# Optimal Exploration is no harder than Thompson Sampling

Zhaoqi Li

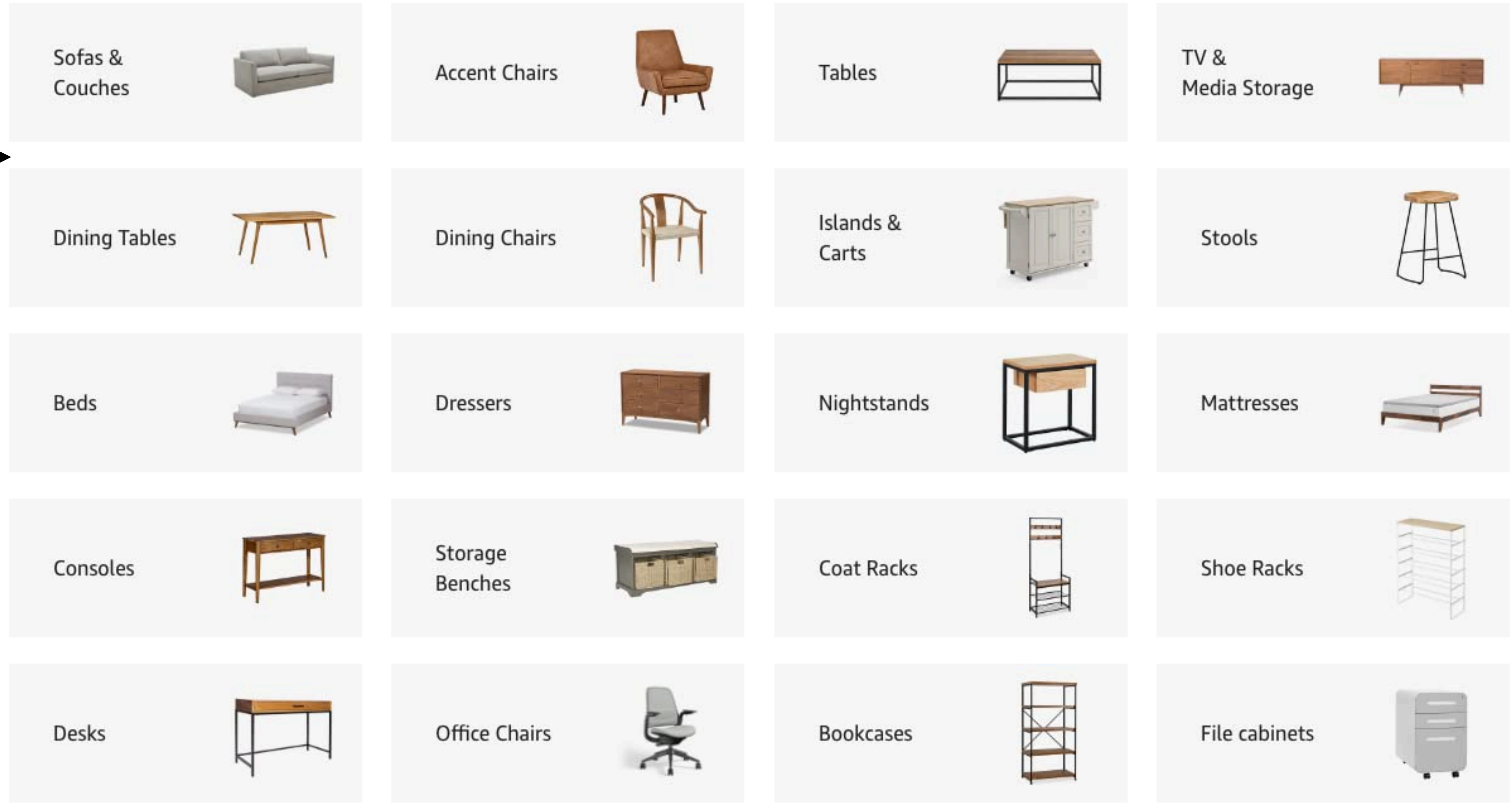
Joint work with Lalit Jain and Kevin Jamieson

