Automatic Plotting with Blindfire Plotter

Jimmy Oh

1. Introduction

Visualizations of data are valuable; the right plot can quickly communicate a feature of the data that may have only become apparent after substantial digging. A valid question then is how to find the right plot.

The *Pairs Plot*, introduced as *Pairwise plots* in Hartigan (1975), addresses this question by providing a scatterplot for every possible pair of variables. A simple idea, yet quite powerful, the pairs plot is capable of providing a wealth of information in a single plot. The *Generalized Pairs Plot* (Emerson et al. 2012) extends the simple idea to add different types of plots, further enhancing its value in data exploration. Yet the pairs plots are limited by the fact they only consider pair-wise relationships, and in many cases that is not enough.

Blindfire Plotter tackles the question by automatically generating a plot for every known combination of factors, from simple plots of pair-wise relationships to more complex plots utilizing superposition and juxtaposition to cross-classify by multiple factors at once. Modular in construction, Blindfire Plotter is easy to customize and extend allowing it to aid in the exploration of any dataset.

1.1 The Data: New Zealand Secondary Schools Data

To illustrate the potential of Blindfire Plotter we will explore New Zealand Secondary Schools Data released on-line by the *New Zealand Qualification Authority* (NZQA). The data relates to the *National Certificates of Educational Achievement* (NCEA) which is New Zealand's nation-wide secondary school qualification system. Though the data is released

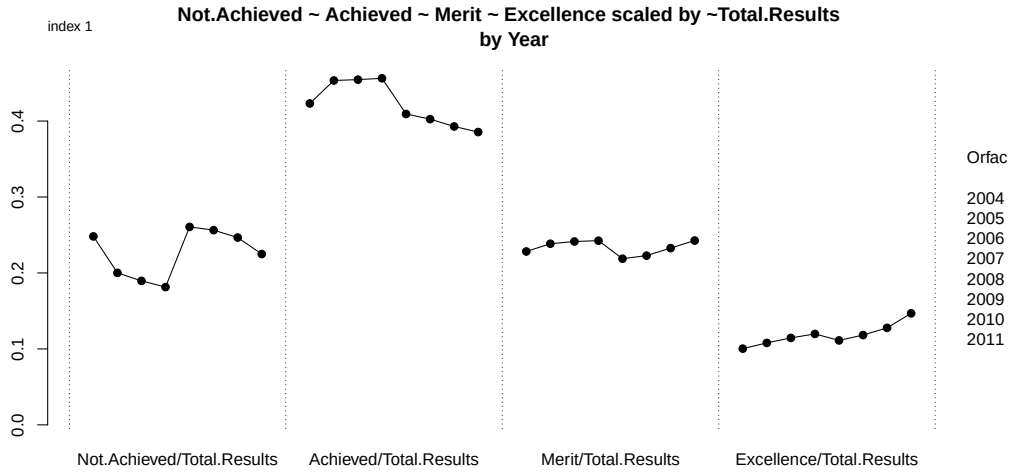

Figure 1: An example output of Blindfire Plotter, here we see the proportion of results falling into each grade category through the years. We see that roughly 20-25% of results are fails (Not.Achieved), with a dip for the years 2005-2007. On the whole, the proportion of good results (Merit and Excellence) seem to be increasing as time passes. The plot type used is a custom line plot function included with Blindfire Plotter.

in a form unsuitable for immediate analysis, for brevity's sake we will not cover the details of processing and assume the data is already in a nice data-frame.

```
  Year    Subject Standard NCEA.Level   Ethnicity Gender
1 2004 Accounting      Ext    Level 1    NZ Maori   Male
2 2004 Accounting      Ext    Level 1    NZ Maori Female
3 2004 Accounting      Ext    Level 1 NZ European   Male

  Total.Results Not.Achieved Achieved Merit Excellence
1          1458          697      365   277        119
2          1990          966      549   322        153
3         12269         3137     3513  3420       2199
```

The data contains in total 12,006 rows covering the Years 2004 to 2011, 33 Subjects and five Ethnicities. Standard denotes whether the examinations were assessed Internally by the schools or Externally by NZQA. NCEA consists of three levels which corresponds to the final three years of secondary school education in New Zealand. Four grades are possible in NCEA: Not Achieved (NA) - Fail, Achieved (A) - Pass, Merit (M) - Good Pass, and Excellence (E) - Very Good Pass.

