Equality and diversity analysis of performance management outcomes: a natural way of presenting results from ordered logistic regression, and sharing methodology using GitHub and R

Sumit Rahman

6 September 2016

## Abstract

In BIS I used ordered logistic regression to analyse performance management outcomes, looking to present the model's estimated effects to senior non-analysts in a way that they understand. The key purpose of this analysis is to establish if a person's age, ethnicity, gender etc affect their probabilities of receiving the various scores in the performance management system. I show a way of deriving these probabilities that focuses on the 'pure' impact of each individual demographic variable after correcting for the confounding effects of the other variables in the model. These probabilities are consistent with the overall observed distribution of outcomes (not something that traditional 'marginal probabilities' achieve) but are also consistent with the estimated effects found in the regression. A way of deriving an effect size for each variable is outlined.

I also explain how this method has been implemented in the R programming language, by writing an R package and making it available on the popular code repository GitHub. The fact that this paper has been written in **knitr** is also discussed, as a way of introducing the reader to the idea of reproducible research.

## 1. Introduction

For several years I have been analysing performance management data in the Department for Business, Innovation and Skills (BIS - now part of the new Department for Business, Energy and Industrial Strategy) for my colleagues in HR who are interested to see if outcomes depend on demographic or other characterstics. I currently use ordinal logistic regression with performance management outcome as the dependent variable and available demographic (and other) information as independent variables. In these sort of analyses it is common to express model estimates using odds ratios, but I don't think these are understood by my HR colleagues or other senior colleagues who are required to make decisions based on this analysis. Model effects given in terms of probabilities would be much more preferable. In this paper I describe one way of deriving 'corrected probabilities' that express the model results and have some desirable properties that make them suitable for reporting to non-analytical colleagues.

The rest of the paper is organised as follows. Section 2 describes the performance management system and the data used for analysis. Section 3 describes the ordinal logistic model and other possible models. Section 4 shows how to derive the 'corrected probabilities'. Section 5 discusses various issues with these probabilities and some ways of extending their use, including a derivation of an effect size for each variable. The final section talks about how this methodology has been implemented in the R programming language, with the code openly available in GitHub, and talks about the use of R Markdown to create reproducible research, which is how the current paper has been produced.

## 2. Data and the performance management system

### 2.1 The performance managment system

Most officials are subject to a system that is largely the same throughout the Civil Service. A major exclusion is the Senior Civil Service, which is subject to its own system. Other exclusions include recent entrants to civil service who are on probation, officials on secondment, maternity leave, long-term sick leave etc. The reporting year runs from April to March, and after March officials are assessed according to how well they have performed against their objectives as agreed at the start of the year. Officials are rated using a scheme with three categories: in BIS these were called *exceeded*, *met* and *improvement needed*. Each departments/agency sets out at the start of the year a 'guided distribution' for these categories and aims to ensure that its ratings conform to this distribution. The distribution is for the full set of ratings - no distribution is specified for subpopulations. In BIS for 2015-16 the guided distribution was 20-25% for *exceeded*, 65-70% for *met* and 10% for *improvement needed*. Halfway through the year a mid-year review takes place but the results from this do not directly impact the end-year ratings and these mid-year results are not used by me in my analysis.

### 2.2 The data

I get data from HR in the form of a rectangular dataset, with fields indicating each person's gender, age (in five-year bands), management group, ethnicity etc. There were approximately 2000 records in BIS. For a careful analysis you need to make the effort to understand how these fields have been populated (which are the result of voluntary self-declarations, for example), look at item non-response, which variables are derived etc. For the purpose of *this* paper details of how the data were cleaned are unimportant and will not be discussed further.

To illustrate the ideas in this paper I will make use of an artifical dataset **test.data** available in my R package **perfman**. This data contains 200 records and all the variables (performance management outcome, gender, ethnicity, disability and age) are categorical. (More details about **perfman**, including how to access it, will be provided in the final section of this paper.)