Submitted to AISTATS 2024

# Motivation: Item Recommendation

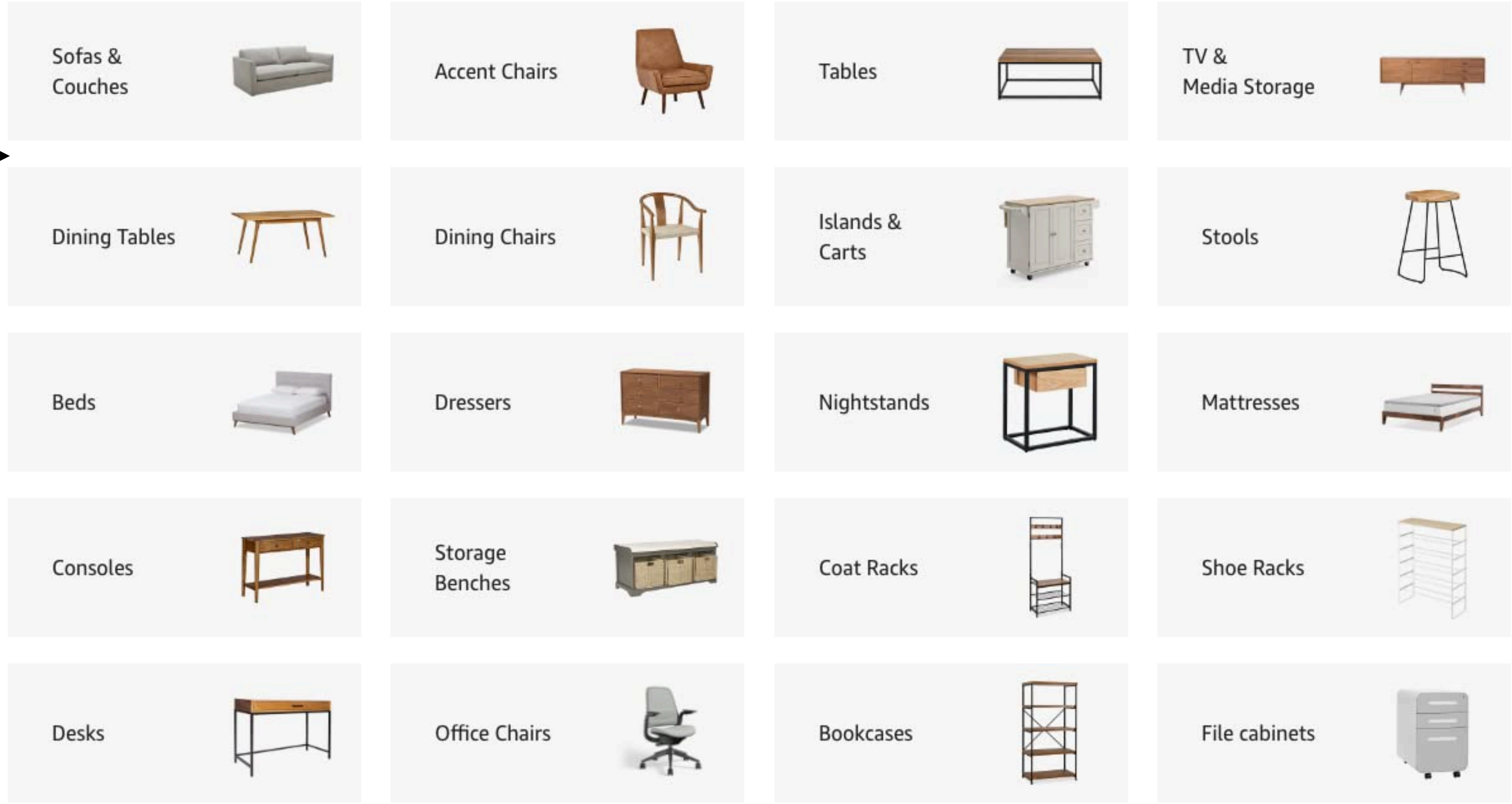
# Motivation: Item Recommendation

**All Furniture**

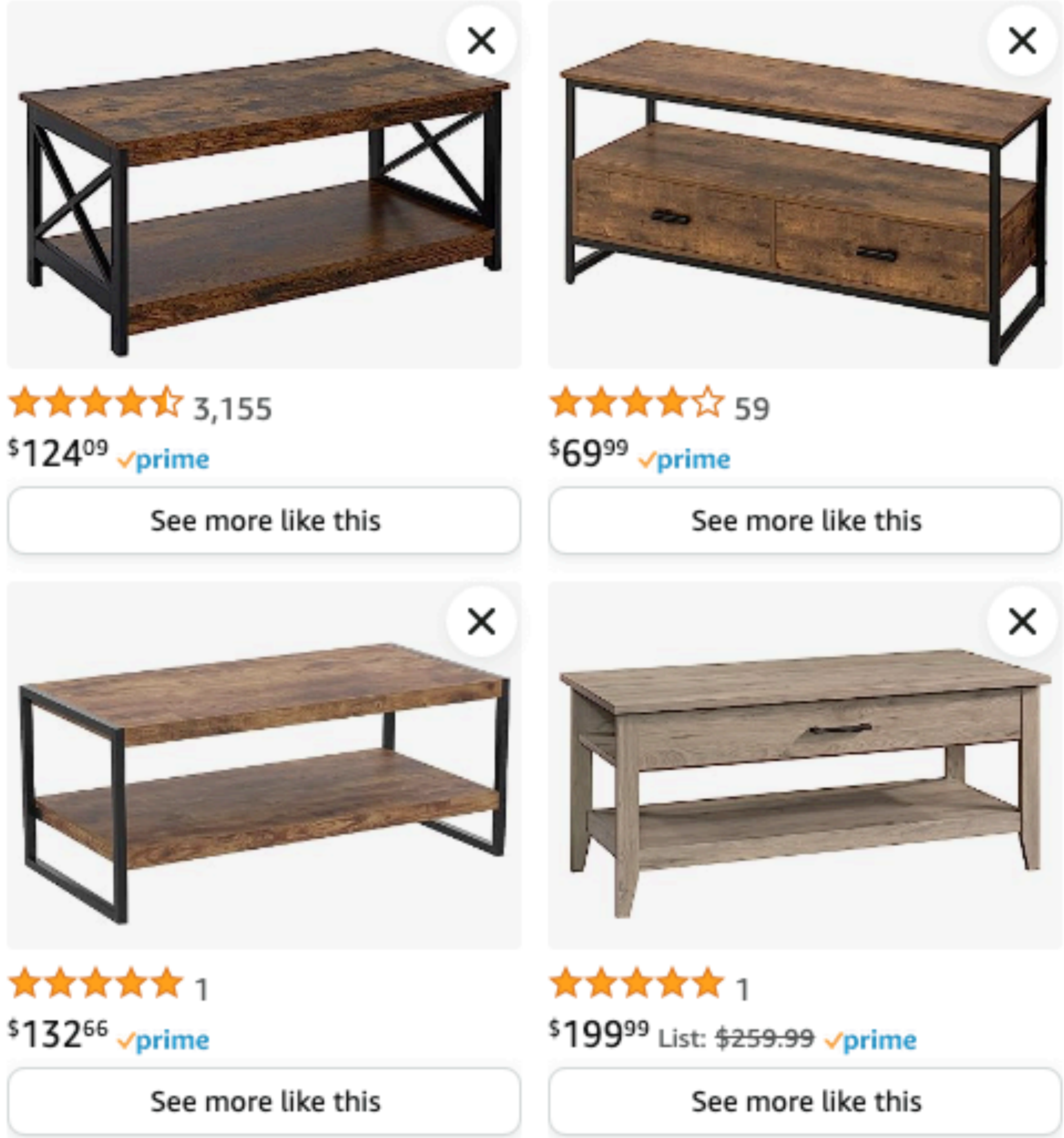


# Motivation: Item Recommendation

**All Furniture**

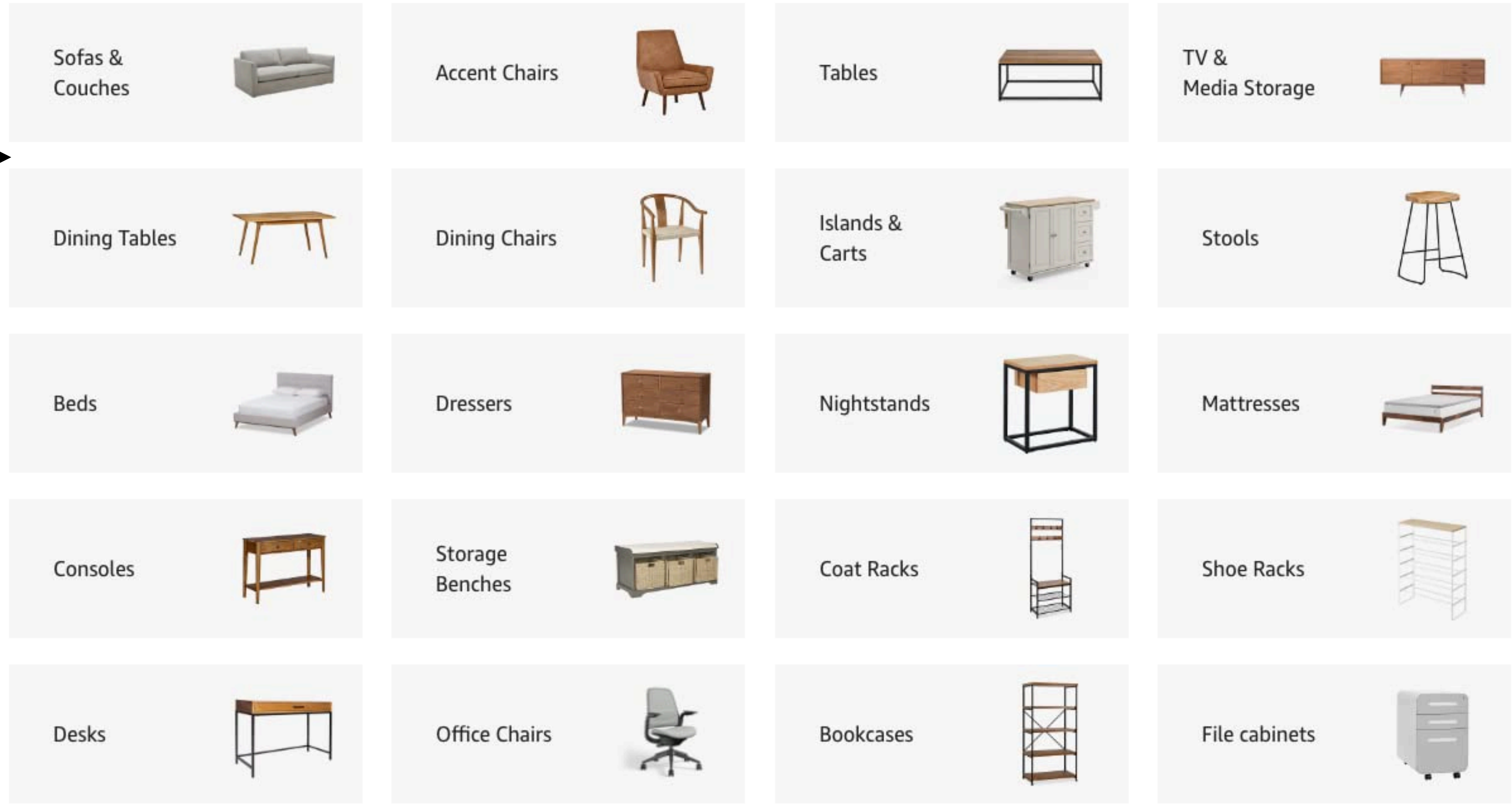



**Tables**

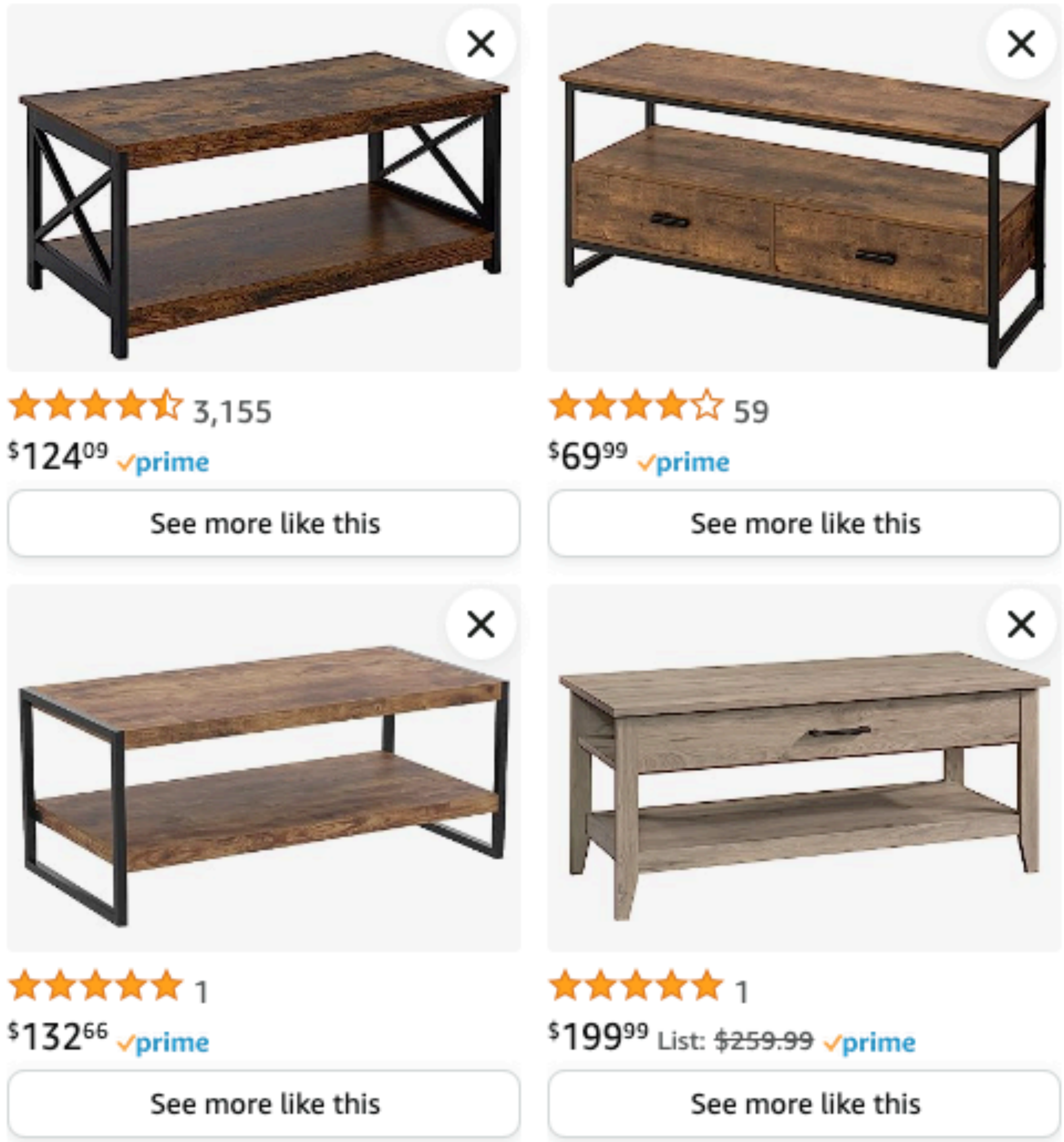


# Motivation: Item Recommendation

**All Furniture**




**Tables**

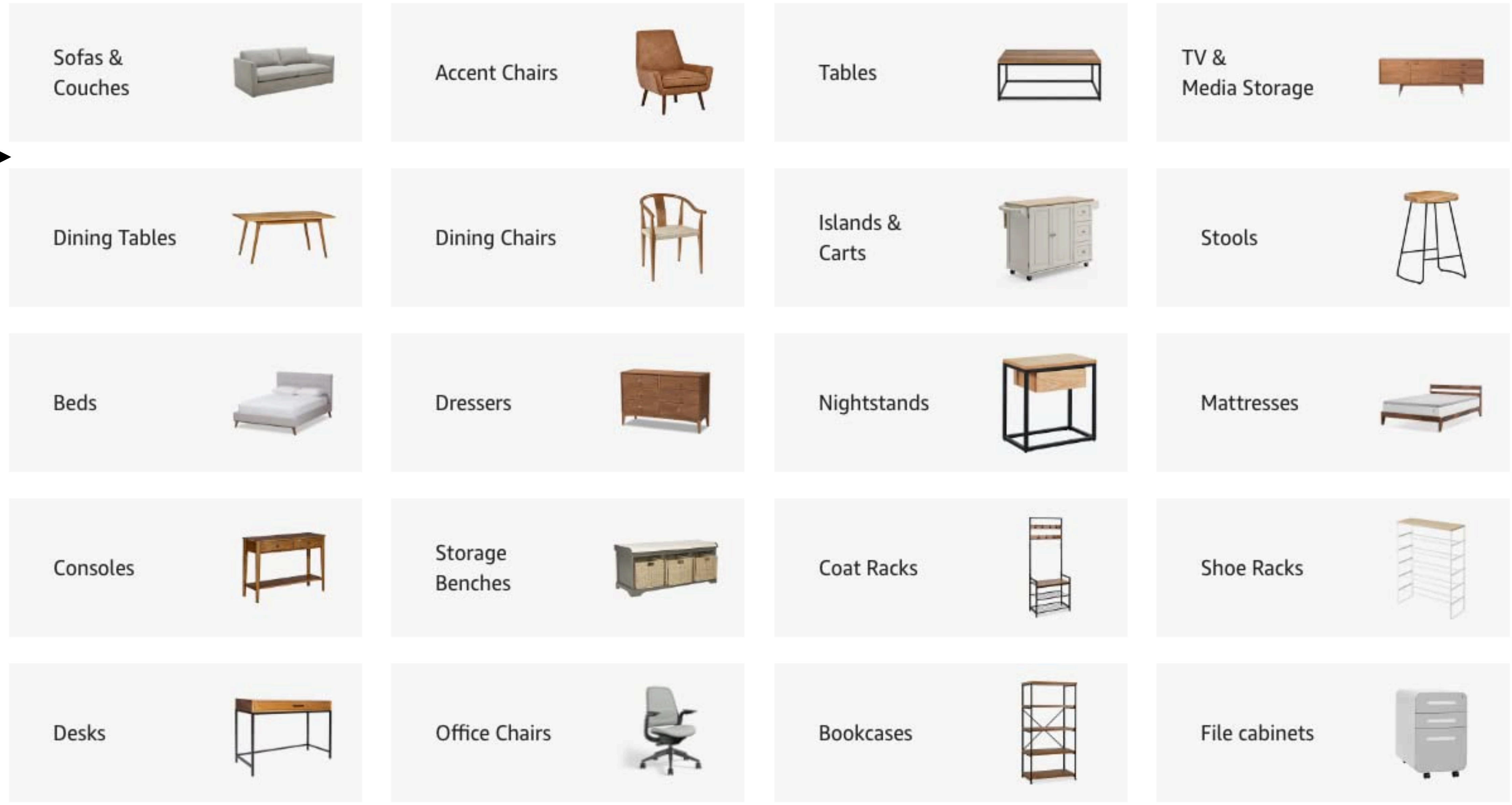



**Find customer's favorite furniture given data from tables!**

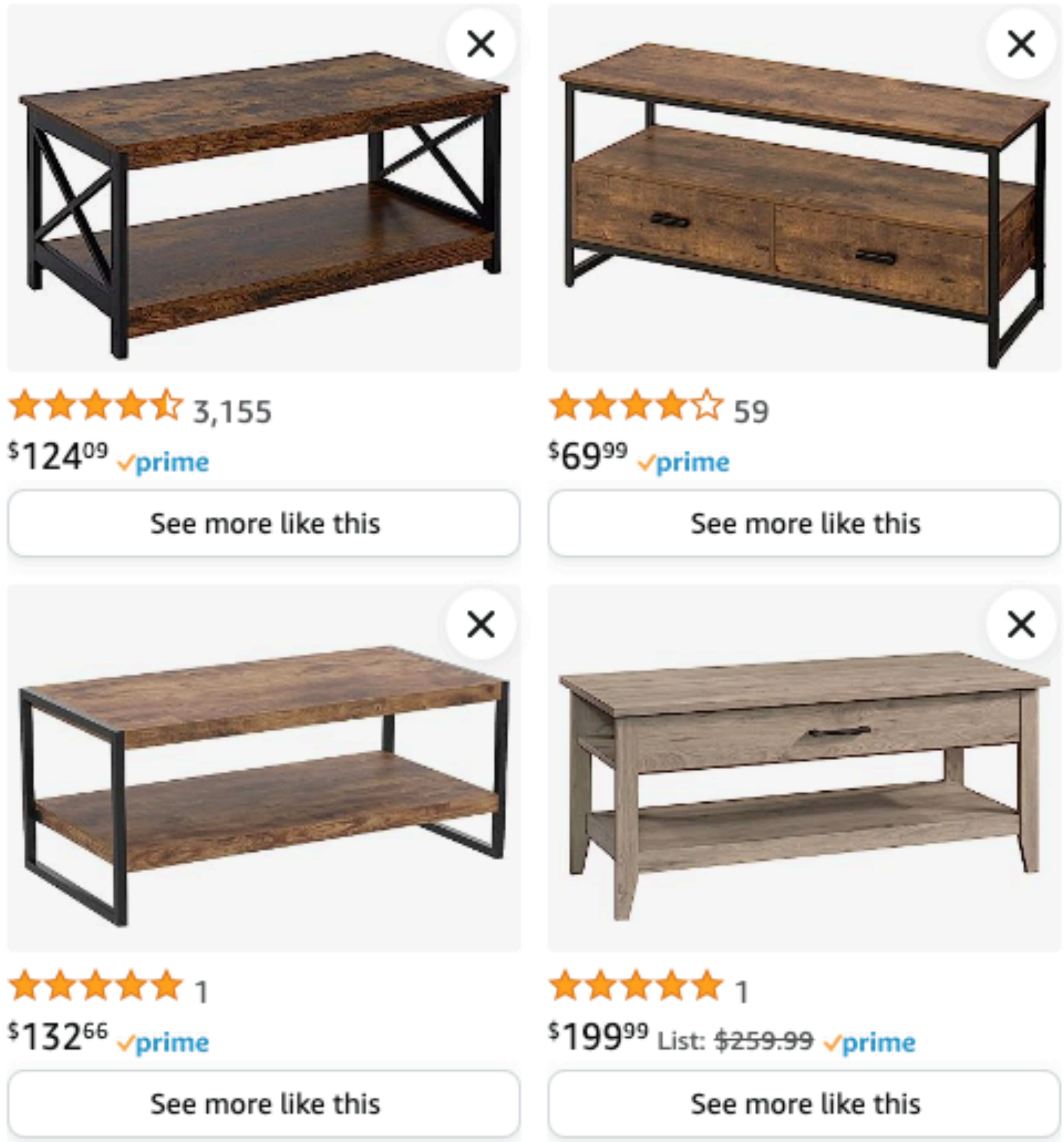
# Motivation: Item Recommendation

$Z \subset \mathbb{R}^d$

**All Furniture**



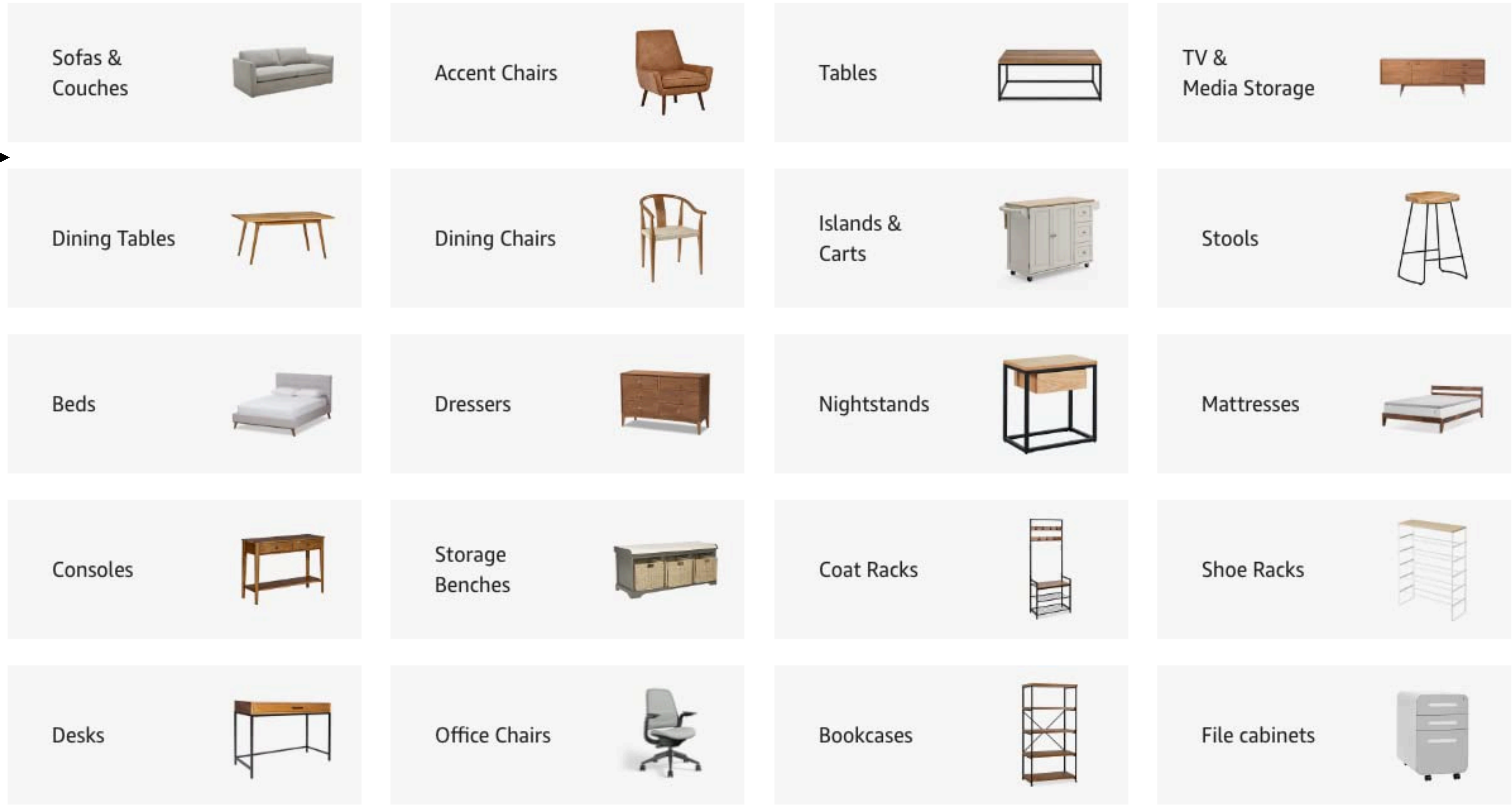

**Tables**



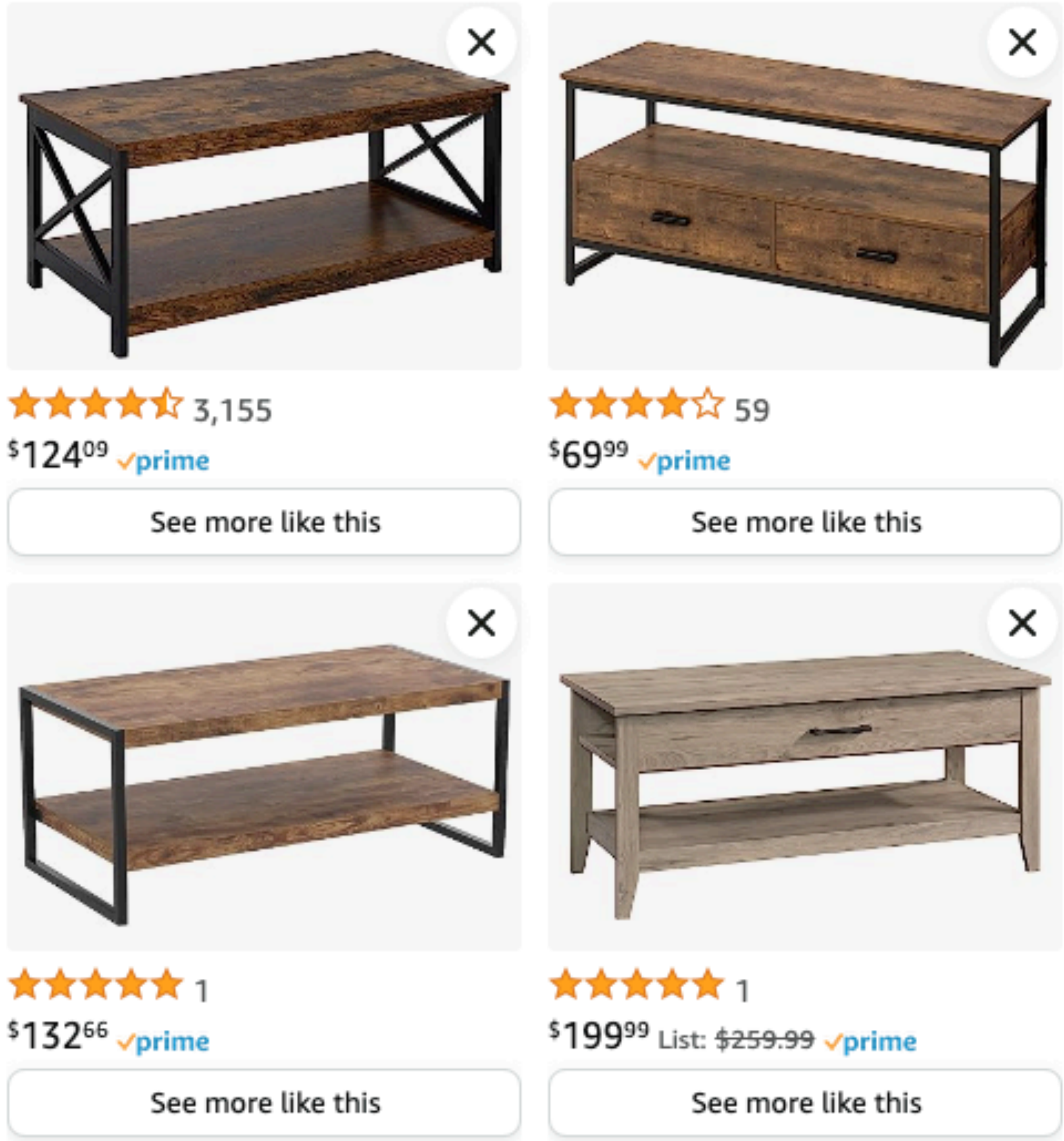

**Find customer's favorite furniture given data from tables!**

# Motivation: Item Recommendation

$Z \subset \mathbb{R}^d$ **All Furniture**




$X \subset Z \subset \mathbb{R}^d$ **Tables**

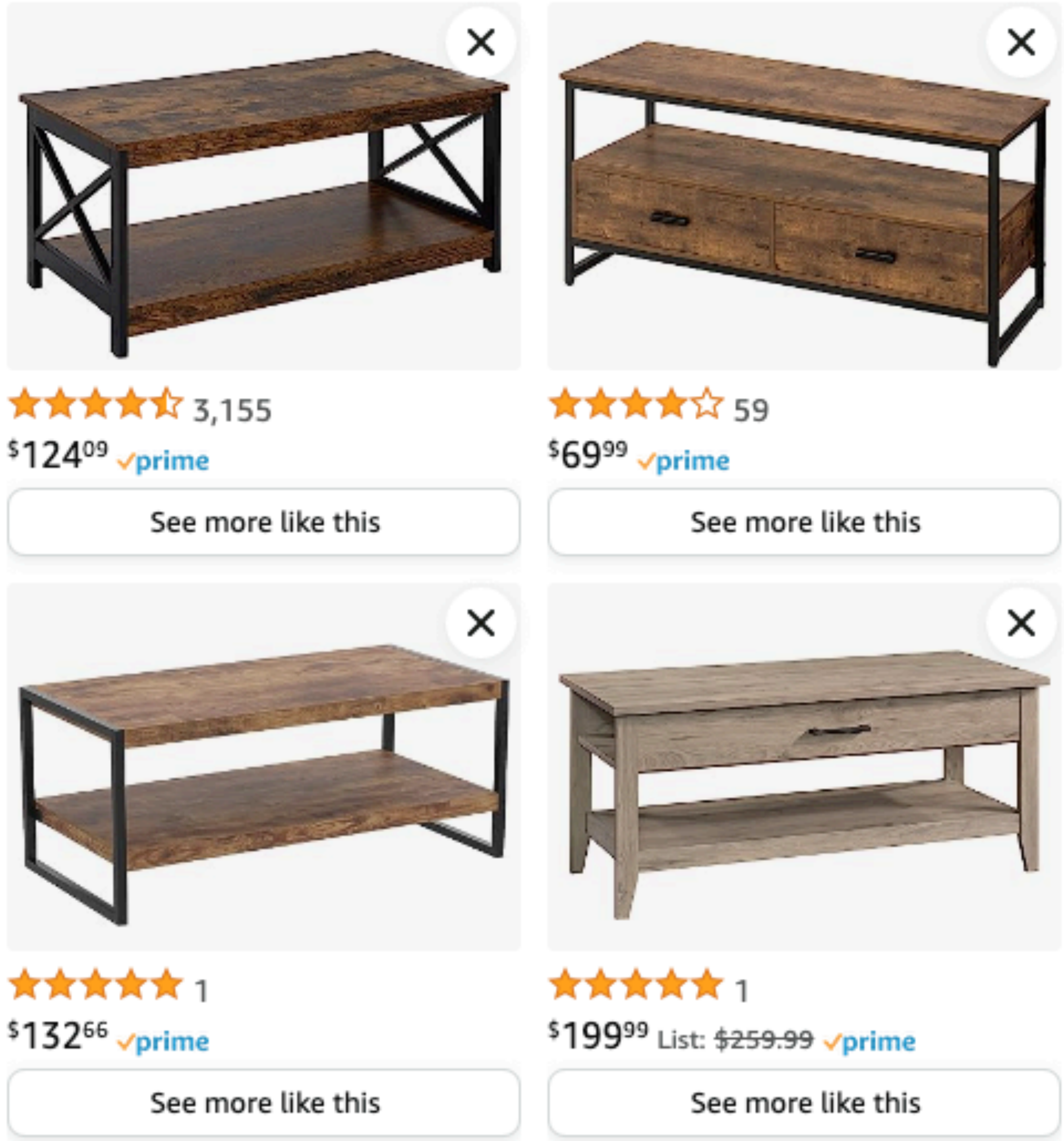


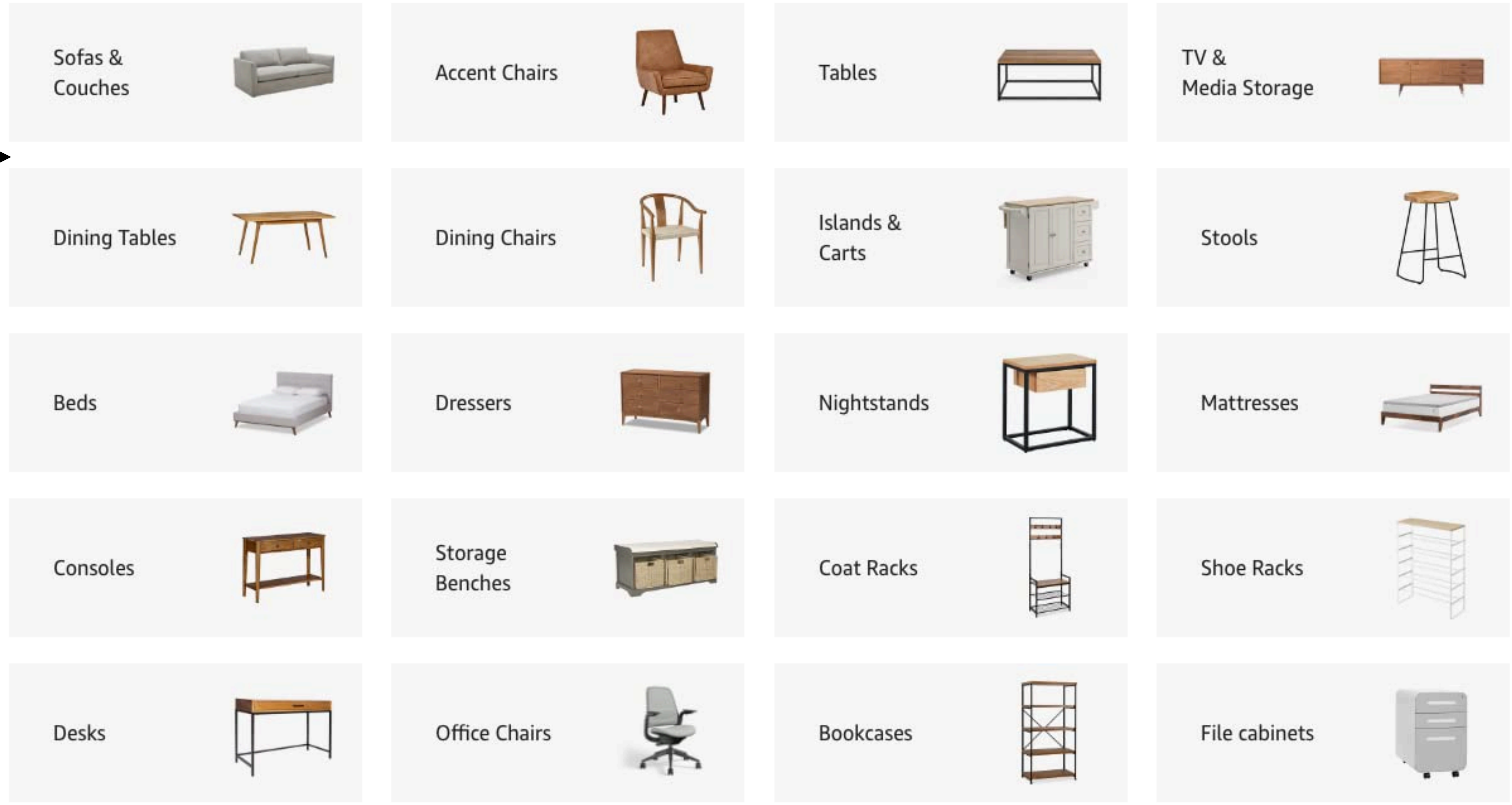

**Find customer's favorite furniture given data from tables!**

# Motivation: Item Recommendation

$Z \subset \mathbb{R}^d$ **All Furniture**

$X \subset Z \subset \mathbb{R}^d$ **Tables**



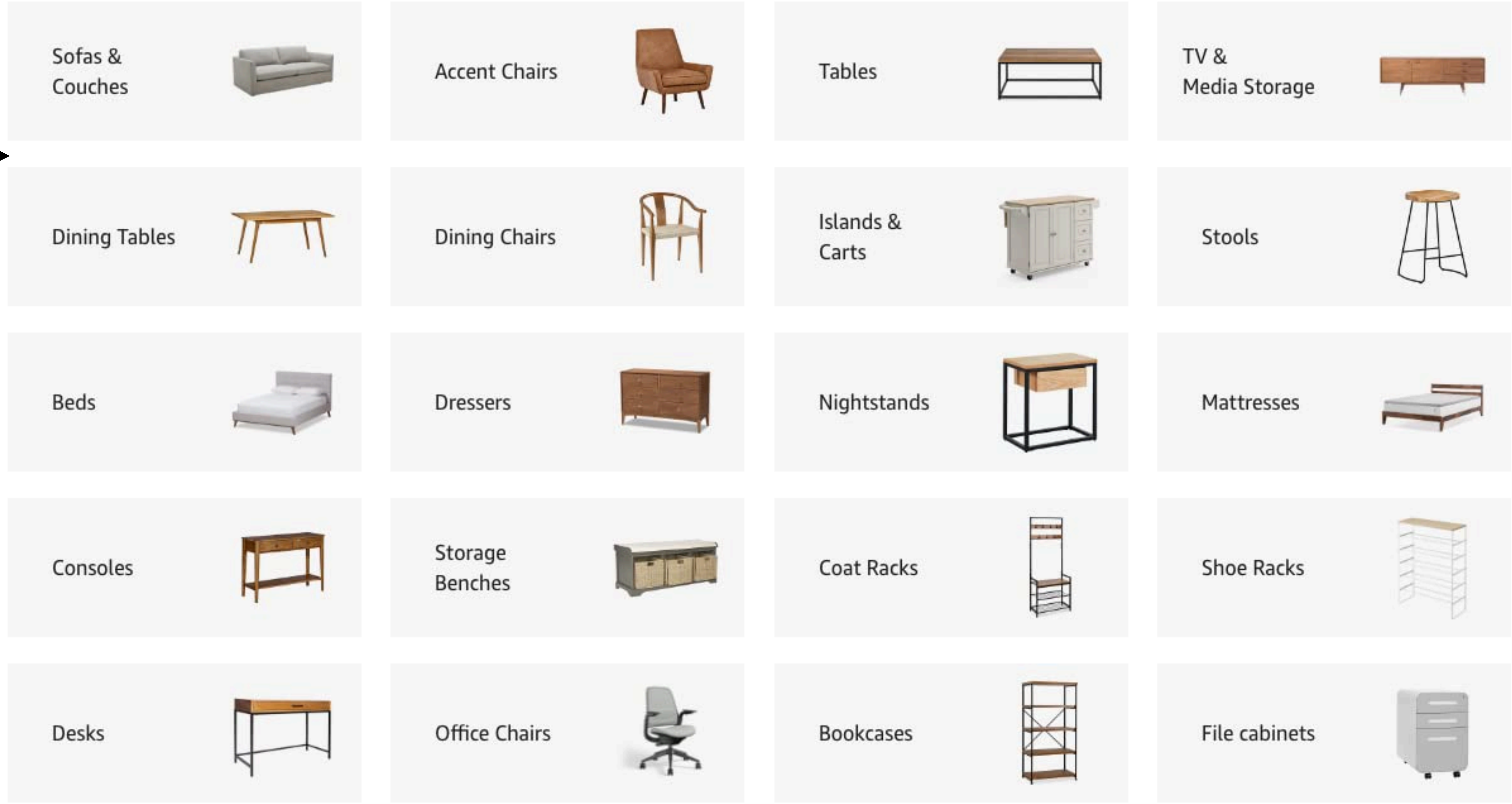




$\theta_\star \in \mathbb{R}^d$

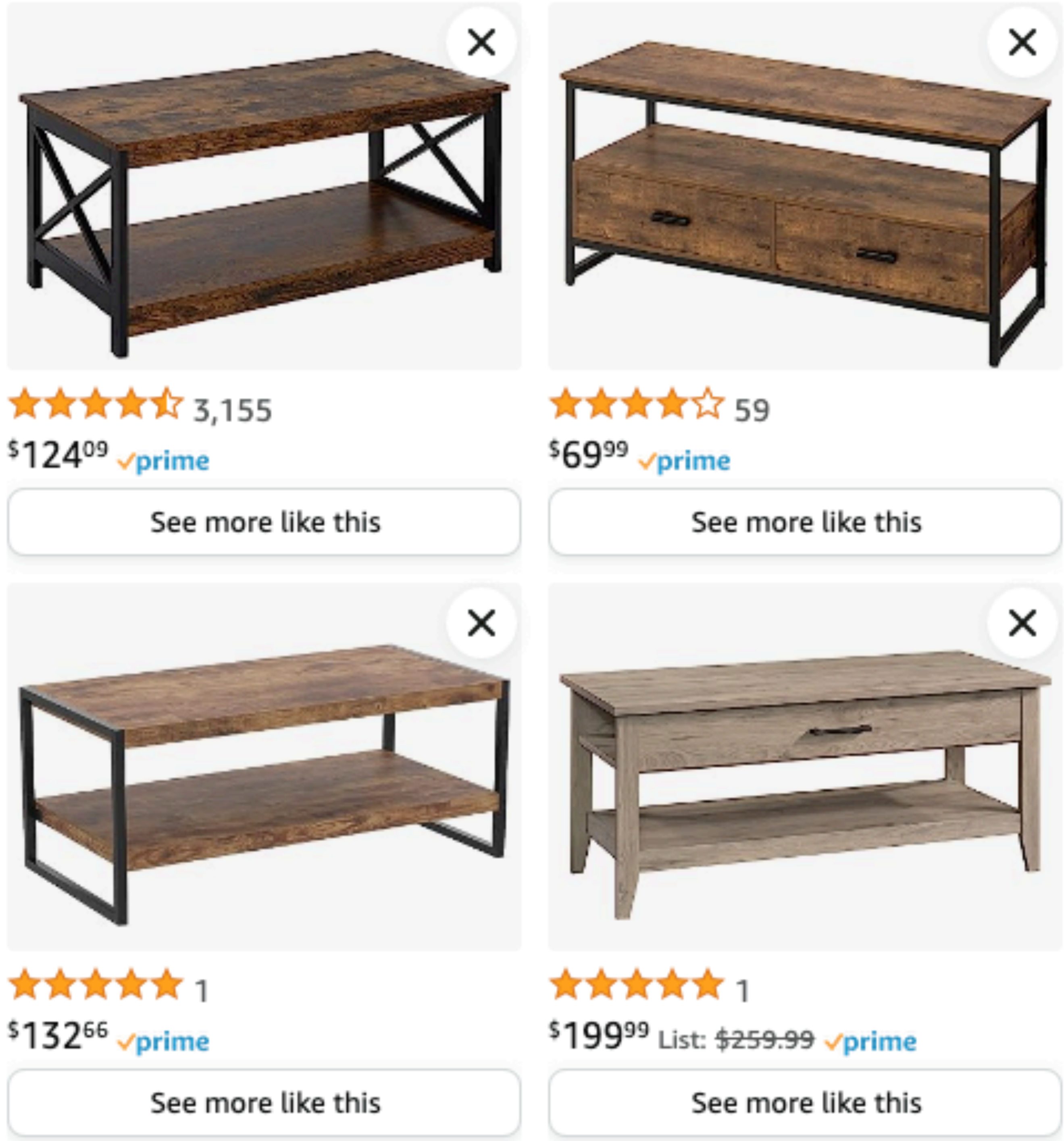
**Find customer's favorite furniture given data from tables!**

# Motivation: Item Recommendation

$Z \subset \mathbb{R}^d$ **All Furniture**




