Active query synthesis for preference learning

Georgia Tech College of Engineering
School of Electrical and Computer Engineering

Namrata Nadagouda, Mark A. Davenport

NEURAL INFORMATION PROCESSING SYSTEMS

Active learning

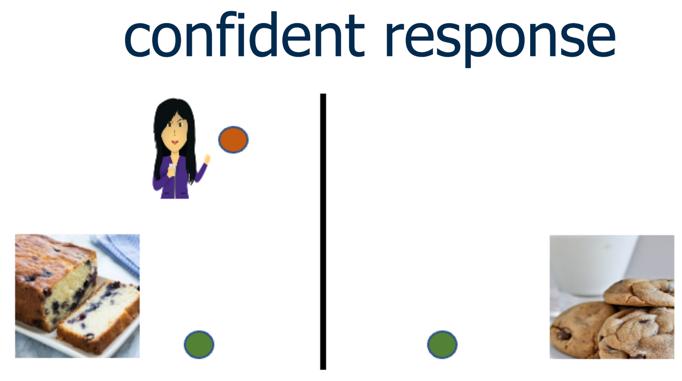
- Involves **adaptively** querying labels for the most informative data points to improve **label efficiency**
- Most querying strategies are often **computationally expensive** and usually involve optimization over a **discrete** space which leads to a **lower accuracy**
- Instead, we can directly **synthesize** the most informative samples which can be very **efficient** and also optimize over a **continuous** space which can lead to a **better model performance**
- If the problem is constrained by the available dataset - **How best to approximate the optimal pair of synthesized points?**

Learning human preferences

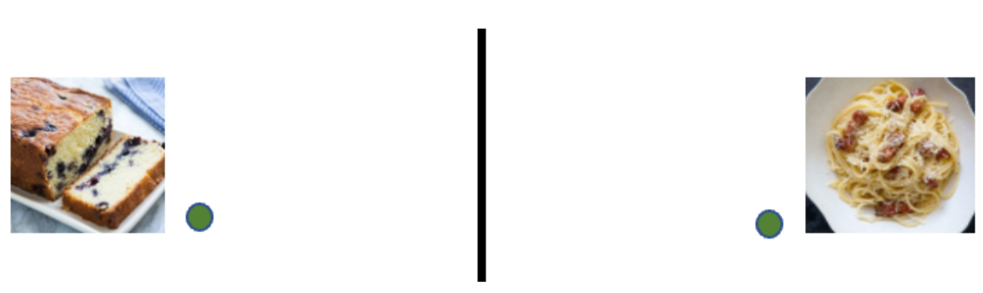
- Estimate a preference vector $w \in \mathbb{R}^d$ based on responses to preferences over pairs of items
- Responses can have different levels of associated confidence

Items too close

Ideal distance - confident response



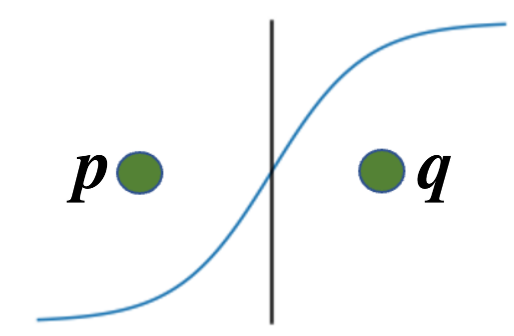
Response - not confident



Items too far

Query response model

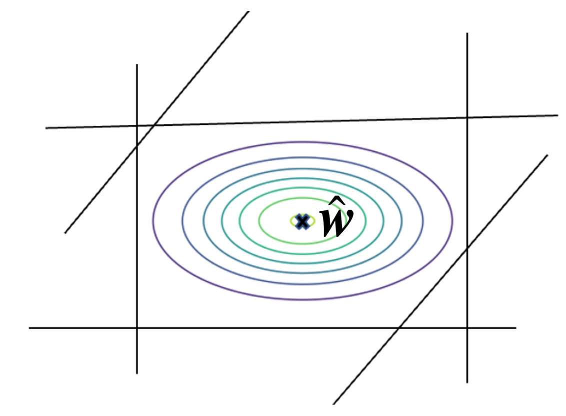
$$P(\boldsymbol{p} \prec \boldsymbol{q}) = \frac{1}{1 + e^{-\left(h(\|\boldsymbol{a}_{pq}\|)(\boldsymbol{a}_{pq}^\top \boldsymbol{w} - \tau_{pq})\right)}}$$



