An Exploration of Typing Tool Development Techniques

By

Kevin Knight, Digital Research Inc.

John Leggett, Digital Research Inc.

E-Mail: kevin.knight@digitalresearch.com or John.Leggett@digitalresearch.com

1. Introduction

The value of research is directly tied to the usability of the findings. This link is especially important when advanced analytical techniques are part of the research program. Unfortunately, analysts often focus too much on the details of producing the results – the statistical significance – and not enough on the detail of how business clients will use the results – the practical or business significance.

A microcosm of this larger industry challenge is the development of segmentation typing tools. While much attention is paid in the industry to the development of segmentations themselves, much less attention is paid to the segmentation tool. A simple review of online discussions regarding segmentation studies shows that we often end up discussing the very fine nuances of the analytical techniques to derive the segments, but rarely have even a broad discussion of how to best make these segments accessible to our clients.

We believe this is to the detriment of the industry. The typing tool is really the "engine" of a segmentation study. Without an effective typing tool, the segmentation can only be a static, point in time analysis. The tool keeps the segmentation alive and allows it to become a dynamic piece of a broad ongoing research program. The typing tool helps answer questions of how to reach people in pre-determined segments, and helps bring the segmentation alive and off the page. The segmentation cannot be applied going forward if the segmentation typing tool fails, and the client team, understandably, is likely to lose faith in the underlying segmentation solution.

In this paper we will discuss some of the challenges and trade-offs to some current typing tool development approaches. In particular we will discuss issues around sensitivity to individual attributes, multicollinearity, sample distribution considerations, scoring predictions and predictive group starting points. Each method has advantages and disadvantages. The industry standard, discriminant analysis, provides a simple and quick way to create these typing tools. Other methods like multivariate logits offer exacting fit to the sample universe, while Classification and Regression Trees (CART) allow for flexible data sets.

The overall goal of the paper is to offer researchers a broader view of typing tools and introduce some considerations that should be included in any typing tool development or evaluation. The cost of a typing tool failing is more than just the cost of the underlying segmentation study, as the ongoing implementation of segments within a corporate strategy can put whole business units on the line. Ultimately, poorly constructed typing tools put the trust in our industry at risk.

2. Discriminant analysis in developing typing tool algorithms

Discriminant analysis (DA) is among the most used multivariate tools in marketing research. It is a fairly flexible technique that can be applied to numerous business challenges, and can be easily run using any of the popular statistical software programs. As discriminant analysis can be used as the basis of a variety of marketing research analytic output, including perceptual mapping, key driver analysis, and verifying differences across known groups, many analysts have at least some level of comfort with the technique.

Using discriminant analysis to generate a typing tool is relatively straightforward. The simplest method requires the analyst to simply capture the Fisher coefficients, a standard output option for most statistical software. These coefficients, and the corresponding constant term, create a series of linear equations, one for each segment that is being typed. In order to use the tool, one simply solves each equation by multiplying the respondents' answers by the coefficients for each segment and adding the appropriate constant terms. Each respondent belongs to the segment for which the linear equation produces the highest score. As the math is straightforward, this is a relatively easy algorithm to implement, so long as all of the items in the typing tool are asked of all respondents.

Clearly, part of the appeal of using DA to generate typing tools is the ease with which they can be created. Discriminant Analysis is not overly computationally intensive, particularly given today's computer environment. Therefore, it is easy for the analyst to test variations of potential predictive variables quite easily. Most software will not only quickly provide the necessary coefficients to run the algorithm, but will also provide an instant snapshot of how well the model fits the data used to build it. With some additional effort, it is not difficult to also test how well the model fits holdout data. Certainly a great deal of the popularity of this method of creating typing tools is the ease with which analysts can create a tool, and even work through a series of tools, testing various alternatives to find the algorithm(s) that will best suit the particular needs of a project.