$X \subset Z \subset \mathbb{R}^d$ **Tables**




$\theta_\star \in \mathbb{R}^d$ **Preferences**

**Find customer's favorite furniture given data from tables!**

# Best-Arm Identification in Transductive Linear Bandits

# Best-Arm Identification in Transductive Linear Bandits

**Measurement Arms:** $\mathcal{X} \subset \mathbb{R}^d$

**Item Arms:** $\mathcal{Z} \subset \mathbb{R}^d$

$\mathcal{X} \neq \mathcal{Z}$

**Unknown Parameter:** $\theta_\star \in \Theta \subset \mathbb{R}^d$

# Best-Arm Identification in Transductive Linear Bandits

**Measurement Arms:** $\mathcal{X} \subset \mathbb{R}^d$

**Item Arms:** $\mathcal{Z} \subset \mathbb{R}^d$

**Unknown Parameter:** $\theta_\star \in \Theta \subset \mathbb{R}^d$

$\mathcal{X} \neq \mathcal{Z}$

**Input:** $\mathcal{X}, \mathcal{Z} \subset \mathbb{R}^d$

**for** $t = 1,2,\ldots$

1. Learner chooses $x_t \in \mathcal{X}$
2. Nature reveals $y_t$

# Best-Arm Identification in Transductive Linear Bandits

**Measurement Arms:** $\mathcal{X} \subset \mathbb{R}^d$

**Item Arms:** $\mathcal{Z} \subset \mathbb{R}^d$

$\mathcal{X} \neq \mathcal{Z}$

**Unknown Parameter:** $\theta_\star \in \Theta \subset \mathbb{R}^d$

**Input:** $\mathcal{X}, \mathcal{Z} \subset \mathbb{R}^d$

**for** $t = 1,2,\ldots$

1. Learner chooses $x_t \in \mathcal{X}$
2. Nature reveals $y_t$

$$y_t = \langle x_t, \theta_\star \rangle + \epsilon_t, \epsilon_t \sim N(0,1)$$