The data is by no means massive but is still sufficiently large to make any manual exploration time-consuming. The thought and work involved in making any individual plot may be modest, five to ten minutes perhaps, but proper exploration of the various interactions will take several plots, quickly pushing the required time to hours. Blindfire Plotter can relieve a user of the time and effort involved in producing the plots, allowing a greater focus on interpretation.

2. Basic Usage of Blindfire Plotter

Blindfire Plotter is implemented as a library of functions in R (R Development Core Team, 2012) and basic usage simply involves making a call to the function `BlindfirePlot`. In this section we cover some simple ways to call this function.

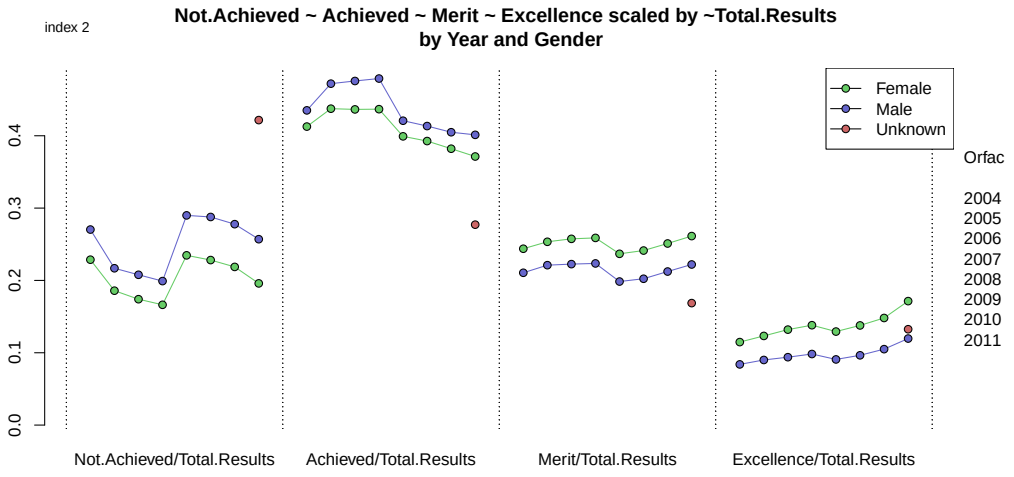

Figure 2: Here we see the proportion of results falling into each grade category cross-classified by Year and Gender. We see that the overall pattern seen in Figure 1 applies to both genders, and that on average Females consistently out-perform Males in the good results categories (Merit and Excellence).

2.1 A Simple Call

A simple call involves just four arguments specifying the Response (`respform`), a Scaling variable for the Response (`scaleresp`) and two categories of factors for Ordered (`orfac`) and Unordered (`unorfac`) variables. The following arguments produce Figures 1 and 3.

respform `Not.Achieved ~ Achieved ~ Merit ~ Excellence`

scaleresp `~ Total.Results`

orfac `"Year"`

unorfac `"Standard"`

Figure 3 is a demonstration of the power of the right plot. It not only reveals a problem with the data, but it also informs us of the exact nature of the problem - we are missing NA results for 2004-2007 for Internal standards. It turns out that when NCEA was first implemented in 2004, schools were not required to report NA results for Internal standards, so no schools did until NZQA changed the policy in early 2008.

Unfortunately, this is a problem that is easily missed with almost any other view of the data. The unusually high NA rate for External standards in 2004 contrives to hide this problem in Figures 1 and 2. It is only after cross-classifying by both Year and Standard that we so clearly see this problem. A pairs plot of the data may have revealed an unusually high number of zero NA results, but would not have conveyed the exact nature of the underlying problem. This is a clear example of the dangers of not examining many, if not all possible combinations of factors and would have led to incorrect analysis if the problem had been left undiscovered. Even after discovery, correcting for the problem can result in a lot of wasted time and effort. Indeed, when this data was first explored, it was done via manual plotting of variables of interest. This problem was only discovered hours into the process and suddenly rendered all past work worthless.

