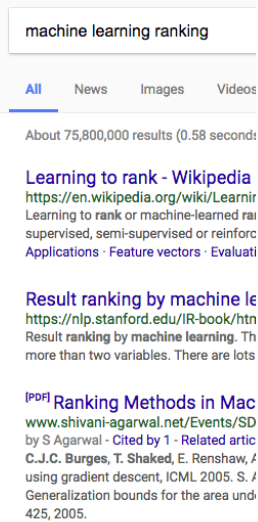
Classification

Jordan Boyd-Graber
University of Maryland
RANKING

Roadmap

- Combining rankings: taking advantage of multiple weak rankers
- Maximum margin ranking: support vector machines
- Reduction to classification: optimizing

Ranking



Given input $x_1 \dots x_r$, return permutation of $[r]$. Permutation often parameterized by vector of scalars $y_1 \dots y_r$.

- Web search (Google used > 200 features)
- Movie rankings
- Dating

What's the goal? Loss functions ...

- Kendall-τ

$$L(y', y) = \frac{2}{r(r-1)} \sum_i \sum_j \mathbb{1}\left[\text{sign}(y'_i - y'_j) \neq (y_i - y_j)\right] \tag{1}$$

What's the goal? Loss functions ...

- Kendall-τ

$$L(y', y) = \frac{2}{r(r-1)} \sum_i \sum_j \mathbb{1}\left[\text{sign}(y'_i - y'_j) \neq (y_i - y_j)\right] \tag{1}$$

- Normalized Discounted Cumulative Gain:

$$D(i) = \frac{1}{\lg(r - i + 2)} \text{if } i \in \{r - k + 1, \dots, r\} \tag{2}$$

$$G(y', y) = \sum_i D(\pi(y')_i) y_i \tag{3}$$

$$\tag{4}$$

Discount function: focus on top k elements in list

What's the goal? Loss functions . . .

- Kendall-τ

$$L(y', y) = \frac{2}{r(r-1)} \sum_i \sum_j \mathbb{1}\left[\text{sign}(y'_i - y'_j) \neq (y_i - y_j)\right] \quad (1)$$

- Normalized Discounted Cumulative Gain:

$$D(i) = \frac{1}{\lg(r - i + 2)} \text{if } i \in \{r - k + 1, \ldots, r\} \quad (2)$$

$$G(y', y) = \sum_i D(\pi(y')_i) y_i \quad (3)$$

$$(4)$$

Gain function: weight examples based on whether they are in important part of list, defined by permutation π. For example, for $r = 5$, the vector $y = (2, 1, 6, 1, 0.5)$ induces the permutation $\pi(y) = (4, 3, 5, 1, 2)$

What's the goal? Loss functions ...

- Kendall-τ

$$L(y', y) = \frac{2}{r(r-1)} \sum_i \sum_j \mathbb{1}\left[\text{sign}(y'_i - y'_j) \neq (y_i - y_j)\right] \tag{1}$$

- Normalized Discounted Cumulative Gain:

$$D(i) = \frac{1}{\lg(r - i + 2)} \text{if } i \in \{r - k + 1, \dots, r\} \tag{2}$$

$$G(y', y) = \sum_i D(\pi(y')_i) y_i \tag{3}$$

$$L(y', y) = \sum_i \frac{1}{G(y, y)} \sum_i (D(\pi(y)_i) - D(\pi(y')_i)) y_i \tag{4}$$

$$\tag{5}$$

Loss function focuses on how wrong top of list is

Examples as feature vectors

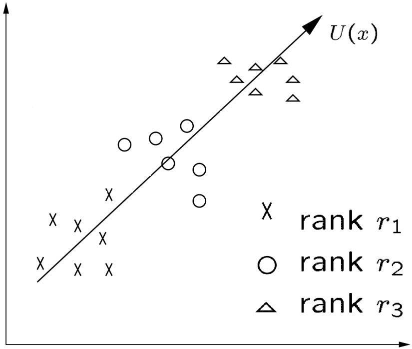
Every example has a feature vector $f(x)$

example	docID	query	cosine score	ω	judgment
Φ_1	37	linux operating system	0.032	3	*relevant*
Φ_2	37	penguin logo	0.02	4	*nonrelevant*
Φ_3	238	operating system	0.043	2	*relevant*
Φ_4	238	runtime environment	0.004	2	*nonrelevant*
Φ_5	1741	kernel layer	0.022	3	*relevant*
Φ_6	2094	device driver	0.03	2	*relevant*
Φ_7	3191	device driver	0.027	5	*nonrelevant*

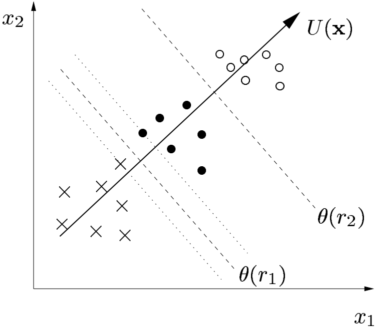
Turning features to rank

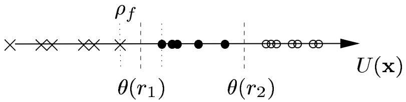
- Have a series of "levels" or ranks $y = 1\ldots$
- We want to find a function to separate examples

$$f(x) \equiv \langle w \cdot \phi(x) \rangle \tag{6}$$



Maximizing the margin





Recap

- Ranking is an important problem
- Different objective function
- Implementation similar to regression