# Best-Arm Identification in Transductive Linear Bandits

**Measurement Arms:** $\mathcal{X} \subset \mathbb{R}^d$

**Item Arms:** $\mathcal{Z} \subset \mathbb{R}^d$

$\mathcal{X} \neq \mathcal{Z}$

**Unknown Parameter:** $\theta_\star \in \Theta \subset \mathbb{R}^d$

**Input:** $\mathcal{X}, \mathcal{Z} \subset \mathbb{R}^d$

**for** $t = 1,2,\ldots$

1. Learner chooses $x_t \in \mathcal{X}$
2. Nature reveals $y_t$

$$y_t = \langle x_t, \theta_\star \rangle + \epsilon_t, \epsilon_t \sim N(0,1)$$

Gaussian Noise

# Best-Arm Identification in Transductive Linear Bandits

**Measurement Arms:** $\mathcal{X} \subset \mathbb{R}^d$

**Item Arms:** $\mathcal{Z} \subset \mathbb{R}^d$

$\mathcal{X} \neq \mathcal{Z}$

**Unknown Parameter:** $\theta_\star \in \Theta \subset \mathbb{R}^d$

**Input:** $\mathcal{X}, \mathcal{Z} \subset \mathbb{R}^d$

**for** $t = 1,2,\ldots$

1. Learner chooses $x_t \in \mathcal{X}$
2. Nature reveals $y_t$

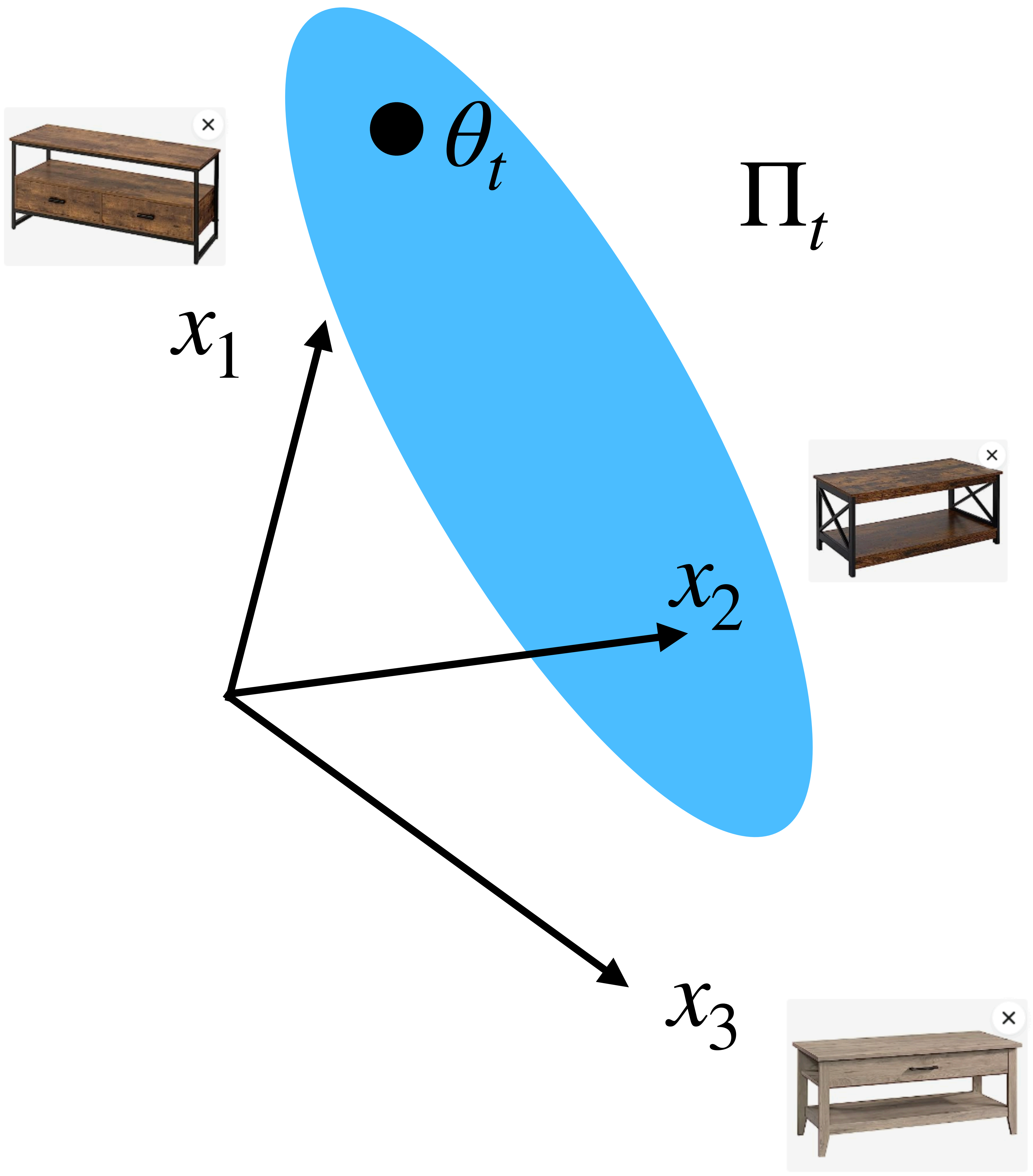
$$y_t = \langle x_t, \theta_\star \rangle + \epsilon_t, \epsilon_t \sim N(0,1)$$

Gaussian Noise

**Best-Arm Identification Problem**

Given $\mathcal{X}, \mathcal{Z} \subset \mathbb{R}^d$ and unknown $\theta_\star \in \mathbb{R}^d$ identify $z_\star := \arg\max_{z \in \mathcal{Z}} \langle z, \theta_\star \rangle$ with high probability as quickly as possible

# Prior Art: Thompson Sampling

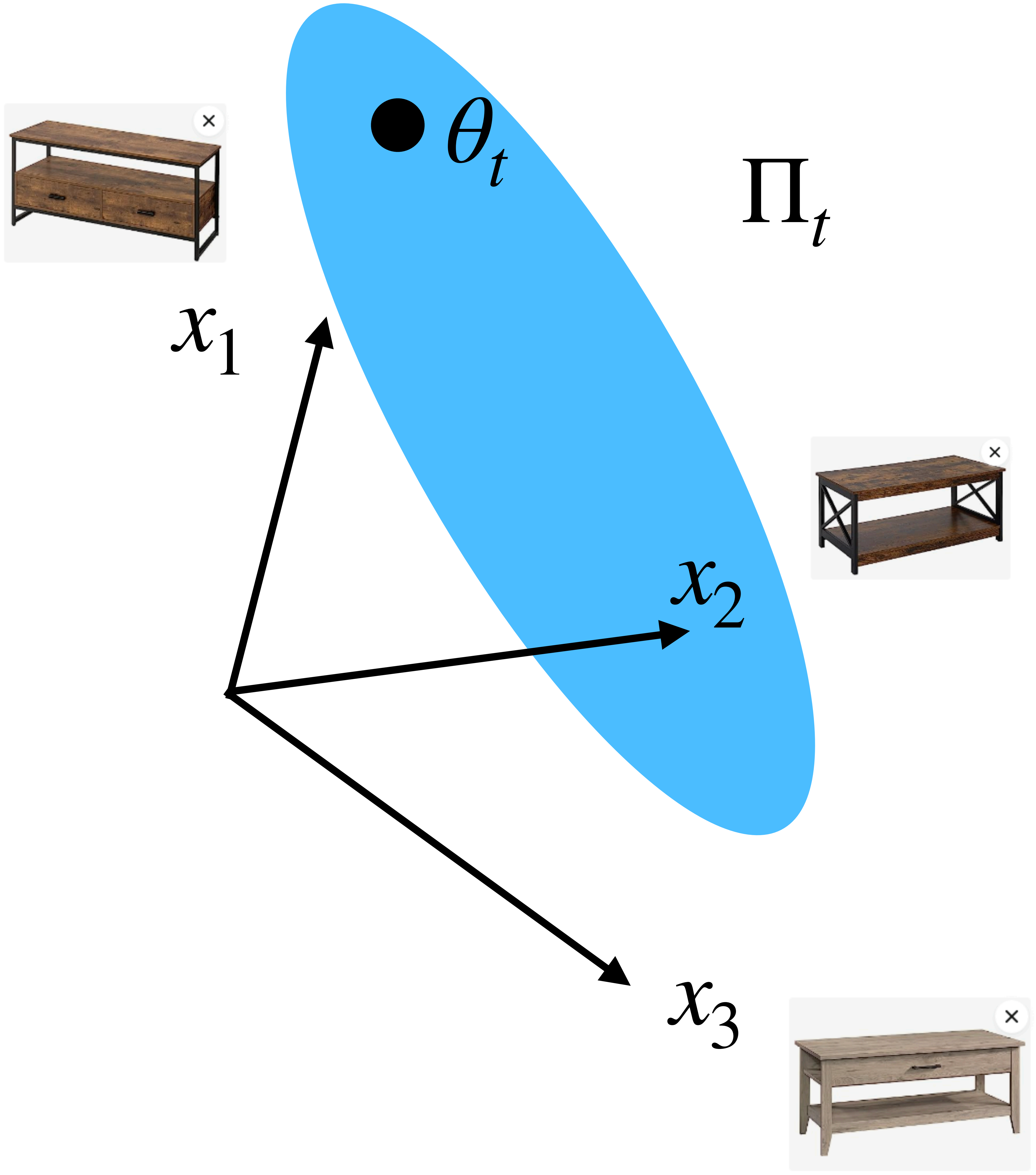


# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} = \mathcal{Z} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0, I)$