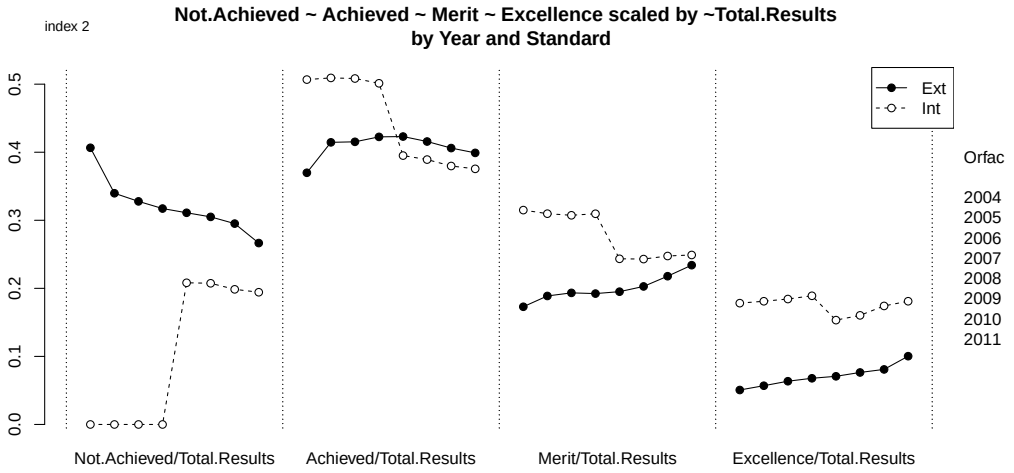

Figure 3: Here we see the proportion of results falling into each grade category (cross-classified by Year and Standard. We now see a critical problem with the data that we did not see before, one that essentially renders Figures 1 and 2 meaningless. If we had been plotting manually, going back and correcting for this problem would have been time-consuming, but with Blindfire Plotter the process is almost effortless.

2.2 Modifying The Call

There are several ways we could attempt to correct for this problem and all of them are time-consuming if the plotting process had been manual.

Two easy ways to correct involves simply ignoring all Internal results, or ignoring the first four years. Both options remove about half of our data but would at least make any previously used plotting code salvageable.

One harder way would be to ignore the NA results, this retains a lot more information as we would still have three results categories remaining, but would require significant changes to any manual plotting code.

Fortunately using Blindfire Plotter, this harder way is in fact very easy. We can adjust our arguments `respform` and `scaleresp` to `Achieved ~ Merit ~ Excellence` scaled by `~Achieved + Merit + Excellence`, removing NA and giving us Figure 4. Other variations to the arguments are possible, such as `~ Merit + Excellence` scaled by `~ Achieved` to plot a ratio of good results (M + E) to a regular pass (A).

2.3 A Full Call

Blindfire Plotter makes variations as seen above very easy and can provide substantial gains in time even when dealing with a single plot. When we expand to a Full Call involving all the various factors, we can achieve massive gains as Blindfire Plotter re-plots a whole family of plots automatically. A Full Call is achieved by extending our `orfac` and `unorfac` arguments. Our `respform` and `scaleresp` could be any number of variations, but we will choose `~ Merit + Excellence` scaled by `~ Achieved`, effectively transforming our response to a single variable. This produces 56 plots in about four seconds.