## 3. Ordinal logistic regression

### 3.1 The model

The model I have been using is sometimes called ordinal logistic regression; it is also known as the proportional odds logistic regression model. (The model is described fully in McCullagh and Nelder (1989), and an accessible introduction is available in Agresti (1996).) It is widely available in statistical software (e.g. the **polr** function in the **MASS** package in R, or the **ologit** function in Stata - the Stata documentation for this function is helpful.) The model exploits the fact that our outcome variable is ordered (*met* is between the other two outcomes) by applying the logistic function to the 'cumulative odds' of each outcome. It is the fact that the outcomes are ordinal that ensures that cumulative odds are meaningful.

### 3.2 Other models

Other models can be used to analyse this sort of data. Agresti (1996) describes some other models that exploit the ordinal nature of the outcome in different ways, such as using paired category logits, and continuation ratio logits. Another option is to ignore the ordinal structure and fit a multinomial logistic model. This might be sensible if some of the assumptions behind the other model are difficult to justify, but the cost is that there will be more parameters to estimate. The ordinal models will be more parsimonious.

It is usually possible to fit models that use a probit link function rather than a logistic one. In practice you do not tend to get results that are very different whether you use the logistic or probit versions of the model.

The method for deriving 'corrected probabilities' outlined in this paper will use ordinal logistic regression, but if you understand the method it will be reasonably straightforward to apply it to these other models. The ordinal logistic model is particularly simple and elegant to use in this case though and makes a good starting point for applying this method.

## 4 Deriving the corrected probabilities

## 4.1 Predicted probabilities from the model

Having fitted the model it is simple to produce predicted probabilities for each individual in the dataset. Statistical software will usually do this straight away, but in the case of ordinal logistic regression we can see a simple formula:

[formula for predicted probabilities]

In this formula **x**b is the usual way of picking out the coefficients corresponding to the relevant dummy variables, and kappa are called 'cuts'. These depend on which outcome we are looking at. In test.data, *improvement needed* is coded as 1, *met* as 2 and *exceeded* as 3, so the probability of *met* for an individual would be given by using **x**b for that person and kappa2 and kappa1 in the formula above. (The way I have presented this formula, you can interpret kappa0 to be minus infinity and kappa3 to be infinity.)

We can then define a function that takes values of **x**b and outputs three probabilities (one for each outome). It is easy to see that these probabilities sum to 1.

Take our example data set, and suppose that we fit a model using only gender and disability as explanatory variables. The estimated coefficients are:

|  |  |
| --- | --- |
| Variable | Coefficient |
| Gender: Female | 0 |
| Gender: Male | -0.79 |
| Disability: Disabled | 0 |
| Disability: Not disabled | 0.35 |
| Disability: Not known | -0.02 |
| Disability: Prefer not to say | 0.66 |

The first person in our dataset is male and not disabled. The **x**b value for him is -0.79+0.35=-0.44; the values for kappa1 and kapp2 are -2.07 and 1.78 respectively. Using the formula above leads to predicted probabilities for *improvement needed*, *met* and *exceeded* of 0.17, 0.74 and 0.1 respectively. In this case the modal probability is for *met*, and this was the actual outcome.

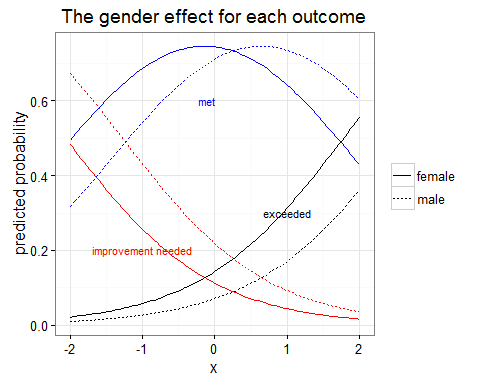
In a similar way we can get probabilities for a person who is male and disabled. These are 0.22, 0.71 and 0.07. This shows the 'disability effect', which is driven by the coefficient 0.35, but the impact on probabilities will differ depending on the baseline probability. This coefficient is a constant effect on the logit, but that means the probability impact is *not* constant.