**for** $t = 1, 2, \ldots$

1. Sample $\theta_t \sim \Pi_t$



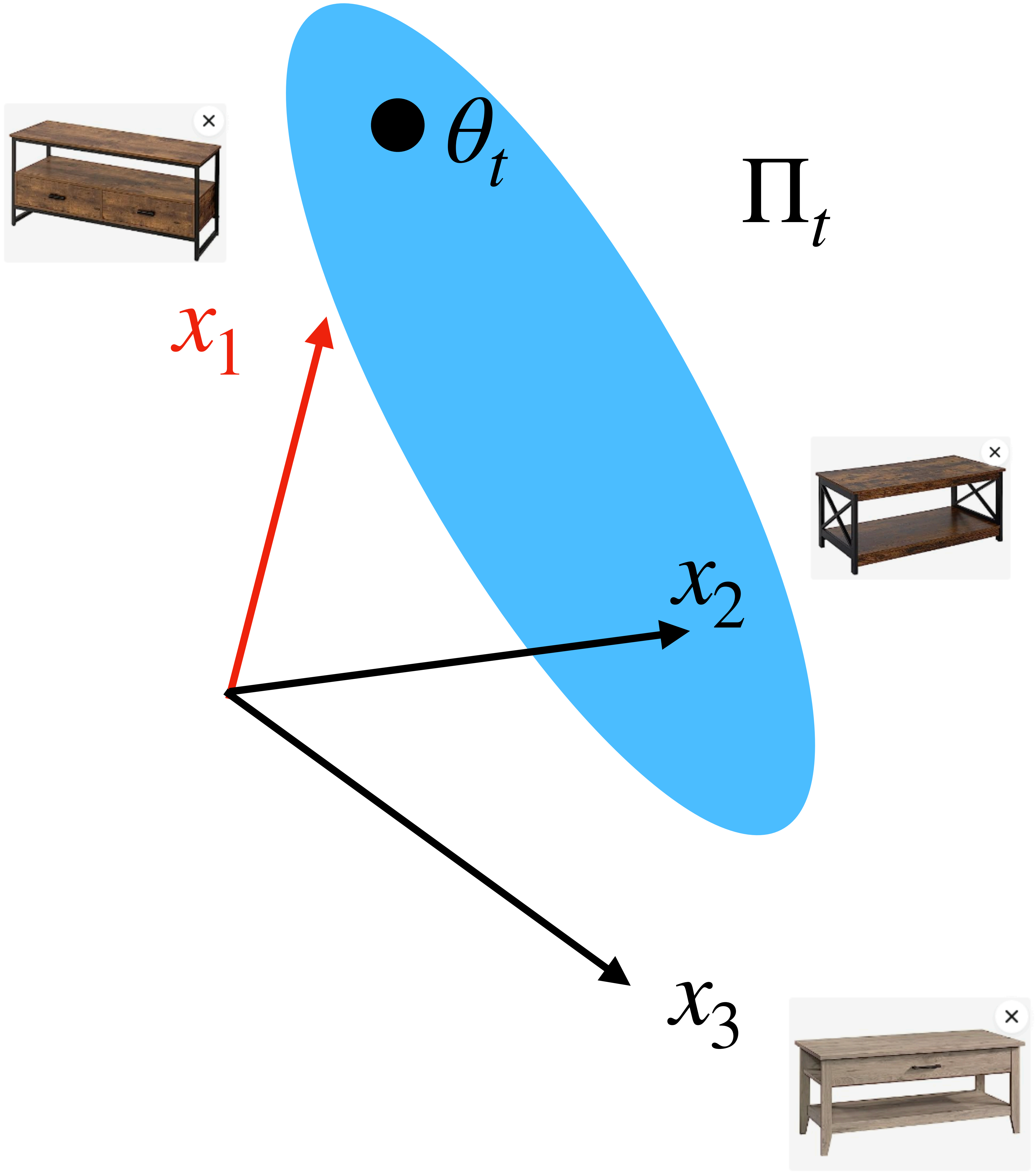
# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0, I)$

**for** $t = 1, 2, \ldots$

1. Sample $\theta_t \sim \Pi_t$

2. Play $x_t = \arg\max_{x \in \mathcal{X}} \theta_t^\top x$

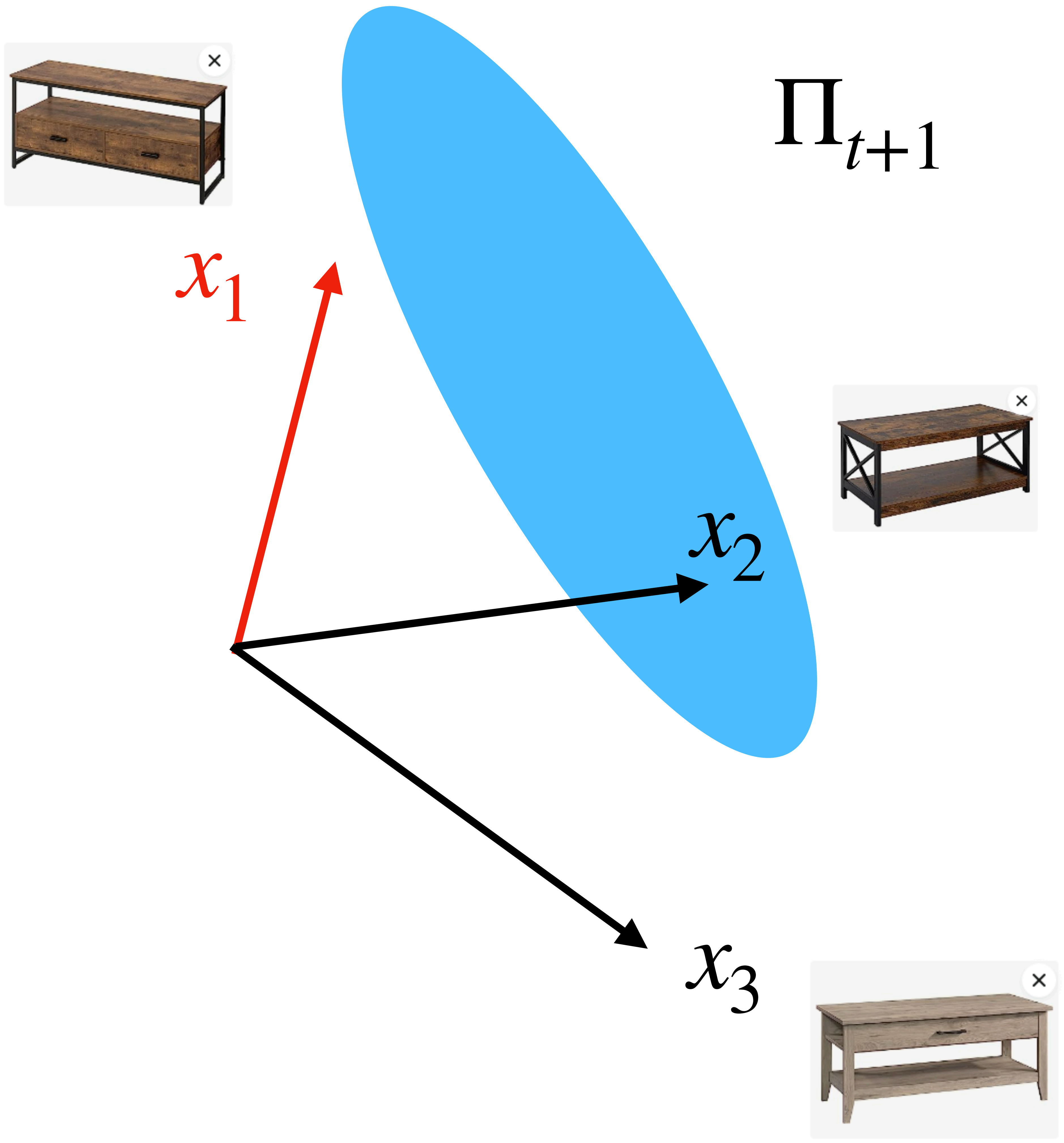


# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0,I)$

**for** $t = 1,2,\ldots$

1. Sample $\theta_t \sim \Pi_t$
2. Play $x_t = \arg\max_{x \in \mathcal{X}} \theta_t^\top x$
3. Observe $y_t$, update $\hat{\theta}_{t+1}$
4. Update $\Pi_{t+1}$

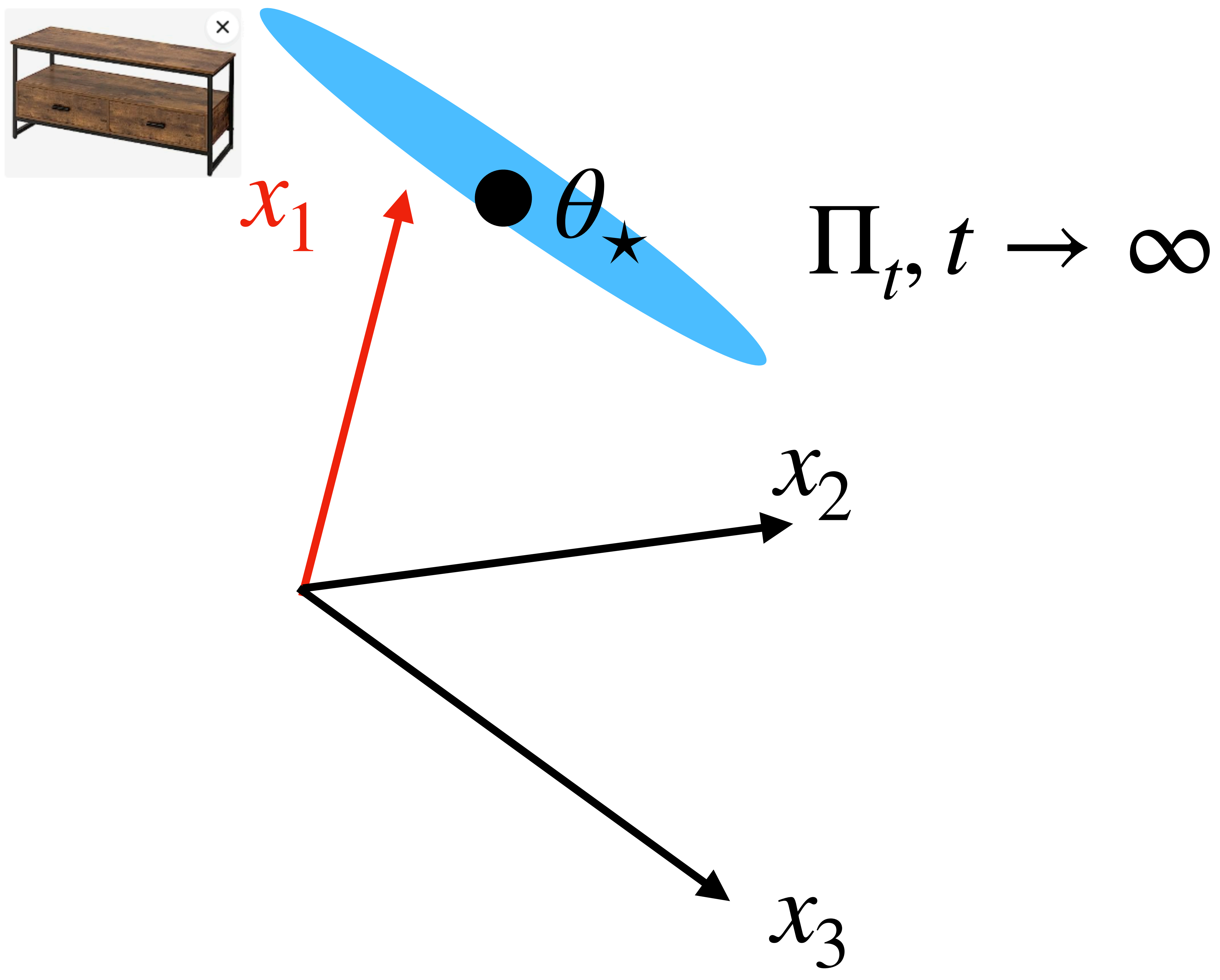


# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0, I)$

**for** $t = 1, 2, \ldots$

1. Sample $\theta_t \sim \Pi_t$
2. Play $x_t = \arg\max_{x \in \mathcal{X}} \theta_t^\top x$
3. Observe $y_t$, update $\hat{\theta}_{t+1}$
4. Update $\Pi_{t+1}$



# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0, I)$

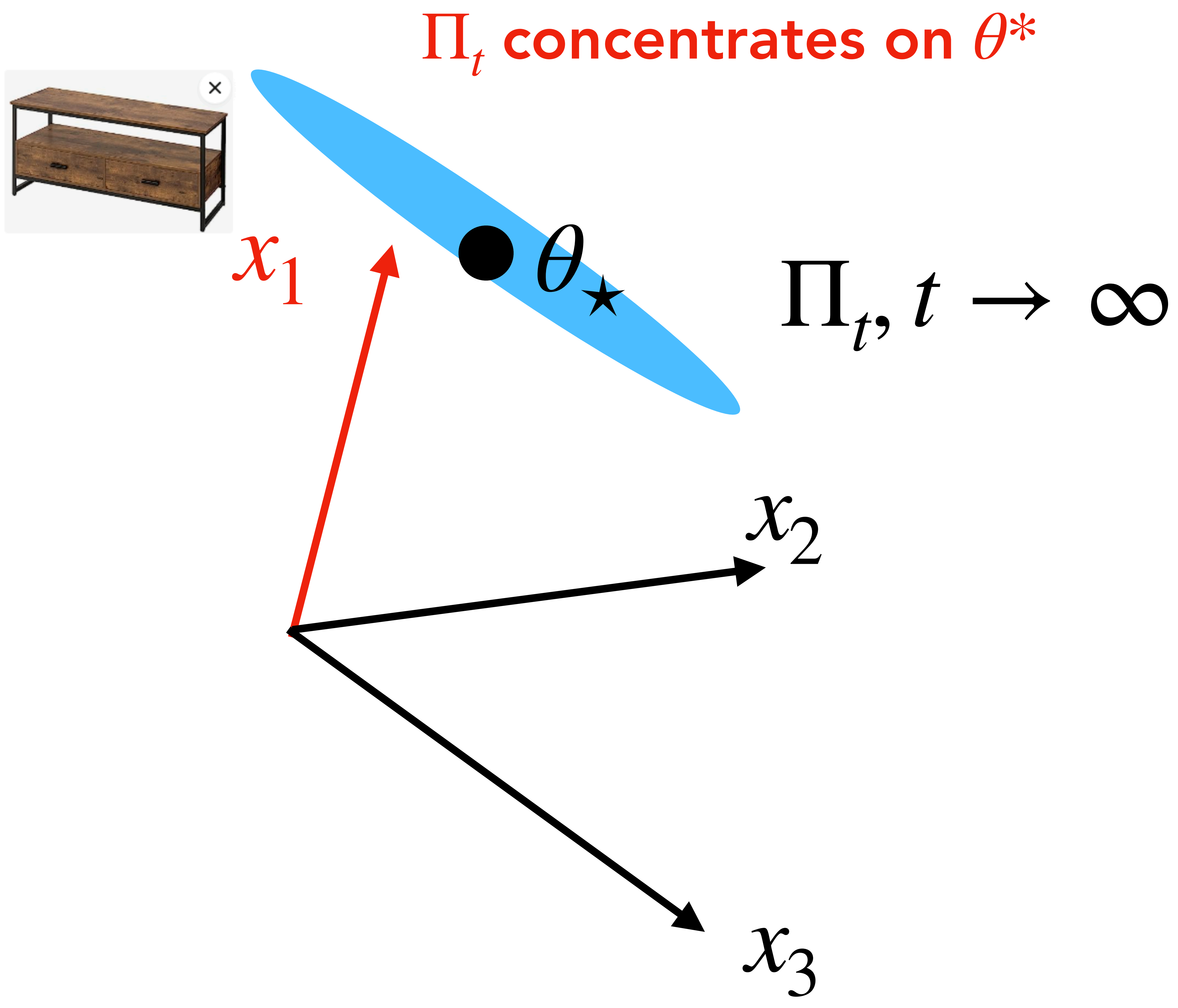
**for** $t = 1, 2, \ldots$

1. Sample $\theta_t \sim \Pi_t$

2. Play $x_t = \arg\max_{x \in \mathcal{X}} \theta_t^\top x$

3. Observe $y_t$, update $\hat{\theta}_{t+1}$

4. Update $\Pi_{t+1}$



# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0,I)$

**for** $t = 1,2,\ldots$

1. Sample $\theta_t \sim \Pi_t$
2. Play $x_t = \arg\max_{x \in \mathcal{X}} \theta_t^\top x$
3. Observe $y_t$, update $\hat{\theta}_{t+1}$
4. Update $\Pi_{t+1}$

# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0, I)$

**for** $t = 1, 2, \ldots$

1. Sample $\theta_t \sim \Pi_t$
2. Play $x_t = \arg\max_{x \in \mathcal{X}} \theta_t^\top x$
3. Observe $y_t$, update $\hat{\theta}_{t+1}$
4. Update $\Pi_{t+1}$

**Why Thompson Sampling?**

# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0, I)$

**for** $t = 1, 2, \ldots$

1. Sample $\theta_t \sim \Pi_t$
2. Play $x_t = \arg\max_{x \in \mathcal{X}} \theta_t^\top x$
3. Observe $y_t$, update $\hat{\theta}_{t+1}$
4. Update $\Pi_{t+1}$

**Why Thompson Sampling?**

**Reward Maximizing** $Reg_{Bayes} = O(d\sqrt{T})$

# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0,I)$

**for** $t = 1,2,\ldots$

1. Sample $\theta_t \sim \Pi_t$
2. Play $x_t = \arg\max_{x \in \mathcal{X}} \theta_t^\top x$
3. Observe $y_t$, update $\hat{\theta}_{t+1}$
4. Update $\Pi_{t+1}$

**<u>Why Thompson Sampling?</u>**

**Reward Maximizing** $Reg_{Bayes} = O(d\sqrt{T})$

# Prior Art: Thompson Sampling

**Input:** $\mathsf{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0,I)$

**for** $t = 1,2,\ldots$

1. Sample $\theta_t \sim \Pi_t$

2. Play $x_t = \arg\max_{x \in \mathsf{X}} \theta_t^\top x$

3. Observe $y_t$, update $\hat{\theta}_{t+1}$

4. Update $\Pi_{t+1}$

**Why Thompson Sampling?**

**Reward Maximizing** $Reg_{Bayes} = O(d\sqrt{T})$

**Computationally requires:**

# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0, I)$

**for** $t = 1, 2, \ldots$

1. Sample $\theta_t \sim \Pi_t$
2. Play $x_t = \arg\max_{x \in \mathcal{X}} \theta_t^\top x$
3. Observe $y_t$, update $\hat{\theta}_{t+1}$
4. Update $\Pi_{t+1}$

**Why Thompson Sampling?**

**Reward Maximizing** $Reg_{Bayes} = O(d\sqrt{T})$

**Computationally requires:**

- Ability to sample from a posterior $\Pi_t$

# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0,I)$

**for** $t = 1,2,\ldots$

1. Sample $\theta_t \sim \Pi_t$
2. Play $x_t = \arg\max_{x \in \mathcal{X}} \theta_t^\top x$
3. Observe $y_t$, update $\hat{\theta}_{t+1}$
4. Update $\Pi_{t+1}$

**Why Thompson Sampling?**

**Reward Maximizing** $Reg_{Bayes} = O(d\sqrt{T})$

**Computationally requires:**

- Ability to sample from a posterior $\Pi_t$
- Easy access to argmax oracle $\arg\max_{x \in \mathcal{X}} \theta^\top x$

# Prior Art: Thompson Sampling

**Input:** $\mathcal{X} \subset \mathbb{R}^d$, **prior** $\Pi_0 = N(0,I)$

**for** $t = 1,2,\ldots$

1. Sample $\theta_t \sim \Pi_t$

2. Play $x_t = \arg\max_{x \in \mathcal{X}} \theta_t^\top x$

3. Observe $y_t$, update $\hat{\theta}_{t+1}$

4. Update $\Pi_{t+1}$

**Why Thompson Sampling?**

**Reward Maximizing** $Reg_{Bayes} = O(d\sqrt{T})$

**Computationally requires:**

- Ability to sample from a posterior $\Pi_t$
- Easy access to argmax oracle $\arg\max_{x \in \mathcal{X}} \theta^\top x$

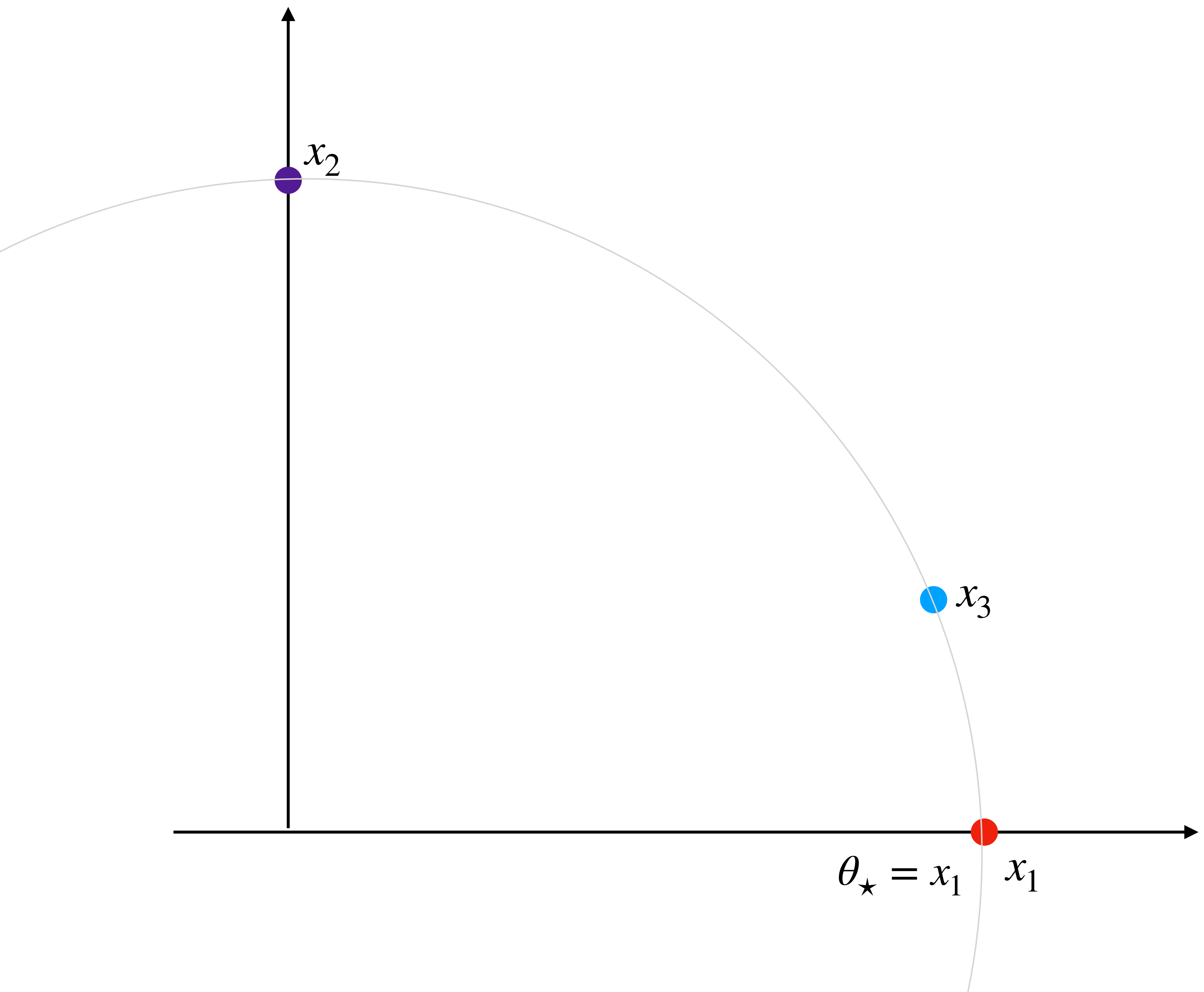
**May never need to explicitly maintain $\mathcal{X} = \mathcal{Z}$!**