respform `~ Merit + Excellence`

scaleresp `~ Achieved`

orfac `c("Year", "NCEA.Level")`

unorfac `c("Subject", "Standard", "Ethnicity", "Gender")`

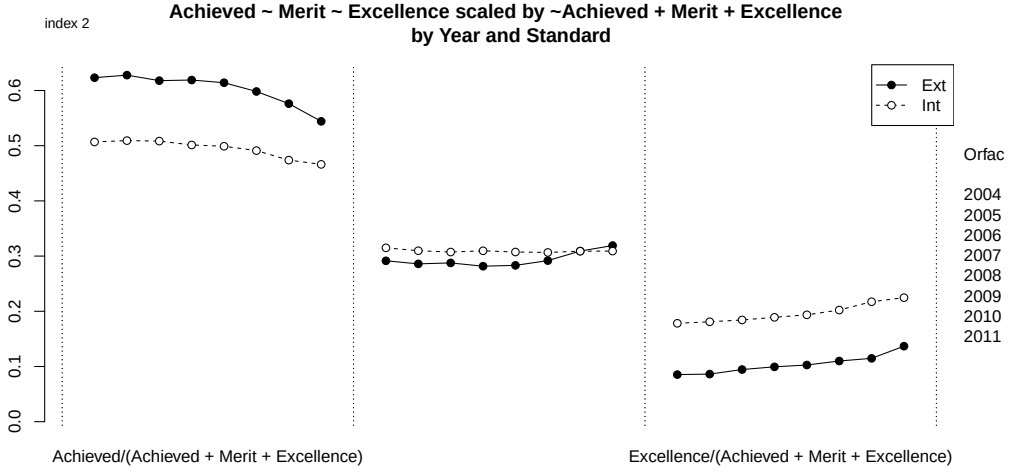

Figure 4: Here we ignore NA results and plot the results falling into the remaining grade categories as a proportion of pass results, cross-classified by Year and Standard. Because of the length of our scaling variable we lose the label for the middle result, but the automatically generated title makes it apparent that this is Merit. We also see a clear pattern of students performing better in Internally assessed standards, with proportionately more students getting a very good result (Excellence).

The types of plots that Blindfire Plotter generates depends on the types of variables specified in the input. Where multiple response variables are specified, as in Figures 1-4, we note one of the axes is given to classifying by the different response variables. Conversely, when a single response is specified, as in Figures 5 and 6, this axis is freed up allowing us to cross-classify by more variables in the same plot. Additionally, whether the factors specified are Ordered or Unordered will affect how they are used in combination with other factors, and such combinations will determine the type of plot used. For instance, connecting the points in a line plot only makes sense if that factor is Ordered.

Despite all this, true to the name Blindfire, this automatic plotting can produce a large number of useless plots such as Figure 5, where Blindfire Plotter attempts to superpose by

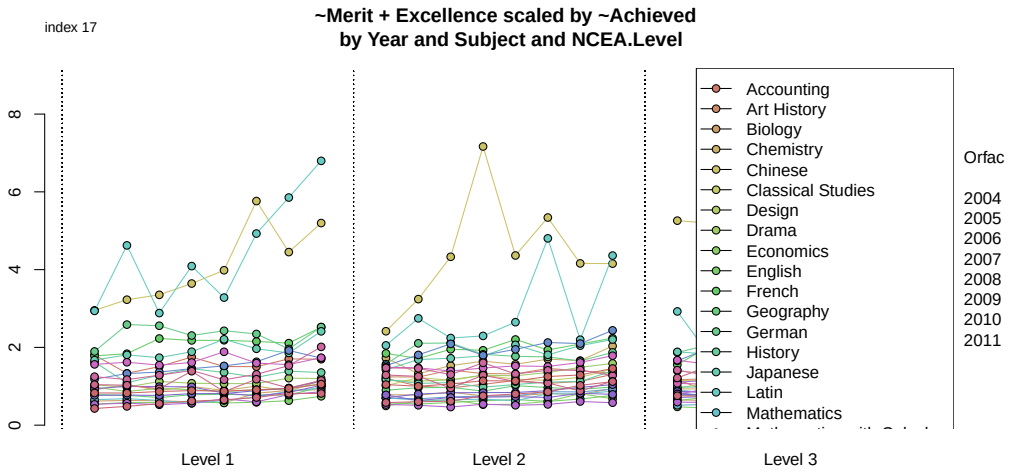

Figure 5: Here the sheer number of subjects (33) has led to just too much data to plot. The nature of automatic plotting can result in output that is completely useless, but these come at a minimal cost precisely because the plotting is automatic.