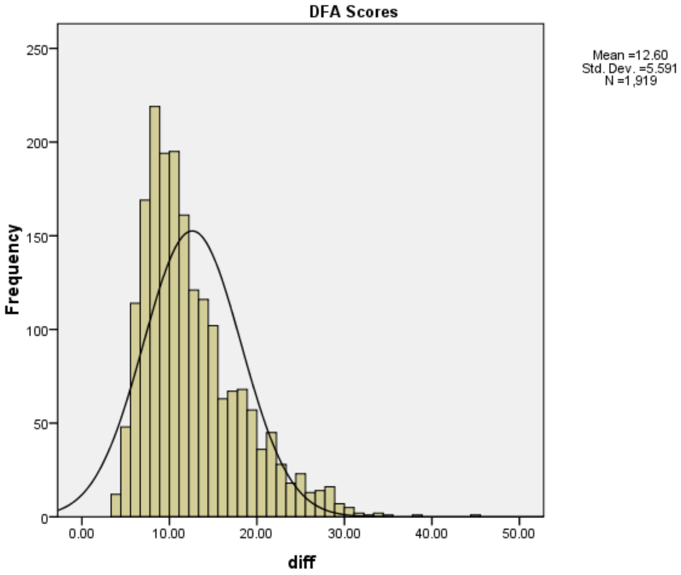
Furthermore, because many analysts are familiar with this technique, they are likely to understand the diagnostic information that is provided with the model output, and can therefore feel confident in understanding how well the model is working from a statistical standpoint. Most users are also familiar with the normality constraints, and understand that tools developed using DA will likely have a high level of predictive success when that assumption is met, and that the tool may not be quite as effective when that assumption is not met. Of course, all other considerations that would normally come into play for model specification with a parametric model should be considered, such as multicollinearity.

In addition to the assumption of normality, DA is sensitive to model specification. This sensitivity is not limited to the obvious need of selecting the appropriate combination of predictive variables. It is also important that the analyst understand which available options of the discriminant analysis to use. For example, in discriminant analysis the analyst can adjust the requirement to specify the prior probabilities to be used. The user can choose between setting all prior probabilities to be equal, or to be based on the group sizes from the data being used to generate the algorithm.

There are important implications for how the tool will perform, particularly out of sample, based on the prior probabilities selection. If there are especially small (or large) segments in the data being used, the analyst must understand if that will be reflective of the populations that will be typed in the future. If that will be the case, prior probabilities should probably be based on group sizes in the data. Otherwise, the tool will likely overfit those small segments and underfit the larger ones. On the other hand, if the sizes of the groups in the data being used to generate the tool are somehow artifacts of that data set, and will not necessarily be reflected in future populations that may be typed, perhaps the prior probabilities should be equal among segments. This is simply one example of an area where it is important to understand not only how discriminant analysis works, but also the segment data being used, and how that relates to future populations that will be typed using the tool. In other words, the analyst must understand the link between the underlying data, the resultant segmentation solution, and the ultimate business issues the segmentation is being used to solve.

Aside from model specification concerns, there are some systemic challenges that analysts need to monitor. In particular, DA has a tendency to yield models that do not provide a great degree of separation between the segments. One example, taken from actual segmentation data, is in Figure 1. This chart plots the differences between the highest and lowest scores from a typing tool developed using DA. What this shows is that even when looking at the difference of the algorithm scores between the *best* segment fit and the *worst* segment fit, those differences tend to be small. In fact, the distribution of those scores skews to the left, meaning that they tend to be closer than if they were normally distributed.

Figure 1



The cause of this can be seen by viewing the Fisher coefficients. Coefficients for each variable tend to move together across segments in terms of magnitude. When reviewing the coefficients from DA, coefficients are often larger for some variables, which therefore have a greater impact on the typing, and smaller for others. This is true across the equations, so that all coefficients for a particular variable or attribute have similar magnitude regardless of which segment is being scored.   As a result, while the responses to these questions will provide differentiation between segments, this differentiation is often more subtle than might be ideal.

One of the keys to evaluating the level of concern needed regarding these high magnitude coefficients is the size of the constant terms across segments. The attributes with larger coefficients will play a large role in offsetting the constant terms that are a part of the algorithm. If these constant terms are of large magnitude and/or are far apart from each other, the scores on the large coefficients might be crucial to offsetting the constant terms; low scores on these attributes might make it impossible to move assignment away from the segment with the highest constant term. This can be a useful feature of discriminant analysis, as certain attributes with larger coefficients will essentially be weighted more heavily in how they influence segment assignment, if there are indeed some attributes that are more significant drivers of segment assignment. However, this can sometimes be an issue, particularly if the attributes with high coefficients are not necessarily ones that are clear segment differentiators on their own, but are just statistical artifacts – another area of statistical significance versus practical or business significance. This marks yet another area that the analyst must be aware of the benefits and costs associated with this method in order to use it properly to create a tool that is best suited to the application at hand.