# Sub-Optimality of TS for BAI

## A hard instance (Soare et al. 2014)

$$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \ldots, d$$

$$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$$

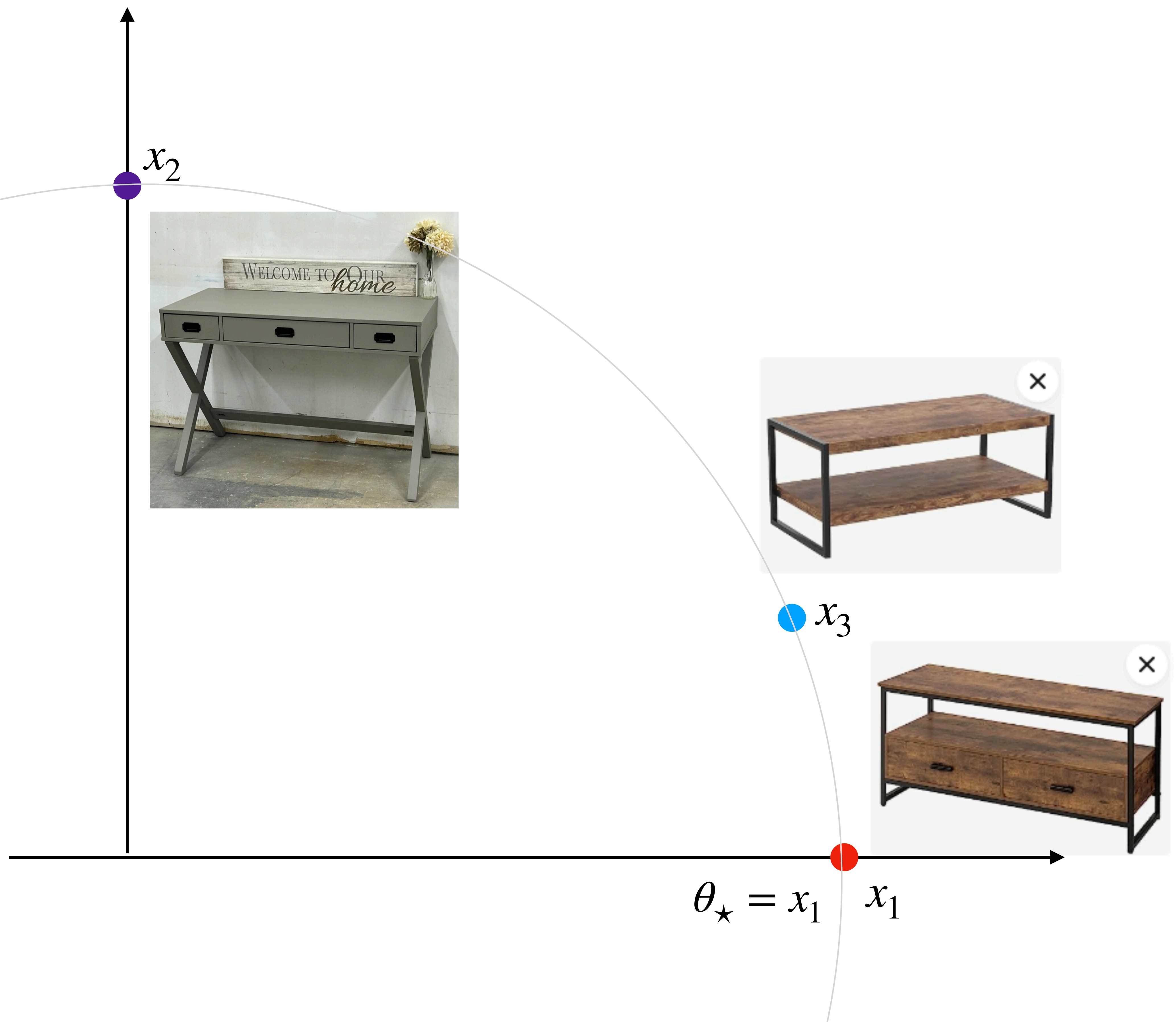


# Sub-Optimality of TS for BAI

## A hard instance (Soare et al. 2014)

$$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \ldots, d$$

$$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$$

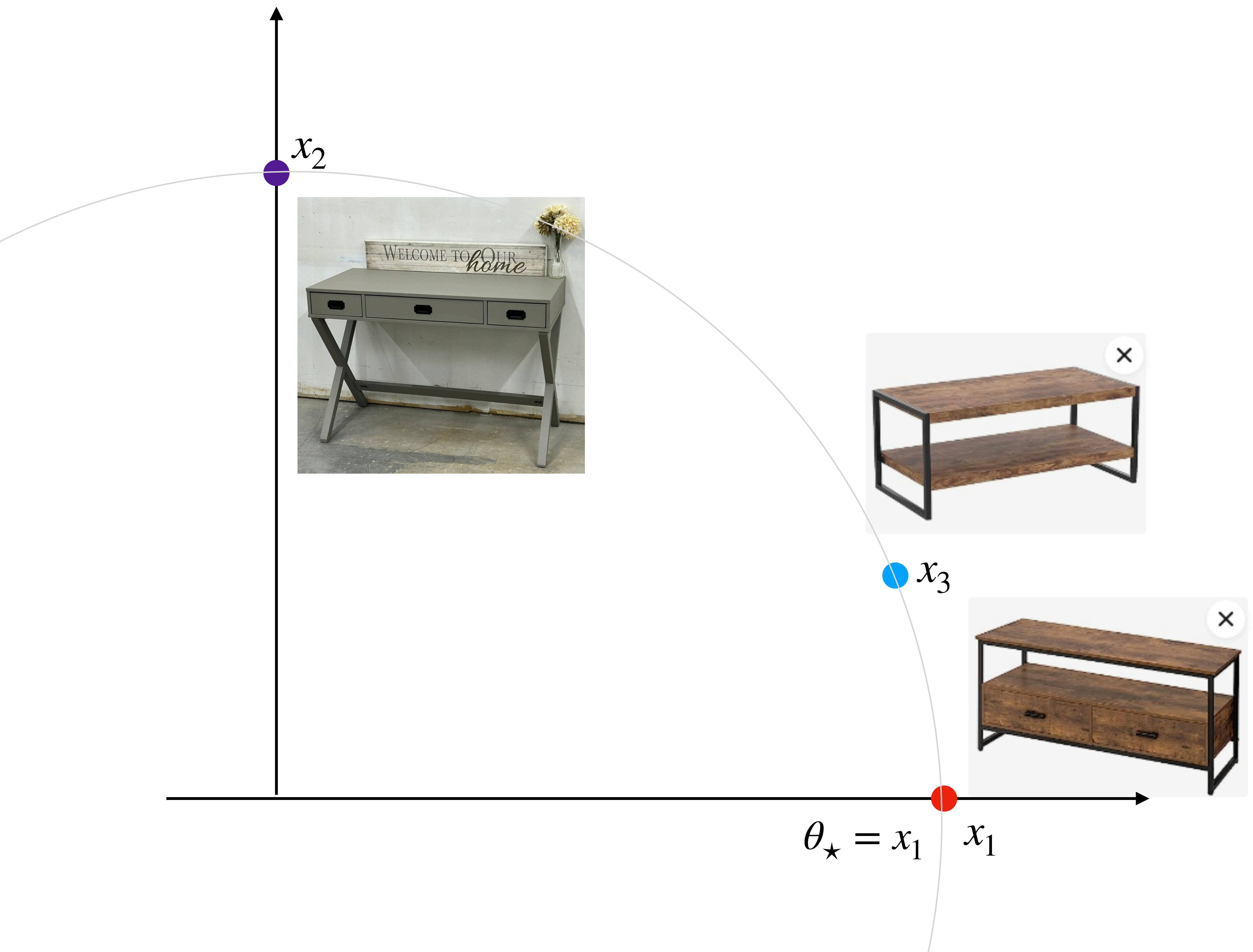


# Sub-Optimality of TS for BAI

## A hard instance (Soare et al. 2014)

$$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \ldots, d$$

$$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$$



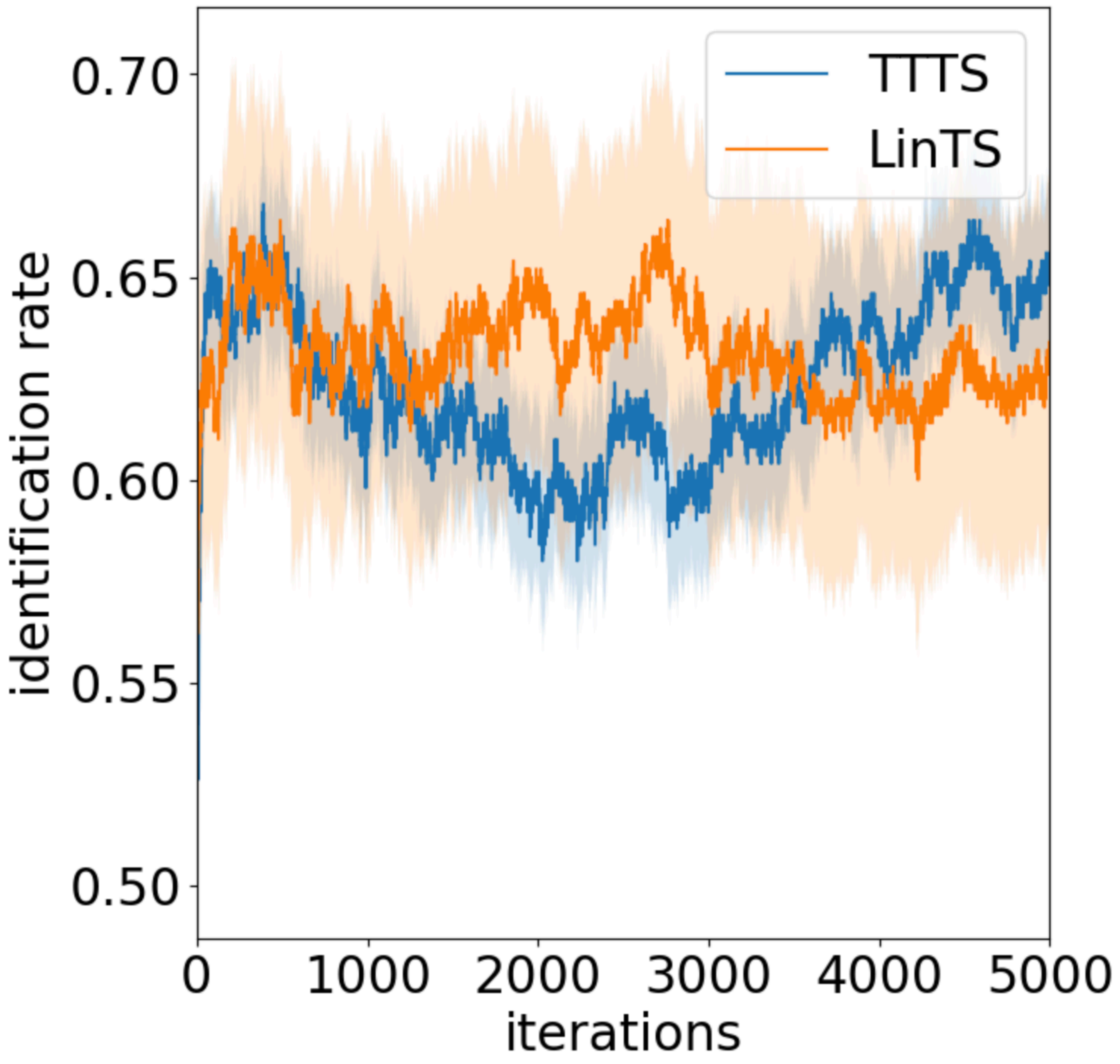



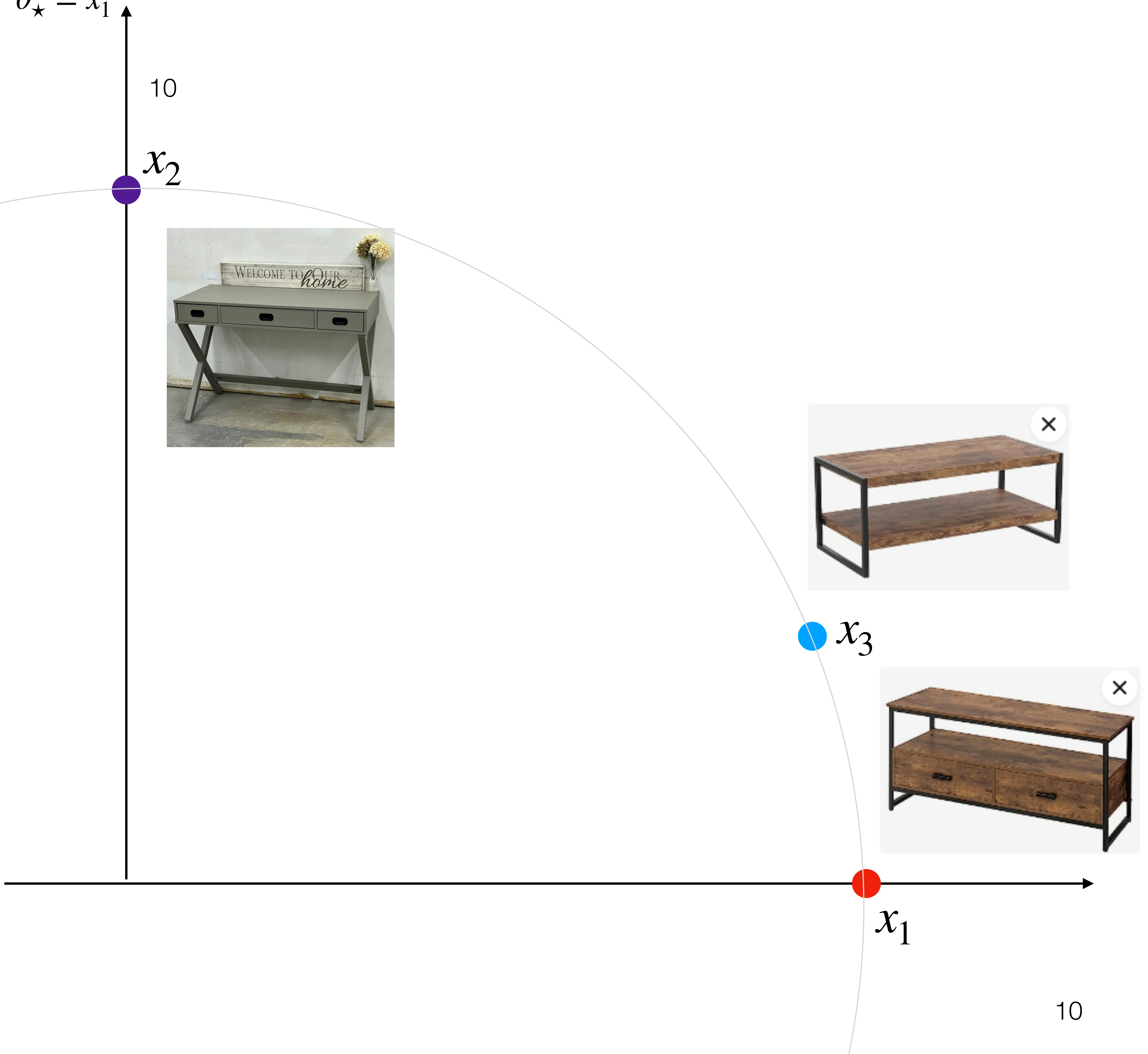

**Figure: Identification rate for TS and Top-Two TS [Russo, 2016]**

# Sub-Optimality of TS for BAI

$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \ldots, d$

$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$

$\theta_\star = x_1$

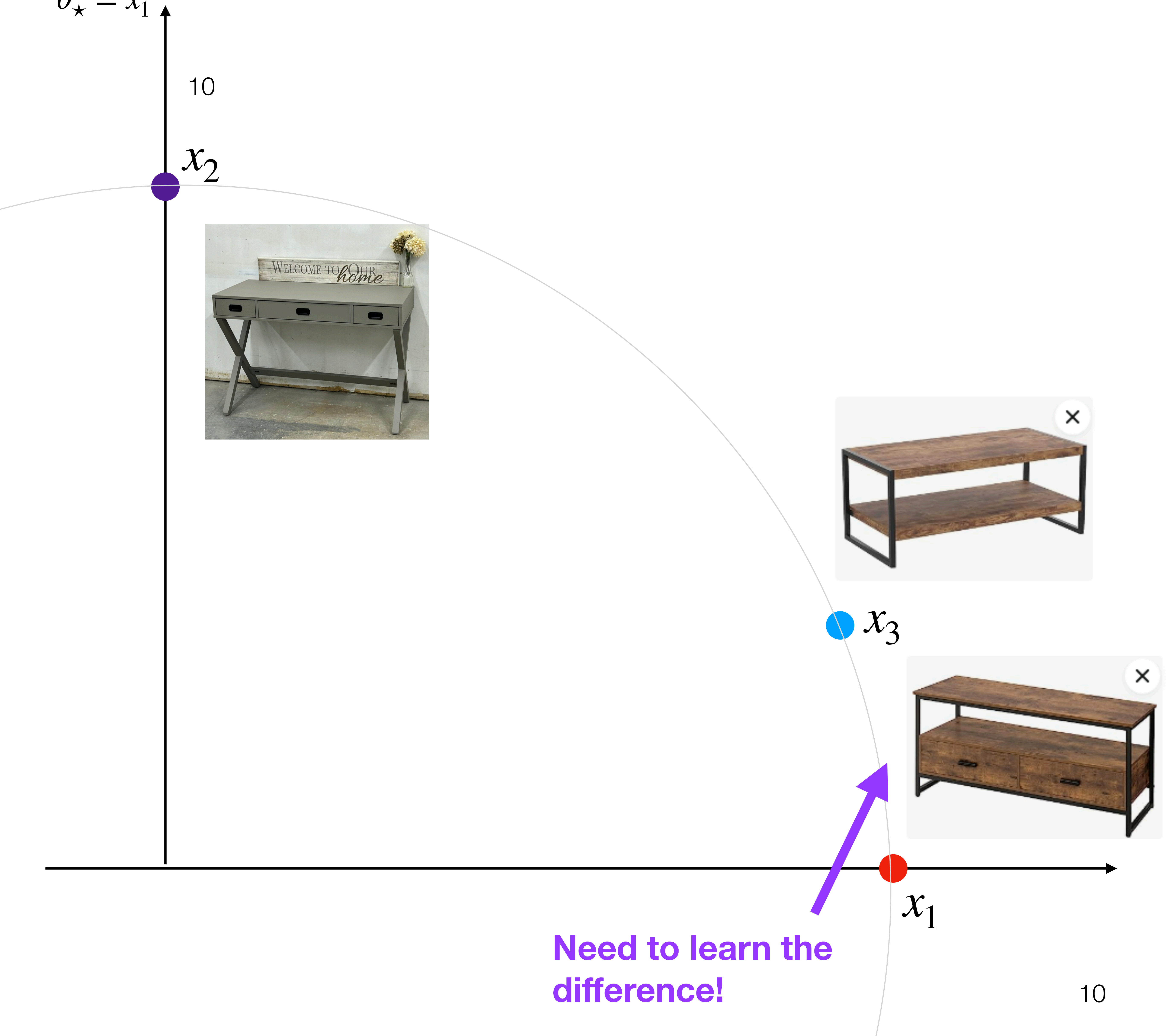


# Sub-Optimality of TS for BAI

$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \ldots, d$

$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$

$\theta_\star = x_1$

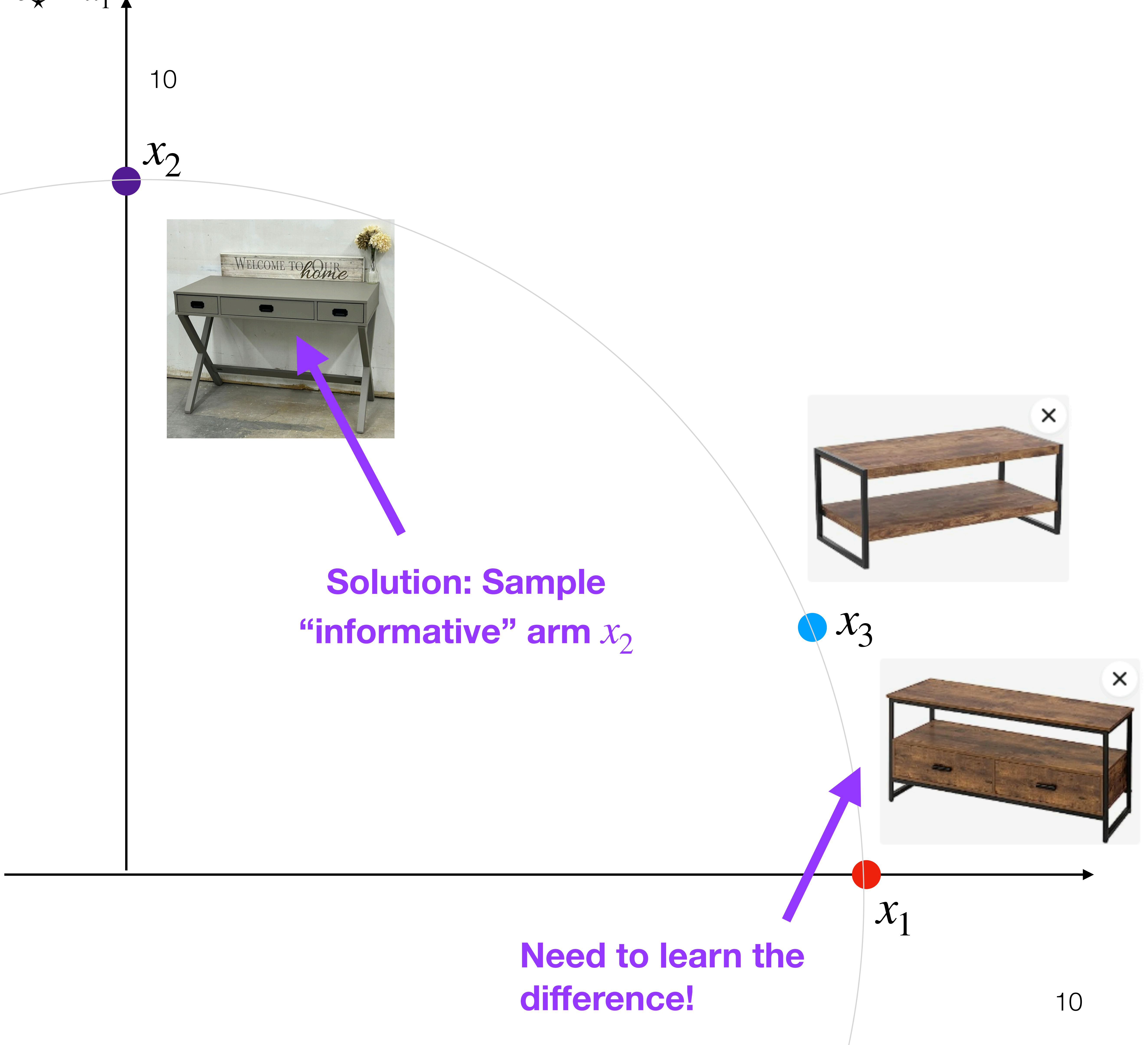


# Sub-Optimality of TS for BAI

$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \ldots, d$

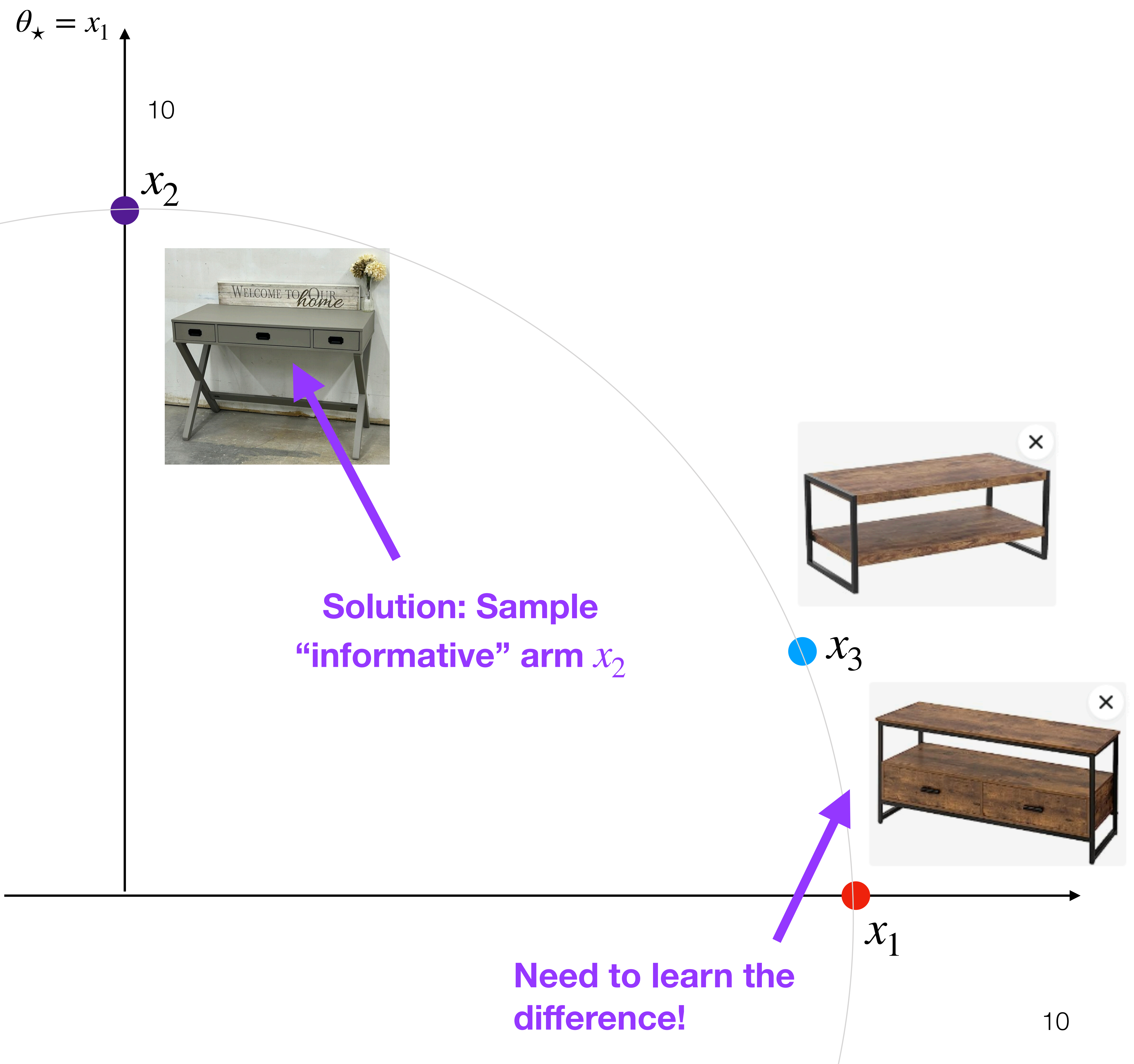
$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$

$\theta_\star = x_1$



$$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \ldots, d$$
$$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$$

# Sub-Optimality of TS for BAI

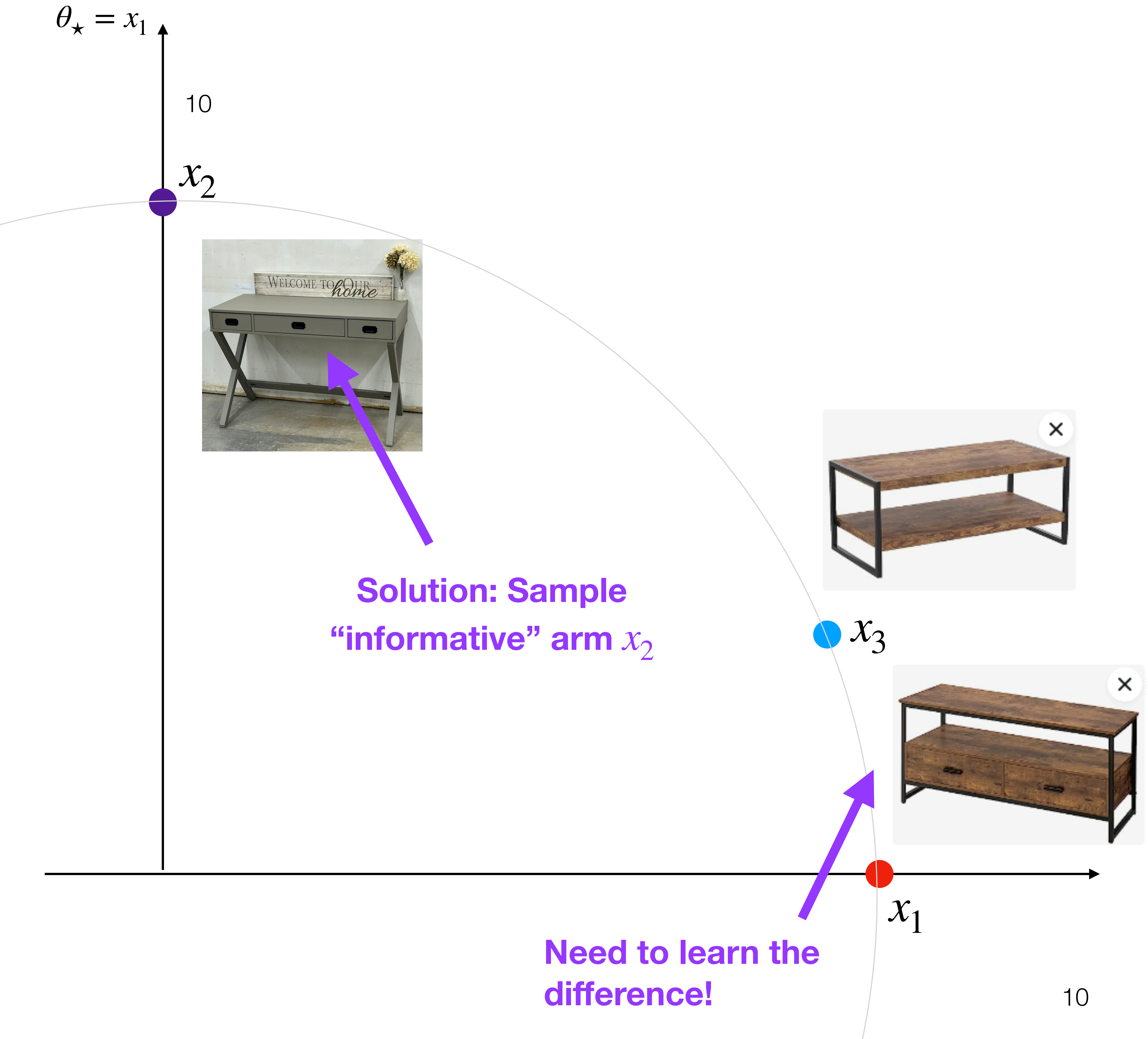



**However, Thompson sampling tends to pull arms with high rewards!**

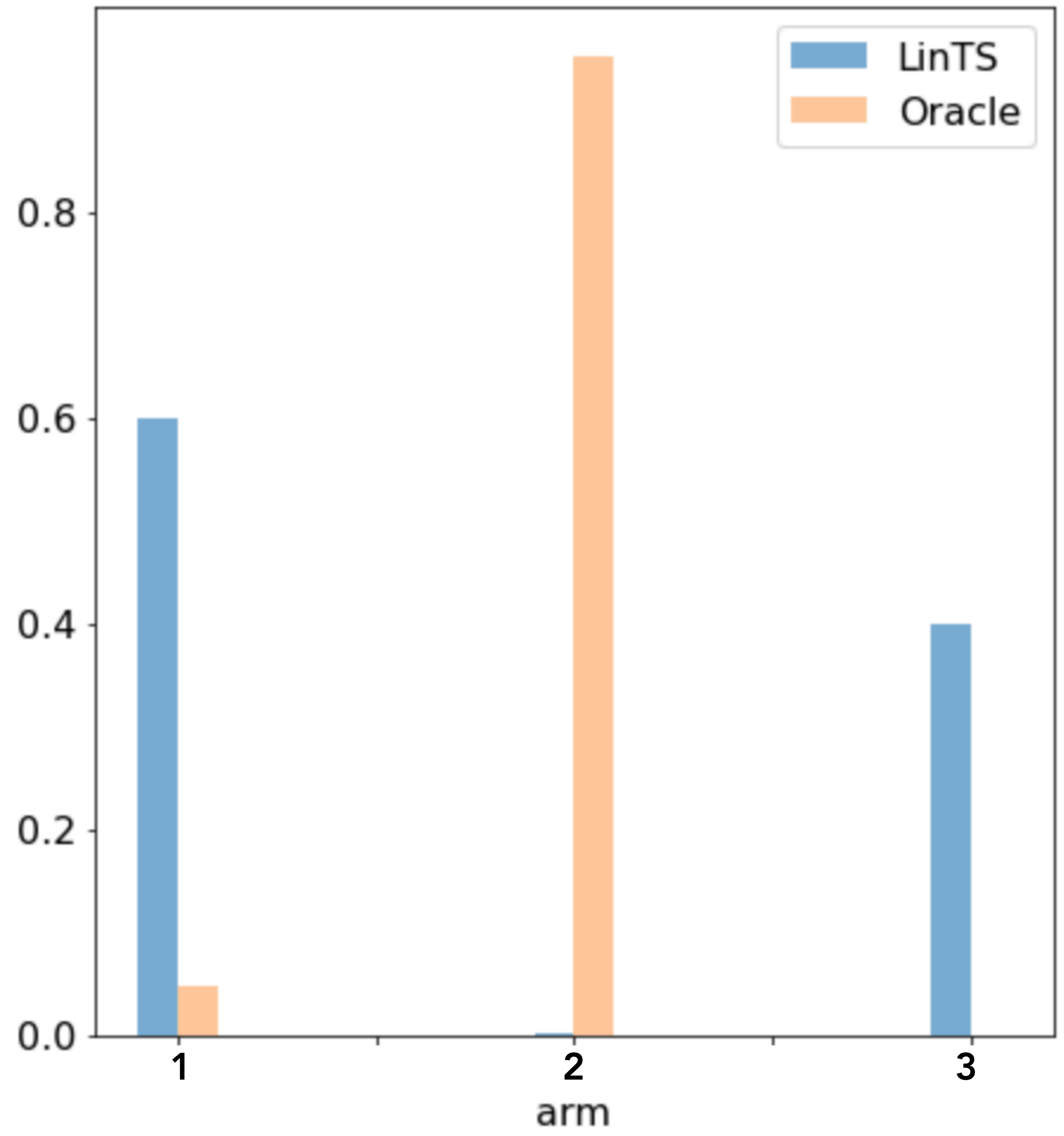
# Sub-Optimality of TS for BAI

$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \ldots, d$

$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$