## 4.2 Generalising the probability function

The function that takes **x** (where **x** is a column of zeros and ones to pick out dummy variables) and produces the three probabilities can be generalised so that *x* is any real number. Call this function P so that P(*x*) gives the three probabilities where **x**b in our formula above takes the value *x*. We can say that the gender effect is the difference between P(*x*) and P(*x*-0.79), whatever the value of *x* is.

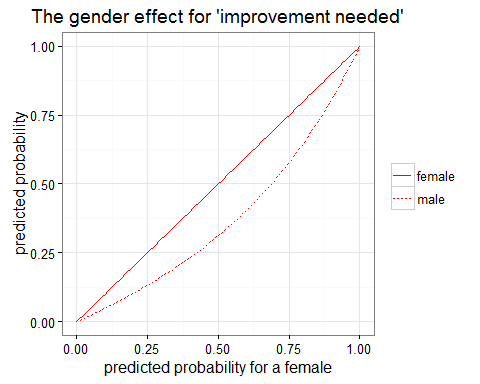
How can we justify the idea that *x* can take any value? One way is to think of it as a summary of a group of people. Suppose we had a group of women, with a variety of disability statuses, their mean **x**b value could take on a range of values (albeit bounded above and below by the range of estimated coefficients for the disability statuses). If we then imagine a similar group of men with an identical distribution of disability statuses, its average **x**b value would be 0.79 less. P(*x*) and P(*x*-0.79) would then be the two probability distributions for outcomes, one for each gender.

## 4.3 Selecting the family member with the best implied distribution



This graph illustrates the gender effect accross a wide range of probabilities. Note that the horizontal axis is the abstract *x* value while the vertical axis is on the probability scale. The lines for men are effectively horizontal translations of the lines for women, as we are comparing *y*=P(*x*) (the line for women) with *y*=P(*x*-0.79) - the male line is the female line shifted to the right by 0.79 units. Because this translation is horizontal, the impacts on probability vary by *x* value. For example, when *x*=-1 the male probability for getting exceeded is 3.1 percentage points less than the female one; but when *x*=0 the difference is 7.3.

Another way of thinking about the impact on probabilities is to ask the question 'if the probability for a particular woman to get a particular outcome is *p*, what would be the corresponding probability for someone with identical characteristics except that he is a man?'. The following graph shows this (note that both axes are on the probability scale and that the female line will be the line '*y*=*x*' by definition):



In many ways I think this last graph is the 'best' answer to the question 'what difference does each demographic variable have on your probabilities of each performance management outcome?' But it still takes a bit of thought to understand fully, and I want to present a single instance of a probability effect that is in some way representative of the overall story in the data.

### 4.3 Selecting a single family member

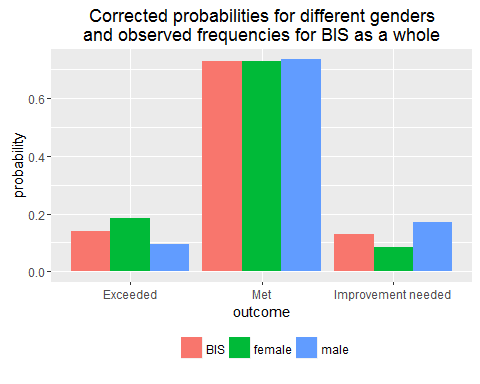
If we go back to the first graph, we can see that each choice of *x* value gives a full set of probabilities for both genders and all three outcomes that are consistent with the effects estimated by the regression and still sum to 1 across the outcomes for each gender. What we have is a family of probability distributions, indexed by *x*. Consider the family member indexed by *x*=0, and the outcome *exceeded*. We can see that the probability for women is 0.14 and the probability for men is 0.07. In a world where the gender effect is the only one, this implies that the probability for the department as a whole will be between these two values. (I am trying to illustrate the gender effect, which has been fitted as a main effect with confounding effects corrected for, so acting as if the gender effect is the only one seems justifiable.) In fact, we know from our dataset the exact number of women and men and so can arrive at a probability for the department as a whole by taking the weighted mean of the female and male probabilities. And we can do this for the two other outcomes. This means that we have a probability distribution for the department as a whole, which I call the *implied distribution*. Different family members (i.e. different choices of *x*) will result in different implied distributions. We can then select the family member whose implied distribution is closest to the actual observed distribution for the department as a whole (or you might want to be as close as possible to some pre-specified target distribution).