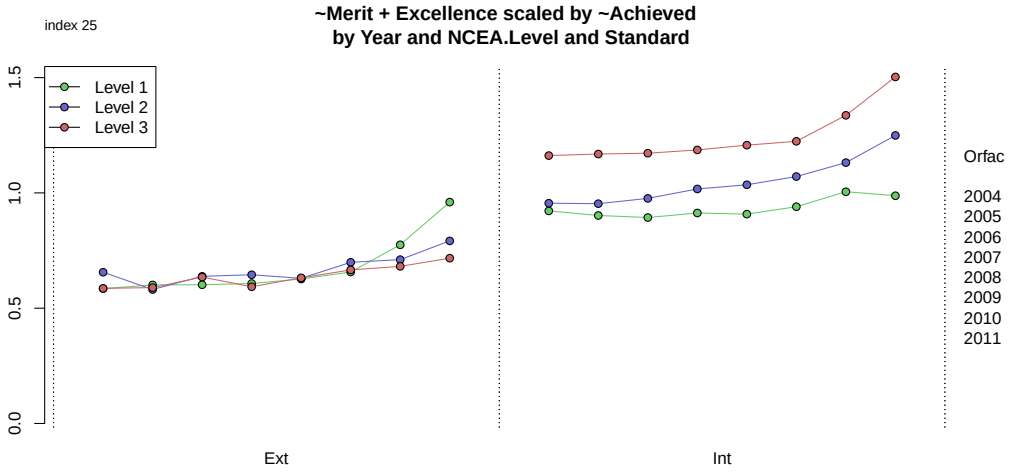

Figure 6: Here we cross-classify by three factors at the same time revealing an interesting feature. We notice that the ratio of good results (Merit + Excellence) to regular pass (Achieved) is roughly the same for the three NCEA levels for Externally assessed standards. Yet we see a clear pattern of higher ratios for higher levels in Internally assessed standards. This incidentally corresponds with teachers having a longer time to get to know their students better, and it is these teachers that mark Internally assessed standards. Coincidence? Perhaps, but automatic plotting of so many combinations can lead to such unexpected insights. Of course, caution must be applied when interpreting such results as dissecting any dataset to a fine enough detail can reveal correlation purely due to chance.

Subject which has 33 categories. The result is simply too much data and the plot does not provide any meaningful insights to the data.

However Blindfire Plotter can also plot several useful views of the data, such as Figure 6, where cross-classifying by three factors at the same time has led to an unexpected insight. In Internal standards students perform better as they progress to the higher NCEA levels, even though the content should become more difficult for the higher levels. We may exclude the possibility that this is the result of the better students remaining, as student performance seems largely unchanged in the External standards.

3. Future Work

Blindfire Plotter has been constructed in a modular fashion and it is already possible to customize and extend its capabilities in various ways, such as specifying custom plotting functions or adding completely new *modules* to cover new combinations of factors. However, continued usage of Blindfire Plotter quickly indicates a new direction to pursue.

Freed from the time cost of plotting, the focus shifts away from single plots that retain a lot of information - a tactic once valuable for maximizing return for a time costly plotting process, to using a package of plots. Such a package can start with *screening plots*, simple plots to quickly identify whether there are any interesting features to examine. Once these are identified, the scope can be narrowed and a collection of more complex plots can be used to investigate the feature. Interactive tools can make it possible for users to logically 'zoom' in on a screening plot to obtain a more complex plot, making the connections even more seamless. The use of screening plots can minimize time wasted looking through meaningless plots, while still taking advantage of examining, however briefly, a view of every possible combination of factors.

4. Conclusion

Visualizing data is valuable and Blindfire Plotter can lead to significant savings in time with its automatic plotting. It is particularly useful for examining variations of the response variable, whether to correct for some problem with the data or simply to obtain a different perspective. As it plots every possible combination, up to a certain complexity, Blindfire Plotter can be used to identify and examine problems that may otherwise have been missed.

Still in active development, Blindfire Plotter already shows impressive potential, making the plotting process much easier, more comprehensive and leading to reconsideration of what might be the most appropriate strategy for data exploration.

The latest source code for Blindfire Plotter is available from `https://github.com/joh024/BlindfirePlotter`.

REFERENCES

Emerson, John W. and Green, Walton A. and Schloerke, Barret and Crowley, Jason and Cook, Dianne and Hofmann, Heike and Wickham, Hadley (2012), "The Generalized Pairs Plot," *Journal of Computational and Graphical Statistics*, URL `http://amstat.tandfonline.com/doi/abs/10.1080/10618600.2012.694762`

Hartigan J. A. (1975), "Printer graphics for clustering," *Journal of Statistical Computation and Simulation*, 4:3, 187-213

R Development Core Team (2012), "R: A language and environment for statistical computing." R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL `http://www.R-project.org/`

Secondary School Statistics, *NZQA*, URL `http://www.nzqa.govt.nz/`