1 Methods

I have three ideas I have tried. In the first two I cluster on the data and then learn a function for each cluster. Then I try to do a linear combination of those functions to get a good output. In the final method I take the entire dataset and use a few known ranking algorithms to learn how to combine those.

1.1 Method One

In this method we perform logistic regression on each cluster and then learn how to combine the resulting models together.

- cluster data
- for each cluster use logistic regression to learn how to rank
- Create a new data data using ranking functions to create a matrix of outputs.
- Learn coefficients for each algorithms output (linear and non-linear combinations)

1.2 Method Two

In this method we use a known learning to rank algorithm on each cluster and then learn how to combine the resulting models together.

- cluster data
- for each cluster use a known Ranking algorithm
- Create a new data data using ranking functions to create a matrix of outputs.
- Learn coefficients for each algorithms output (linear and non-linear combinations)

1.3 Method Three

In this method we perform learning to rank using multiple learners on the test data and then learn how to combine the models together.

- For each ranking algorithm train on the entire data set
- Create a new data data using the outputs of each ranking functions
- Learn coefficients for each algorithms output (linear and non-linear combinations)

2 Combining

For combining I have two ideas: use logistic regression to learn the coefficients, use a weighted sceheme that penalizes outputs which are wrong. For our weighting scheme we apply the following algorithm

- Let $C =< c_1, ..., c_m >$
- $c_j = \frac{1}{N} \sum_{i=1}^{N} \frac{x_{ij}}{(x_{ij} - y_i)^2 + 0.00001}$
- Normalize C