What we need now is a distance function to let us measure how close any implied distribution is to the observed (or target) distribution. What I have used in practice is a sum of squared differences function, though it is easy to think of other options (such as sum of absolute differences). One advantage of taking squared differences is that large differences in any one outcome get penalised heavily, leading to an optimised choice of implied distribution that has no big differences, even at the expense of tolerating several small differences in other outcomes.

Using this distance function, the following table shows how the distance is calcualated for the family member *x*=0:

|  |  |  |  |
| --- | --- | --- | --- |
| gender | *exceeded* | *met* | *improvement needed* |
| female | 0.14 | 0.74 | 0.11 |
| male | 0.07 | 0.71 | 0.22 |
| implied | 0.11 | 0.73 | 0.17 |
| observed | 0.14 | 0.73 | 0.13 |
| difference | -0.03 | 0 | 0.04 |

The 'distance' between implied and observed is the sum of these three differences after they have been squared. A different value of *x* will lead to a different implied distribution and so a different distance. After finding the *x* value that gets us closest to the observed distribution, the probabilities associated with that family member are what can be reported as 'corrected probabilities'. In this example here for gender the optimal value of *x* is 0.3063715, and the corrected probabilities are illustrated below:



## 5 Interpretation and extensions

### 5.1 Interpreting the corrected probabilities

How should we interpret these corrected probabilities? The important thing to remember is that they are consistent with the effects that have been estimated in the regression model. The modelling is meant to tease apart the confounding effects of other variables, so I describe these corrected probabilities as the probabilities we'd get for this variable if there were no other effects but the overall probabilities for the department were unchanged. This way there is no confusing talk about the 'average person' in the department. It would be interesting to do some comparisons between the chosen *x* values (that minimised the distance between the implied and observed distributions) and measures of the 'average *x* value'. My sense is that these will be pretty similar but it ought to be investigated.

### 5.2 Small effects and poorly fitting models

Where the effect is small the differences in probability for any given outcome will be small. This can lead to problematic corrected probabilities when the logisitic model is not a very good fit. For example, it is possible that the corrected probabilities of *met* for both women *and* men are both higher than the observed relative frequency of *met* in the department as a whole. This will be because the distance function has been found to be optimised even with this anomaly, presumably because there are even bigger problems in getting the implied distribtution right for the other outcomes.

### 5.3 Interactions and continuous predictors

There is no space in this short paper to outline how this method can be extended to deal with interactions and continuous predictors, but I have been able to do this. The current version of the **perfman** package does not deal with interactions or continuous predictors, but this is certainly possible.

### 5.4 Effect sizes

Another worthwhile extension of this work is to come up with effect sizes so that we can rank the various effects (for example to say that the gender effect is bigger than the disability effect). This might be useful information for HR and senior leaders to make informed decisions about what effects should be prioritised in departmental efforts to deal with the equality and diversity issues that the analysis has brought to light. I have come up with a way of calculating effect sizes, though there is no space here to describe it properly and it has not yet been implemented in the **perfman** package. The basic idea is to use the corrected probabilities for a particular demographic variable to produce a contingency table of 'corrected frequencies' for the department (e.g. in the case of gender have a contingency table of gender by outcome). Then the usual sort of chi-squared analysis for testing for an association between outcome and gender (for example) can be performed. Cohen (1988) gives some effect size estimates that are based on this procedure.

## 6 Some data science ideas

## 6.1 Writing code and sharing it

I have been mentioning an R package **perfman** that I have written to implement the methodology described in this paper. The R programming language is freely available and is widely used by statisticians around the world. A package is essentially a set of functions that can be installed quickly by an R user on her computer for immediate use. It is straightforward to write a package so that help files are also accessible to users, and you can also include examples, tutorials, data sets etc. I have made the **perfman** package available by depositing it on GitHub, a popular repository for code, which means that you can examine the code and other information easily by going to <https://github.com/sumitrahman/perfman>. R users can make use of the **devtools** package to install the package by entering the command install\_github(âsumitrahman/perfmanâ) in R. Once the package is installed, you can use the **fit.model** function on a dataset and then the **recommend** function. For example, if you enter **fit.model(test.data)** and then **recommend(âgenderâ)** you will see the corrected probabilities behind the chart in section 4.3.

