Pairwise comparison

Active ranking [Jamieson and Nowak 2011]**:** Recovering a full ranking from pairwise comparison requires $O(n^2)$ comparisons if the pairs are randomly selected, but we can achieve $O(n \log n)$ comparisons by a standard sorting algorithm. If we have an d-dimensional embedding of items and define the ranking associated with the distance from a reference r_σ which represents a user preference, then the ranking can be recovered from $O(d \log n)$ comparisons. Figure 1 shows an example of 3 dimensional feature space ($d = 3$). Suppose we discretize each dimension into two values, 0.0 and 1.0. Then, we have $2^3 = 8$ options in the space. The active ranking approach uses a reference point r_σ in the space to represent a user preference, and the ranking (the numbers at vertices) is defined by the distance from the reference.

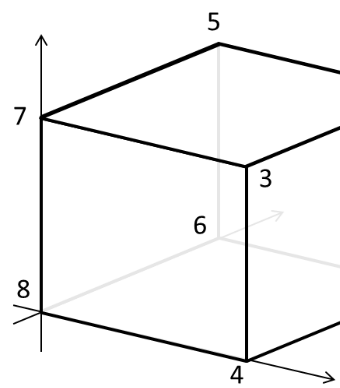

Figure 1. *The reference point* $\boldsymbol{r_\sigma}$ *represents a user preference, and the ranking is defined by the distance from* $\boldsymbol{r_\sigma}$ *to the items.*

Active collaborative permutation learning [Wang et al. 2014]: This approach applies the collaborative filtering to the ranking learning problem. Let X be the ranking matrix, each row of which represents the ranking of each user. Then, we want to decompose X as $X = UV^T$ with constraints of trace-norm and max-norm while minimizing the estimation error using hinge loss. This method requires only $O(n + m)$ comparisons in total, where n is the number of items and m is the number of users.
Although this approach is attractive in terms of the required number of comparisons, I am very skeptical about the accuracy. This approach is based on the matrix factorization, and as far as I know, the matrix factorization does not work very well in case the original matrix is sparse.

Experiments

I synthesized a set of pairwise comparisons based on my preference regarding zoning (Table 1). Here, I used three dimensional feature space, each dimension of which represents the proximity to the commercial, industrial, and park zone, respectively. Each dimension is discretized into three values, 0.0, 0.5, and 1.0. Thus, the total number of options is $3^3 = 27$.

Proximity to commercial zone	Proximity to industrial zone	Proximity to park	Ranking
1.0	1.0	1.0	19
1.0	1.0	0.5	20
1.0	1.0	0.0	21
1.0	0.5	1.0	3
1.0	0.5	0.5	4
1.0	0.5	0.0	6
1.0	0.0	1.0	1
1.0	0.0	0.5	2
1.0	0.0	0.0	5
0.5	1.0	1.0	22
0.5	1.0	0.5	23
0.5	1.0	0.0	24
0.5	0.5	1.0	10
0.5	0.5	0.5	11
0.5	0.5	0.0	12
0.5	0.0	1.0	7
0.5	0.0	0.5	8
0.5	0.0	0.0	9
0.0	1.0	1.0	25
0.0	1.0	0.5	26
0.0	1.0	0.0	27
0.0	0.5	1.0	16
0.0	0.5	0.5	17
0.0	0.5	0.0	18
0.0	0.0	1.0	13
0.0	0.0	0.5	14
0.0	0.0	0.0	15

Table 1. *The feature vectors and corresponding scores of options.*

Summary of results: Table 2 shows the summary of results.

Method	**Required comparisons**	**Kendall tau distance**
Modified quadratic kernel	$O(n \log n)$	0.0284
Quadratic kernel	$O(n \log n)$	0.0313
Our original approach	$O(n \log n)$	0.0826
Simple components ranking	$O(d \log d)$	0.0968
Active learning	$O(d \log n)$	0.2243

Figure 2. *The summary of the results. The modified quadratic kernel achieved the best accuracy , while the simple components ranking can achieve a good accuracy with just a limited number of comparisons.*

Our model, which uses a vector to represent a user preference and uses dot product to define the scores (i.e. rankings): In our model, we want to minimize the following negative logarithm of the cost function:

$$E(\boldsymbol{w}_u) = \sum_t \left[\log\left\{1 + \exp\left(\boldsymbol{f}_j^T \boldsymbol{w}_u - \boldsymbol{f}_i^T \boldsymbol{w}_u\right)\right\} + (y_t - 1)\left(\boldsymbol{f}_j^T \boldsymbol{w}_u - \boldsymbol{f}_i^T \boldsymbol{w}_u\right)\right]$$

The derivative of this by $\boldsymbol{w}_u$ is

$$\frac{\partial E}{\partial \boldsymbol{w}_u}(-\log P(Y,\boldsymbol{f})) = \sum_t (\boldsymbol{f}_j - \boldsymbol{f}_i)\left(y_t - \frac{1}{1+\exp(\boldsymbol{f}_j^T\boldsymbol{w}_u - \boldsymbol{f}_i^T\boldsymbol{w}_u)}\right)$$

Then, by gradient descent, the estimate user preference vector was $(0.571, -0.808, 0.137)$, and Kendall tau distance, which is widely used in permutation learning evaluation, was 0.082621. Figure 2 shows the relationship between the estimated scores and rankings.