$(\boldsymbol{a}_{pq}, \tau_{pq})$ represents the bisecting hyperplane

$h(\|\boldsymbol{a}_{pq}\|)$ is the response confidence function

User estimated as the mean of the posterior distribution

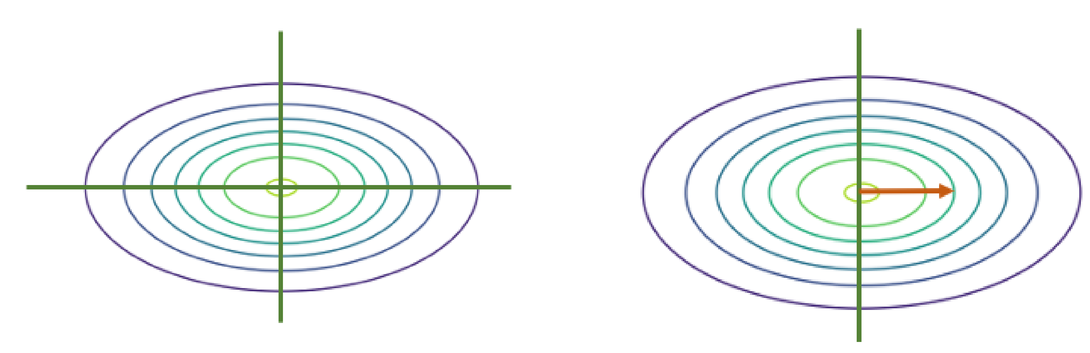


Query synthesis

Maximization of mutual information

$$I\left(Y; W \mid (\boldsymbol{a}, \tau)\right) = H\left(Y \mid (\boldsymbol{a}, \tau)\right) - \underset{W}{\mathbb{E}}\left[H(Y \mid W, (\boldsymbol{a}, \tau))\right]$$