3. Classification and Regression Trees (CART)

Tree models are computationally intensive techniques that allow an analyst to determine how segmentation variables can be divided into subsets based on their relationship to a series of predictor variables. Classification trees involve a categorical dependent variable while regression trees involve a continuous dependent variable. The two approaches are similar enough that they tend to be referred to together as CART (classification and regression trees). While technically when used to create segmentation typing tools we will always be using classification trees, for convenience we will follow common practice of referring to these as CART models. Using CART models as a technique to develop typing tools is a popular approach that has some obvious appeal.

CART models are extremely flexible, and particularly effective in uncovering complex dependencies between predictor variables. CART models are adept at uncovering non-additive interactions among variables, as well as allowing for nonlinear relationships between variables. For these reasons, particularly when identifying a complex model specification, CART can be used to identify an algorithm that will match the learning data set with very strong accuracy. Importantly, CART is not overly sensitive to violation of basic data assumptions, such as normality, and is in an extremely flexible and customizable method.

A particularly nice feature of algorithms developed using CART is that the tools can be adaptive. While tools using the coefficient scoring method require responses from every question before segmentation membership can be calculated, CART allows for algorithms that adapt to ask only the questions required to reach a terminal node. Using the example shown in Figure 2 (below), if a respondent is first asked question B1_2 and responds with a value less than 3.5 that respondent can next be shown question B2_15 (those answering greater than 3.5 would get question B1_38). If the respondent answered B2_15 with a response less than 2.5, they could immediately be placed into Segment 7, with no additional questions asked. This approach can be useful, particularly when issuing a computer based survey (online or CATI), as it can help shorten survey length (it can also be applied with a paper-based survey, with explicit branching instructions embedded in the survey).

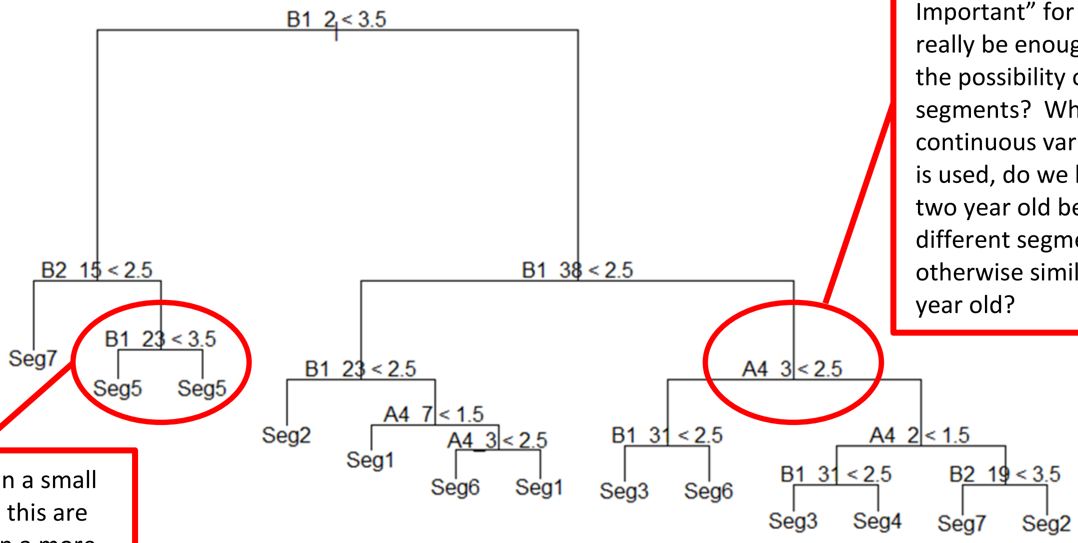
One of the potential risks when using CART is the model's tendency to overfit the relationship. This can be particularly problematic when the sample being used to create the algorithm is not representative of the population being modeled, or not applicable to other subsets that may be analyzed by the typing tool in the future. This potential problem is often exacerbated when the underlying model is not overly complex (i.e. there are simple or no interactions, there are only a few variables, and the relationship are largely linear). In this case, CART only returns approximations of the actual relationship, resulting in a bumpy as opposed to smooth fit, and will often use too many parameters.