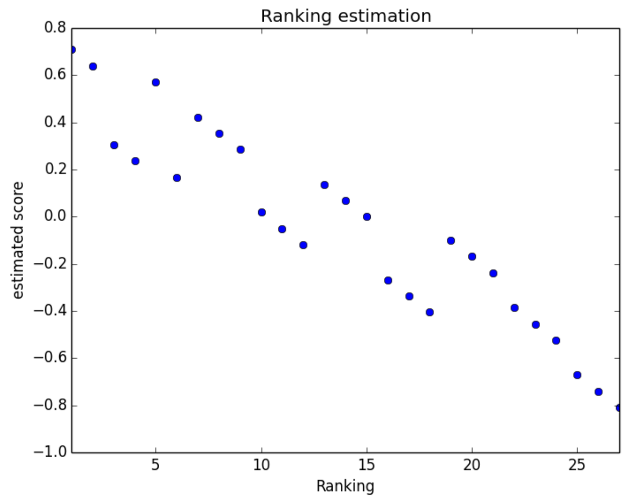

Figure 3. *The relationship between the estimated scores and the rankings.*

Active learning model, which uses a reference point to represent a user preference and uses the distance to define the rankings: In this model, the negative logarithm of the cost function becomes

$$E(\boldsymbol{w}_u) = \sum_t \left[\log\left\{1+\exp\left(\|\boldsymbol{f}_i - \boldsymbol{w}_u\|^2 - \|\boldsymbol{f}_j - \boldsymbol{w}_u\|^2\right)\right\} + (y_t - 1)\left(\|\boldsymbol{f}_i - \boldsymbol{w}_u\|^2 - \|\boldsymbol{f}_j - \boldsymbol{w}_u\|^2\right)\right]$$

The derivative of this by $\boldsymbol{w}_u$ is

$$\frac{\partial E}{\partial \boldsymbol{w}_u} = \sum_t -2(\boldsymbol{f}_i - \boldsymbol{f}_j)\left(y_t - \frac{1}{1+\exp\left(\|\boldsymbol{f}_i - \boldsymbol{w}_u\|^2 - \|\boldsymbol{f}_j - \boldsymbol{w}_u\|^2\right)}\right)$$

The estimated reference point was $(0.678, -0.691, 0.246)$, and Kendall tau distance was 0.091168.

More complex model

The above results imply that we need more complex model to achieve 100 % accuracy. Figure 4 shows the 2 dimensional feature space ignoring the third dimension, the proximity to parks, for simplicity. This indicates that the linear representation cannot fully represent the user preference.

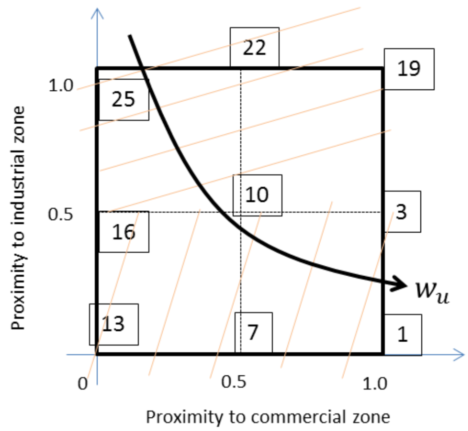

Figure 4. *The 2 dimensional feature space ignoring the third dimension, the proximity to parks.*

Typical solution is to increase the dimensionality of the feature space, and kernel trick can keep the complexity of the computation same degree as the original space.

Quadratic kernel: I first tried a quadratic kernel, $(x_1, x_2, x_3, \sqrt{x_1 x_2}, \sqrt{x_2 x_3}, \sqrt{x_3 x_1})$, which yielded Kendall tau distance of 0.031339. The relationship between the estimated scores and rankings show some improvement (Figure 4).