First compute the optimal hyperplane

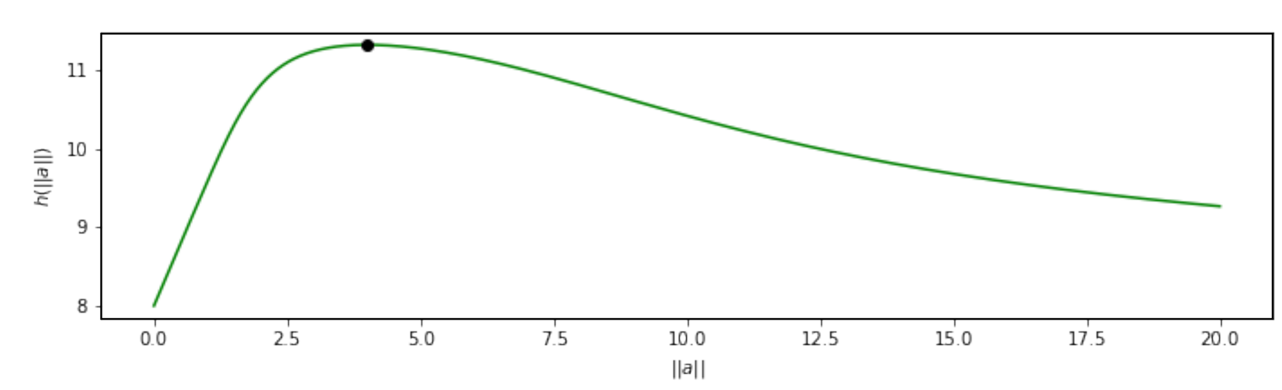


Equi-probable responses

Direction of maximum variance

Response confidence function

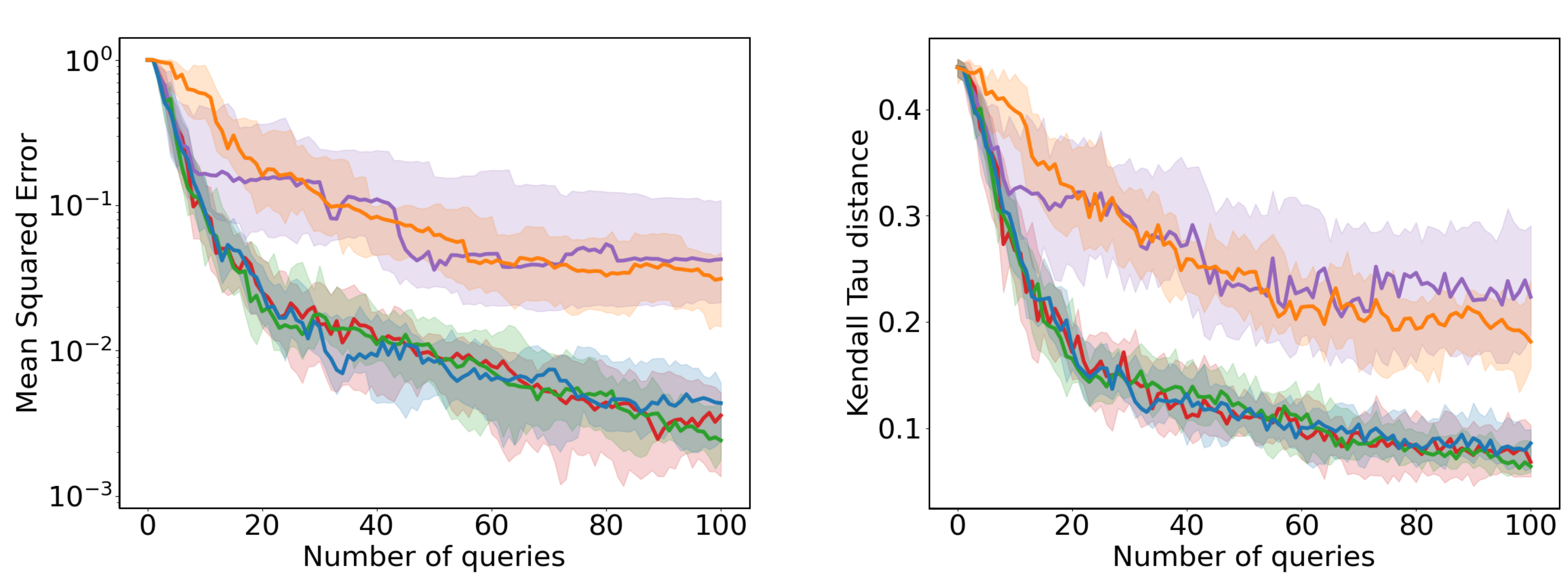
Then compute the optimal pair of points



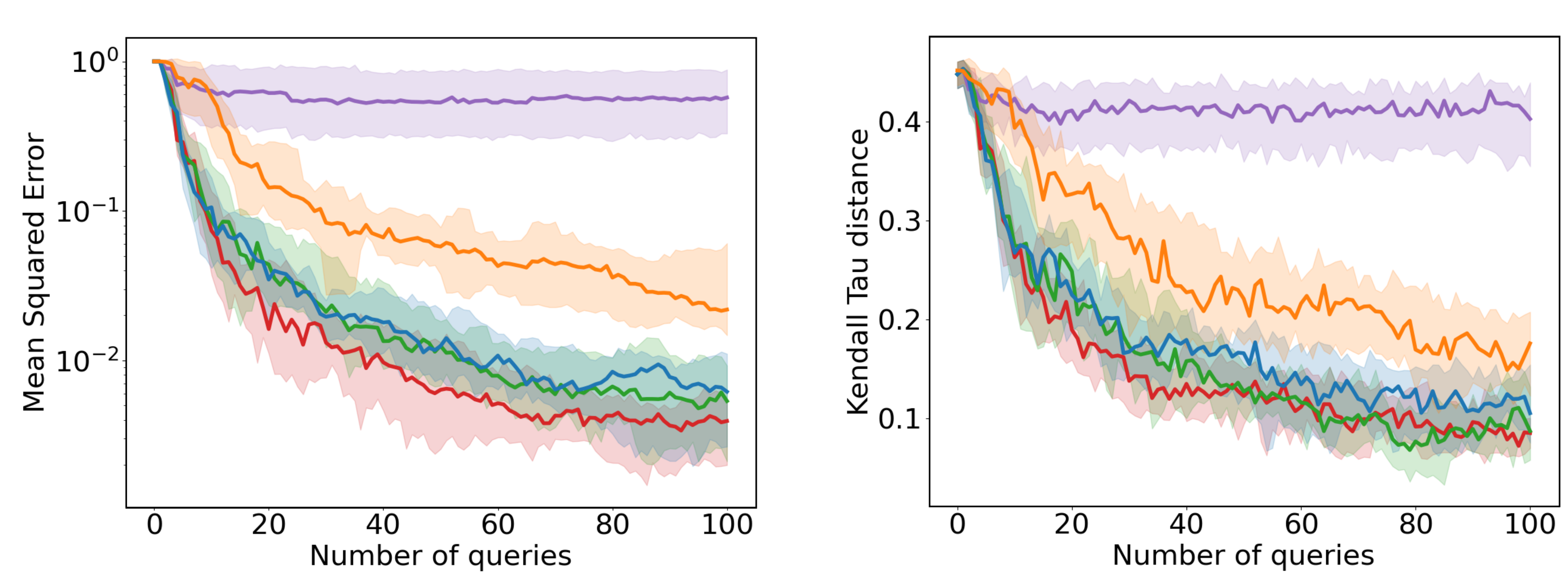
Select $(\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{q}})$ such that $\boldsymbol{a}_{\tilde{p}\tilde{q}} = \arg\max h(\|\boldsymbol{a}\|)$

Experiment results

Simulated dataset - Items $\mathscr{D} \sim \mathrm{U}[-4, 4]^4$, user $\mathrm{W} \sim \mathrm{U}[-1, 1]^4$, number of items N



N = 1000



N = 25

- Active synthesized queries
- Approximation I queries
- Approximation II queries
- Active discrete queries
- Random queries

Approximation methods

- Approximation I

$$\boldsymbol{p}_1 = \underset{\boldsymbol{p} \in \mathscr{D}}{\arg\min} \|\tilde{\boldsymbol{p}} - \boldsymbol{p}\|$$

$$\boldsymbol{q}_2 = \underset{\boldsymbol{q} \in \mathscr{D}}{\arg\min} \|\tilde{\boldsymbol{q}} - \boldsymbol{q}\|$$

- Approximation II

Let $\mathscr{P} = kNN(\tilde{\boldsymbol{p}})$ and $\mathscr{Q} = kNN(\tilde{\boldsymbol{q}})$

$$(\boldsymbol{p}_2, \boldsymbol{q}_2) = \underset{p^* \in \mathscr{P},\, q^* \in \mathscr{Q}}{\arg\max} I\left(Y; W \mid (\boldsymbol{p}^*, \boldsymbol{q}^*)\right)$$

Result metrics

- Mean Squared Error - MSE$(\boldsymbol{w}, \hat{\boldsymbol{w}})$
- Kendall Tau distance - measures dissimilarity between rankings of items w.r.t $\boldsymbol{w}$ and $\hat{\boldsymbol{w}}$

Results

- Method II achieves a much better performance than method I
- Performance of II deteriorates significantly as the number of items decreases

Next steps

- Analyze the deviation of Approx. I and Approx. II from the optimality criteria
- Conduct experiments with real world data