There are other potential disadvantages with using CART that should also be noted. CART can be particularly time intensive, both in developing the tool, but also in creating the user interface for the tool. Creating the tool is necessarily an iterative process, and requires a great deal of exploring various options as well as potential model specifications. Given the complexity of CART models, they can be more "black box" in nature than some of the other methods of creating algorithms, the diagnostic phase can be more difficult, as the impact on the model of individual variables is more complex and more difficult to evaluate than other techniques. When CART is at its best, creating algorithms involving particularly complex relationship between a high number of predictor variables, is when this task is most daunting, particularly as there are likely to be a high number of terminal nodes, let alone all of the nodes passed along the way to segment assignment.

One area that this can be of concern is that the way a tree operates (each node represents a split based on the response to one question that may or may not exclude a respondent from one or more segment designations) can mean that small changes in inputs, if they occur at a cut point, could result in completely different segment assignments. It is important to consider whether this is an acceptable part of the algorithm, particularly when using rating scales that may include some grey area. For example if a cut point occurs between scale choices of "Very Important" and "Somewhat Important", do we really believe that this distinction on one question is enough to eliminate the possibility of a respondent being assigned to one segment? In a coefficient-based algorithm, responses increase (or decrease) a respondents' likelihood continuously depending on where they fall on the scale for each question used in the tool. The danger in coefficient-based tools is that they treat the scale as being consistent across responses (unless terms are raised to powers in the model specification, which allow for non-linear fits), which is not ideal, but with CART there can be situations where answering "Somewhat Important" has the same impact as answering "Not At All Important". If some of these distinctions happen to be artifacts of the sample being used to develop the tool, this can have a deleterious effect on the ability of a tool to work with independent data sets. This can be exacerbated when continuous variables are used as predictive variables. A classic example would be if age is used. A CART algorithm necessarily uses cut points to determine which branch to follow, but is it really reasonable in many cases to find a particular age where one year of difference should really impact segment assignment? There may be some instances where this is acceptable, and certainly age is often broadly categorical, but most of the time this will force a false level of precision in the tool. More problematic, depending on where an age cut occurs, it may render the tool useless to run on a subset of the sample defined by age, such as research on Millennials or Baby Boomers.

Again considering the quality control issue with CART, it is easy to spot anomalies on a simple tree like that shown in Figure 2 (below). However, for a tree with hundreds of nodes (or more) it is extremely difficult for the analyst to be able to have a complete understanding of where there may be issues with the algorithm. As a result, the analyst will be largely at the mercy of broad diagnostics that are part of the model, and will be hard-pressed to truly understand the ins and outs of how the model performs without spending a great deal of time reviewing complicated tree diagrams or testing scenarios.

Figure 2



In a 6 point scale, will the difference between "Very Important" and "Somewhat Important" for one question really be enough to eliminate the possibility of certain segments? What if a continuous variable like age is used, do we believe a forty-two year old belongs in a different segment than an otherwise similar forty-three year old?

While easy to spot in a small tree, anomalies like this are harder to discover in a more complex tree with many nodes.

4. Linear probability model(s)

Linear probability models are sometimes critiqued because they can yield results that are not realistic. A linear probability model is essentially a standard OLS regression that is run for a dependent variable that is a binary choice term (often coded as zero or one). In theory the results of these models could be interpreted as a way to measure the likelihood or probability of a string of characteristics yielding a value of one for the dependent variable. At issue with these models is they can produce probabilities of greater than one (greater than certainty) or less than 0 (less than no chance). For that reason, logit and probit distributions are used to create models that smooth results along the tails to correct for the possibility of unrealistic probabilities. For the purposes of creating a segmentation tool however, this need not be an issue that an analyst should be concerned with.

For segmentation with *n* segments, it is relatively straightforward to run *n* linear probability models given the ease of running a series of OLS regressions with modern statistical software. The output of these models, constant terms as well as coefficients, can be used in the same manner as the Fisher coefficients from discriminant analysis, as a series of linear equations. In this case, as the outcomes are ultimately applied as relative measures of how well a respondent fits with each segment it is not really an issue if scores are higher than one or lower than zero. This simply identifies really strong, or really weak fits. It is possible to simply take the scores as they are output from the series of linear equations and assign the segment that is associated with the strongest fit, as is commonly done with the output from discriminant analysis. It is also straightforward at this point to normalize the scores to a particular data set. This can be useful for at least two reasons. Importantly, it provides a measure of how well each respondent fits into each segment in terms that are easy to understand (i.e. greater than 1.96 or roughly two standard deviations from the mean tells us that we have a very strong fit). This is also helpful in the event that one of the equations tends to yield particularly high or low scores relative to the others. Again, a good analyst should give some thought to how this algorithm should ultimately be run and applied in order to fit the needs of the project, in some cases using the raw scores may be more appropriate, while in others standardizing the scores may be essential.