There are a number of advantages of using this way of sharing methodology. The code makes the steps explicit and allows users to implement it immediately. By being open, other users can inform me of errors or improve the code in different ways. Github has several features which make it ideal for collaboration. For example, someone could âforkâ my repository (i.e. make a copy without affecting my version), amend the code so that it can deal with probit ordinal regression as well as logistic, and then inform me: if I like it I can easily fold it back into the original repository.

Clearly there are lots of datasets we work on which we cannot openly publish in this way because of confidentiality, but generally there is no reason why we should not be open about our methods. I suggest that using Github as a way of sharing code (in any language) is a sensible way of making our methodology more widely understood, more robust, better quality and ultimately can make our work more efficient.

## 6.2 Reproducibility - this paper

One useful feature of using R with the freely available RStudio (an interface for using R) is that it is easy to make a start with producing reproducible research. This is a way of writing reports of analysis where the statistical outputs are created and plugged into the report by software. For example, the paper you are reading now was almost entirely written using an R markdown file (which is available for examination in the **perfman** Github repository â look for the file called âGSSM21 paper.Rmdâ). The numbers in this paper (for example in section 4), including the tables, and even the charts are all the result of code and this code can be examined by anyone. In other words I am âshowing my workingâ. The value of the male coefficient hasnât been copied by me from a computer screen and typed into or pasted into this document â it has been calculated and placed into this document within R itself, using the **knitr** package. You could reproduce this paper in seconds and then check to see that the numbers and charts really do come from the underlying data. (You would need to have the **perfman** package installed.)

In cases where the underlying data is confidential and cannot be shared, the ability to reproduce the research in this way disappears. I think there is merit in providing an artificial dataset that is similar in structure to the real data, so that users can still see that the code works and how it turns data into the statistical outputs in your report. This is essentially what I have done here: instead of sharing the actual performance management outcomes for BIS colleagues, I have made the artificial **test.data** and used this to illustrate the methodology for arriving at corrected probabilities. Without actual data to work on, you still have effectively published the code behind your report, but users have to trust you that there is data that the code can work on.

Another advantage of the reproducible research approach is that if I change the underlying data, I can update this entire paper with the press of a button: all the numbers, tables and charts will be consistent with the new data. Obviously commentary on the results would need to be updated if there are substantive changes, but hopefully you can see how this system could save a lot of time for statisticians who have to produce regular reports every month or quarter using data that is updated at these frequencies.

There are a handful of additions I have made to the Word document that is produced by the software to get to the final version you are currently reading â the formulas and **b** and kappa signs have been added by me after the fact (in fact it is possible to write these within the software but it requires knowledge of âLaTeXâ which I do not have) and I have made a few cosmetic formatting changes such as fonts for code.

To learn more about reproducible research, I recommend Xie (2015), or Roger Pengâs online course on reproducible research (which assumes you know how to use R) at <https://www.coursera.org/learn/reproducible-research>.

# References

Agresti, A. (1996). *An introduction to categorical data analysis*. New York: Wiley.

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences*. Hillsdale, N.J.: L. Erlbaum Associates.

McCullagh, P. and Nelder, J. (1989). *Generalized linear models*. 2nd edition. London: Chapman and Hall.

R Core Team (2015). *R: A language and environment for statistical computing*. Vienna: R Foundation for Statistical Computing. URL <https://www.R-project.org/>.

RStudio Team (2015). *RStudio: Integrated Development for R*. Boston, MA: RStudio, Inc. URL <http://www.rstudio.com/>.

StataCorp. (2013). *Stata 13 Base Reference Manual*. College Station, TX: Stata Press

Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*. 4th edition. New York: Springer

Xie, Y. (2015). *Dynamic documents with R and knitr*. 2nd edition. Boca Raton, FL: CRC Press