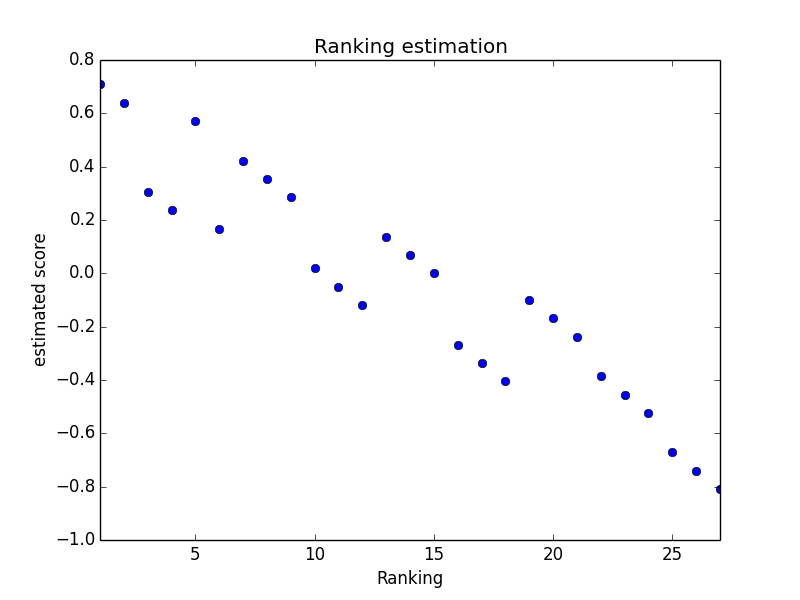
# Pairwise comparison

本当に我々の線形モデルで、ユーザのpreferenceを表現できるのか？実際に3次元のケースでやってみた。商業地への近さ、工業地への近さ、公園への近さの3つの要素について、それぞれ1.0、0.5、0.0の値（合計通り）についてランキングを作成してみた。

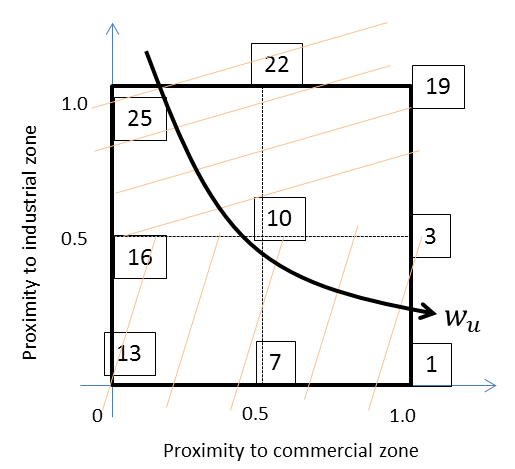
|  |  |  |  |
| --- | --- | --- | --- |
| 商業地への近さ | 工業地への近さ | 公園への近さ | ランキング |
| 1.0 | 1.0 | 1.0 | 19 |
| 1.0 | 1.0 | 0.5 | 20 |
| 1.0 | 1.0 | 0.0 | 21 |
| 1.0 | 0.5 | 1.0 | 3 |
| 1.0 | 0.5 | 0.5 | 4 |
| 1.0 | 0.5 | 0.0 | 6 |
| 1.0 | 0.0 | 1.0 | 1 |
| 1.0 | 0.0 | 0.5 | 2 |
| 1.0 | 0.0 | 0.0 | 5 |
| 0.5 | 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 |

これに基づいて、pairwise comparisonの結果を作成し、Gradient Descentを使ってpreferenceベクトルを推定すると、以下を得た。

しかし、これを使って上の27通りのオプションのスコアを計算すると、下図のようにいくつか予測エラーが発生した。この原因について考察してみた。



例えば、公園への近さが1.0の条件、つまり、商業地への近さと工業地への近さの2次元で考えてみる。ランキングは下図のようになる。これを見ると分かるとおり、直線ではpreferenceベクトルを表現できない。



直線で表現するためには、例えば工業地への近さについて、0.5と1.0の距離を広げるという案がある。