Use of linear probability models tends to solve one of the issues that appear prevalent when using discriminant analysis. Using raw scores or normalized scores, the spread between lowest and highest scores follows much closer to a normal distribution using these methods. This can be seen in Figure 3 and Figure 4, showing a typical distribution of differences for each method of scoring. Likewise, the coefficients tend not to move together in magnitude in the same way the Fisher coefficients do. This suggests there can be some advantages in terms of getting clearer separation between segments from this method.

Figure 3

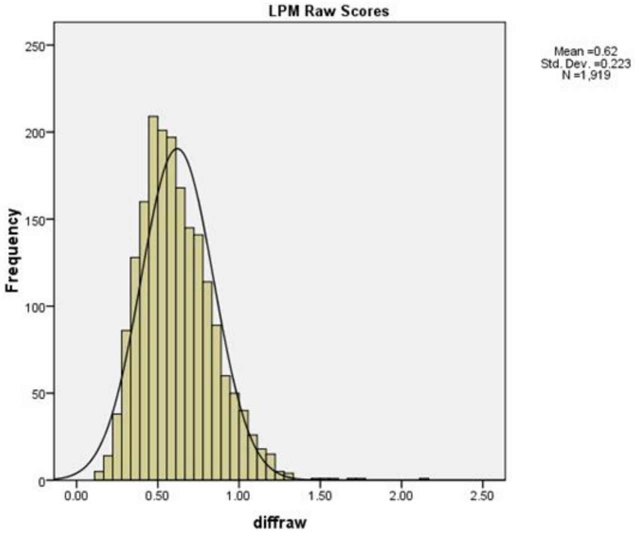
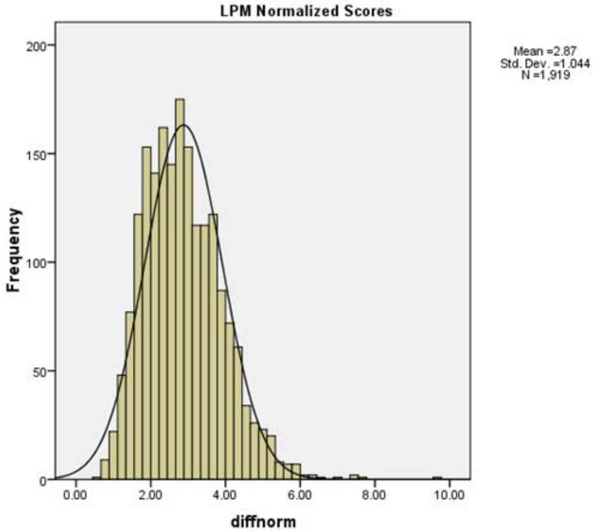


Figure 4



One issue with using this method is the greater difficulty in creating and testing the algorithms. Many other methods can very simply yield results for all segments at once, and often include diagnostics about in-sample fit as part of the standard output. Running a series of linear probability models can be much more intensive however, as separate output is required for each segment. That output then must be combined and applied to run the algorithm, and even in-sample fit measures need to be calculated manually. Of course, for most software this can be automated to some extent through skillful programming, but even if this is set up, it is clearly more labor intensive than some other methods. This in turn can make it more time consuming to run various iterations in order to find the optimal model specification. For every iteration of the tool the analyst must re-run all the models and combine the new output in order to test the results. This can be a burdensome process. On the other hand, this requirement of a more hands-on approach can force the analyst to pay closer attention to all details of the algorithm as it is being created.

5. Technique Independent Considerations

Regardless of the underlying statistical process used, when putting together typing tools for segmentations, analysts need to consider a number of factors. Foremost, it is important that the analyst always keep in mind the ultimate needs of the client in putting together a typing tool. If possible, the typing tool should be developed in concert with the development of the segmentation. Of course, the tool should be as accurate as possible in the statistical sense, but it should also satisfy the more qualitative requirements of non-analyst scrutiny. In other words, be as accurate as possible in the business sense, not just the statistical sense.

