Evaluating Feature Rankings in Conjoint Analysis with Weighted Rank Differencing

Derek Miller

18 May 2018

Introduction

Discrete choice models used in conjoint analysis estimate the effects of various levels of different features of a product. After estimating these coefficients, clients and practitioners sometimes want to know which features or levels are most important to the respondent. Ranking the coefficients of a conjoint model helps inform business decisions.

While methods exist for evaluating ranked features[1], they are usually insufficient to address ranking in the context of conjoint analysis. To my knowledge, a general method for comparing the ranking of linear coefficients of a model to the ranking of known "true" coefficients does not exist. This document introduces a mathematical metric that can be used to compare the estimated feature rankings to true feature rankings, provided they exist. The typical use case for such a procedure is to evaluate the performance of a model on a ranking task. This method was developed to evaluate the results of a conjoint model at a well-known survey software company.

[1] For example, see information retrieval.

Weighted Rank Differencing

Let $\mathcal{F} = \{f_i\}_{i=1}^{n}$ represent the set of n distinct features whose ranking is to be estimated by some model $\mathcal{M}$. Define $\hat{r}(\mathcal{F})$ to be the *estimated ranking of* $\mathcal{F}$ *by* $\mathcal{M}$. The goal is to develop a metric to evaluate how far away the estimated ranking $\hat{r}(\mathcal{F})$ is from the true ranking $\rho(\mathcal{F})$. For simplicity, I will abbreviate the notation so that $\hat{r}(\mathcal{F}) = \hat{r}$ and $\rho(\mathcal{F}) = \rho$.

Example. Suppose model $\mathcal{M}$ is a standard multinomial logit model estimated on survey data about people's choices regarding different toothpaste options. Features being estimated are price, brand, and tooth-whitening. The corresponding feature set is $\mathcal{F} = \{f_1, f_2, f_3\}$ where each number is assigned to one of the three features. The true ranking by order of importance is price (f_1), brand (f_2), and whitening (f_3), thus $\rho = [1, 2, 3]$. The model estimates the ranking of these features to be price (f_1), whitening (f_3), and brand (f_2). So the estimated ranking of $\mathcal{F}$ by $\mathcal{M}$ is $\hat{r} = [1, 3, 2]$, since f_1 was ranked first, f_2 was ranked third, and f_3 was ranked second.

One way to measure the distance between ρ and $\hat{r}$ is to compute the sum of the absolute value of the differences in rank of the ranked feature lists. Using the above example, this would mean subtracting $[1, 3, 2]$ from $[1, 2, 3]$ to get $[1, 2, 3] - [1, 3, 2] = [0, -1, 1]$. Taking the entrywise absolute value produces $[0, 1, 1]$. Finally, summing the values gives $0 + 1 + 1 = 2$. So the distance between the true ranking and the estimated ranking is 2.

Unfortunately, this method doesn't tell us the score of the worst possible ranking. In our example, the worst possible ranking is $[3, 1, 2]$ which gives a score of 3[2]. The simple rank difference can be adjusted to tell us how

[2] For a uniform weighting scheme, the reverse of the true ranking produces the worst possible score. However, this is not true in general; the worst possible score depends on the weighting vector w and involves finding $\hat{r}$ that maximizes $\sum w * d(\hat{r}, \rho)^T$ where $d()$ is the absolute value of the rank difference described in the example.

close the estimated score is from the worst possible score by dividing by the estimated score from the worst possible score. Following the example, the adjusted estimated rank difference is $\frac{2}{3} \approx .667$ with 0 being the same as the true ranking and 1 being the worst possible ranking.

Still, this method is not satisfactory. How do we know that the estimated ranking is better than random? Knowing the distribution of the distance of such scores would help us evaluate the final score in context. Additionally, it is usually the case that we prefer certain rankings over others with the same score. For example, the ranking $[3, 2, 1]$ gives the same score of 2 as in the example but for certain applications, it might be preferable to correctly estimate the first ranking since switching lower rankings matter less to us.

A natural remedy to this is to apply a weighting scheme to the absolute rank difference vector. Suppose w is a vector with entries between 0 and 1 and $\sum w = 1$. Then the *weighted rank difference*[3] metric is computed as the weighted sum of the absolute rank difference vector, i.e. $wrd(\hat{r}, \rho) = \sum w * d(\hat{r}, \rho)^T$.

Example. Continuing the above example, suppose we want to put more weight on features whose true rank is higher. This means that if a high-ranking feature disagrees with the true rank of the feature, it will be penalized more than the disagreement between low-ranking feature and its true rank. Recall from the example that the absolute rank difference vector was $[0, 1, 1]$. We can choose our weight vector to be $w = [.7, .2, .1]$. Then the weighted rank difference score is now $[.7, .2, .1] * [0, 1, 1]^T = .7 * 0 + .2 * 1 + .1 * 1 = .3$ which is the final score since the worst possible score for this example is 1.

Note that the choice of weighting scheme will depend on the application. For estimating the distance between the coefficients of a model, one

[3] See Chapter 8 of *Who's # 1? The Science of Rating and Ranking* by Amy Langville and Carl Meyer for a similar approach.

appropriate weighting scheme would be to look at the absolute value of the difference of the true coefficients. If the true coefficients of the features are close to each other, then we expect that the estimated ranking will be more likely to mis-rank these two features than two features whose coefficients are very different. Thus the differences between the true coefficients can be used as a weighting scheme. Additional weight can be put on the rankings by applying a function to the coefficient differences so that certain ranks are considered more important. For example, giving more importance to higher ranked features might be represented applying an exponential function to the computed weights, i.e. $f(w) = e^w$. Alternatively, if estimating the coefficients of the levels withing a given feature (such that they sum to zero), applying a quadratic function such as $f(w) = w^2$ gives more importance to the end points of the rankings. It is left to the practitioner to determine the appropriate weighting scheme for their application.

Conclusion

Weighted rank differencing is a proper distance metric that allows practitioners to compare the rankings produced by a given model to the true rankings. This is helpful when testing new ranking algorithms on a simulated data set where true rankings are known. Weighted rank differencing can also be used to compare the distance between the estimated rankings of different models. However, this will not indicate which model is closer to the true ranking since they are not known. The most important step in using this method is determining the weighting scheme that is applied to the absolute rank difference vector.