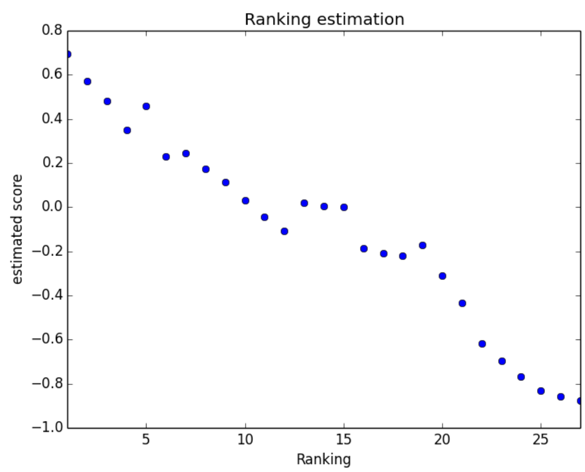

Figure 5. *The relationship between the estimated scores and rankings when quadratic kernel is used.*

Modified quadratic kernel: I also tried a little modified quadratic kernel, $(x_1, x_2, x_3, x_1^2, x_2^2, x_3^2)$, which yielded a little better Kendall tau distance of 0.02849. However, the graph shape of the relationship between the estimated scores and the rankings seemed worse.

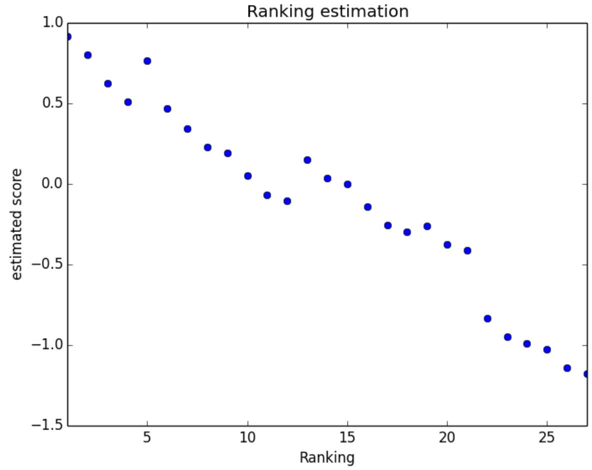

Figure 6. *The relationship between the estimated scores and rankings when a modified quadratic kernel is used.*

Simple ranking model

Here, I used much simpler model that ranks the components of the 3 dimensional feature representation, the proximity to the commercial, industrial, and park zones as well as their inversed components, and defines the preference vector in the following way: for the first three ranked components, the value corresponding to i-th ranked component is 2^{1-i}, and 0 for others. For instance, if the ranking of the six components, the proximity to the commercial, industrial, and park zones, and their inversed factors are $(2,6,3,5,1,4)$, then the preference vector will be defined as $(0.5,0,0.25,0,1,0)$. This approach requires much smaller number of comparisons. For $d = 8$, it requires only 24 comparisons.

The estimated preference vector was $(0.5, 0.0, 0.25, 0, 1, 0)$, and Kendall tau distance was 0.096866, which is worse than the previous approaches but not so bad as expected. Figure 6 shows the relationship between the estimated scores and the rankings.

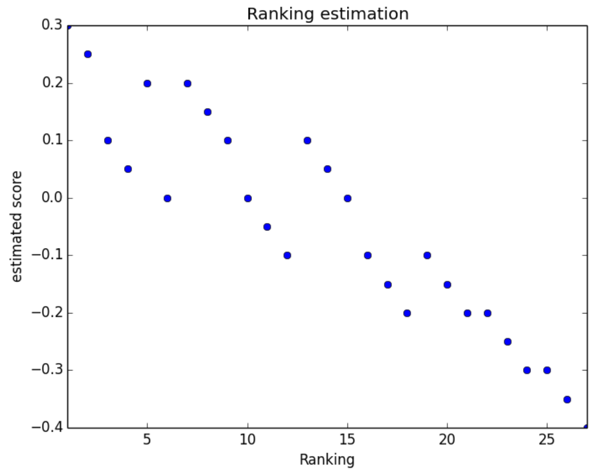

Figure 7. *The relationship between the estimated scores and the rankings when the simpler model is used.*

Conclusions

I think that we can use the simple ranking model which requires only $O(d \log d)$ comparisons while can achieve good accuracy. For the case $d = 8$, for instance, we need only 24 comparisons, and by using this approach, we do not need any adaptive approach for comparisons in order to reduce its number.

I haven't tried *collaborative* approach yet, so I will try it, but I doubt its effectiveness from my experience.

References

Jamieson, K. G., Nowak, R. D. 2011. Active ranking using pairwise comparisons. *NIPS*, pp. 2240-2248.

Wang, J., Srebro, N., Evans, J. A. 2014. Active collaborative permutation learning. *SIGKDD*. pp.502-511.