It is critical that the analyst always keep the client needs in mind when developing a typing tool and not get trapped in the beauty of the techniques. The ultimate needs of the client often will place constraints on the tool that must be considered during the development phase. If the tool is to be used often for qualitative recruitment, or is to be included in short quantitative surveys, the analyst should be aware that there may be a need to compromise accuracy to some extent for brevity.

In some cases, typing tools will need to be developed that will also be able to type existing customers in a CRM database. In these cases the tool must be developed using predictor variables that are currently maintained, or easy to acquire. Finally, if the tool will be used to type various subgroups in the future, it will be important to develop a tool that is not based on those subgroup categories (or at least can still type accurately when some of the predictor variables are limited). These are only some examples of ways that client needs can constrain typing tool development. It is important that analysts consider any client needs that may apply, and adjust the typing tool accordingly.

When practical, the best situation is when an analyst can develop the typing tool at the same time the initial segmentation is being developed. In some cases this is not practical for a variety of reasons (including tools being developed for existing segmentations), but when it is possible it can not only enhance the effectiveness of the tool, but can also help aid the development of the segmentation itself. By developing the tool in concert with the segmentation, the analyst can ensure that the segmentation solution lends itself to a strong tool. When this is not the case, it is important that the analyst can determine if this is a flaw in the segmentation, or in the tool. In some cases, the tool itself may define a clearer segmentation than the techniques used to create the initial segmentation. In all segmentations, there are those that are strong fits in a segment, and some respondents that are weaker fits, or may be between segments. In some cases, reviewing the segments defined by the typing algorithm may place these weaker fit respondents better than the original segment definitions. In these cases, the best solution may be to define the segments using the tool, rather than the initial multivariate results. In some sense, this would have the added benefit of making the tool more accurate to the segments being analyzed, although it would still be incorrect to assume the tool is 100% accurate, as it would still likely fall short of that lofty threshold with out of sample respondents. In some other cases, lack of ability to develop a satisfactory typing tool might cause the analyst to re-think the segmentation itself, opting for a different solution. In best cases, coming at the segmentation development from both angles simultaneously should lead to better segmentations overall, as well as better tools.

In some cases, analysts can become too narrow sighted, tweaking their typing tools and segmentations based on statistical measures. Certainly this is an important part of tool development - seeking a strong model and the 'best fit' - but it is possible to become too bogged down in statistical measurement, and lose sight of what is practical. For this reason, it is important to perform more qualitative tests on the typing tools as well.

To perform these qualitative tests, have non-analysts “kick the tires” of a proposed typing tool solution. This test can be a crucial way to determine how well the tool actually works. Sometimes this review can yield significant issues with a tool, such as a person who clearly should belong to one segment being mistyped. In these cases the analyst should be able to determine the cause of the mistyping, and be able to either explain why it is not an issue, or be able to create a new algorithm based on that learning. Other times this more qualitative review will help flesh out poorly asked question combinations, or other model specification issues that may or may not need to be addressed. Either way, this less rigorous form of external validation is an important step to ensuring that the final typing tool is sound both statistically, but also practically.

Ultimately, the analyst has a lot to consider in creating an effective and useful typing tool. This is an important task, as the success or failure of the underlying segmentation solution can often depend on the ability to successfully type future respondents. As noted, there are many tools available to analysts and three common/accessible ones are discussed in this paper.

6. Conclusion

The selection of the appropriate tool often depends on the underlying data, and the needs of the client. Each tool has its advantages and disadvantages which analysts should be aware of. Likewise each segmentation project has its own unique quirks and challenges that must be considered when creating a typing tool. It is certainly beneficial for analysts working on segmentation projects to be familiar with a number of different methods, as well as the potential benefits and pitfalls of each method. In many cases, it can be instructive to develop typing tools for the same segmentation using multiple methods as part of the development process. Often the tools will be similar, as the same predictor variables will prove to be important differentiators in all models, but the subtle differences can be extremely helpful in diagnosing potential areas for improvement. This also provides an opportunity for the analyst to learn if one technique is truly more effective for that particular segmentation. By spending the extra time and effort, the analyst can help ensure that they develop the optimal typing tool that will ultimately lead to client satisfaction.

This paper is not meant to be an exhaustive evaluation of potential techniques, but rather a discussion of the more popular methods used to create typing tools. We believe that the industry would be well-served to continue looking at more efficient and accurate ways of typing respondents – not just statistically but also practically. We believe this area is ripe for further research.