from other software.

Counterpoint

It is important to note that SAS does not adhere to the open source model and that restricts availability in comparison to some of the other widely used contemporary data science tools like the R programming language or Python. SAS does provide a university edition to combat this problem, however it has not seen the type of following that open source languages enjoy. When a language is locked down in this way it means that only the people under the employ of that company hold rights to improve upon it. It could be pointed out that type of strategy limits the ability of the language to tap into a wider range of ideas for evolution. Ideas that might reside in other industries, be bound by geographical restrictions, or reside inside the halls of academia where this language ironically began.

In Conclusion

The SAS offering available today is a far cry from the days at NC State. It has reached a level of refinement that has seen wide spread adoption and a reputation for delivering results that data science professionals can trust to be accurate and repeatable. The gap between computer scientist and statistician is something SAS sought to remedy. As pointed out by Ryan in a presentation with Goodnight (1978) when SAS began to take hold,

> The impact of computer science on statistical computing, while important, has not been as great as it should. Part of this is caused by statisticians being unaware of relevant research in computer science. A related, and possibly more serious problem, is that computer scientists are not aware of some of the interesting problems in statistical computing. (para. 2)

It is clear that SAS has closed that gap immensely and current versions of SAS embody that vision. While some might object to the financial business model of the company, it promotes a

high standard of quality and “under one roof” mentality that allows it to retain a niche market in the analytics industry. An industry that will continue to see widespread growth due to new data sources and technological improvement of the ones currently available.

References

Images Courtesy of SAS University Edition, Created by Jonathan Crawford using installed example data sets and provided tutorial, 4/25/2019.

Edward A. Greenberg, Wm. Max Ivey, and Bruce R. Lewis. 1981. Comparison of some available packages for use in research data management. SIGSOC Bull. 12-13, 4-1 (May 1981), 1-8. DOI: http://dx.doi.org.proxy.lib.fsu.edu/10.1145/1015528.810922

G. Rex Bryce, James W. Frane, Thomas A. Ryan, Jr., James Goodnight. 1978. Statistics and computer science: Recent development in BMDP computing algorithms. In Proceedings of the 1978 annual conference - Volume 2 (ACM '78), Vol. 2. ACM, New York, NY, USA, 552-. DOI=http://dx.doi.org.proxy.lib.fsu.edu/10.1145/800178.810089

S. Snezana and M. Violeta, "Business intelligence tools for statistical data analysis," Proceedings of the ITI 2010, 32nd International Conference on Information Technology Interfaces, Cavtat, 2010, pp. 199-204. URL: http://ieeexplore.ieee.org.proxy.lib.fsu.edu/stamp/stamp.jsp?tp=&arnumber=5546396&isnumber=5546329

A. Jalanila and N. Subramanian, "Comparing SAS® Text Miner, Python, R: Analysis on Random Forest and SVM Models for Text Mining," 2016 IEEE International Conference on Healthcare Informatics (ICHI), Chicago, IL, 2016, pp. 316-316. doi: 10.1109/ICHI.2016.58 URL: http://ieeexplore.ieee.org.proxy.lib.fsu.edu/stamp/stamp.jsp?tp=&arnumber=7776374&isnumber=7776301

E. Slanjankic, H. Balta, A. Joldic, A. Cvitkovic, D. Heric and E. Veledar, "Data mining techniques and SAS as a tool for graphical presentation of principal components analysis

and disjoint cluster analysis results," 2009 XXII International Symposium on Information, Communication and Automation Technologies, Bosnia, 2009, pp. 1-5. doi: 10.1109/ICAT.2009.5348419 URL: http://ieeexplore.ieee.org.proxy.lib.fsu.edu/stamp/stamp.jsp?tp=&arnumber=5348419&isnumber=5348395

NA. (2017). How SAS Began, Retrieved 4 25, 2019, from SAS-Company Information: https://www.sas.com/en_us/company-information/profile.html#1966-1980

High, Peter (2014). An Interview With The Godfather Of Data Analytics, SAS's Jim Goodnight Retrieved 4 25 2019 from Forbes: https://www.forbes.com/sites/peterhigh/2014/05/12/an-interview-with-the-godfather-of-data-analytics-sass-jim-goodnight/#7f7173341ba8