**However, Thompson sampling tends to pull arms with high rewards!**



# Existing Optimal Approaches

# Existing Optimal Approaches

**More recently …**

# Existing Optimal Approaches

**More recently …**

**Optimal Approaches:** LinGame [Degenne, Menard, Shang, Valko '20]

# Existing Optimal Approaches

**More recently …**

**Optimal Approaches:** LinGame [Degenne, Menard, Shang, Valko '20]

For $t = 1,2,\cdots$

# Existing Optimal Approaches

**More recently …**

**Optimal Approaches:** LinGame [Degenne, Menard, Shang, Valko '20]

**For** $t = 1,2,\cdots$

1. compute a *leader* $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$

# Existing Optimal Approaches

**More recently …**

**<u>Optimal Approaches:</u>** LinGame [Degenne, Menard, Shang, Valko '20]

**For** $t = 1,2,\cdots$

1. compute a *leader* $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. compute a *challenger* $\theta_t = \arg\min_{\theta \in \Theta^c_{\hat{z}_t}} \sum_x \lambda_{t-1,x}(x^\top(\theta - \hat{\theta}_t))^2$

   whose best arm is *not* $\hat{z}_t$

# Existing Optimal Approaches

## More recently …

**<u>Optimal Approaches:</u>** LinGame [Degenne, Menard, Shang, Valko '20]

**For** $t = 1,2,\cdots$

1. compute a *leader* $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. compute a *challenger* $\theta_t = \arg\min_{\theta} f(\lambda_{t-1}, \hat{\theta}_t, \theta)$

   whose best arm is *not* $\hat{z}_t$

# Existing Optimal Approaches

## More recently …

__Optimal Approaches:__ LinGame [Degenne, Menard, Shang, Valko '20]

**For** $t = 1,2,\cdots$

1. compute a *leader* $\hat{z}_t = \arg\max_{z \in \mathcal{Z}} z^\top \hat{\theta}_t$
2. compute a *challenger* $\theta_t = \arg\min_{\theta} f(\lambda_{t-1}, \hat{\theta}_t, \theta)$ whose best arm is *not* $\hat{z}_t$
3. update the sampling distribution $\lambda_t$

# Existing Optimal Approaches

**More recently …**

**<u>Optimal Approaches:</u>** LinGame [Degenne, Menard, Shang, Valko '20]

**For** $t = 1,2,\cdots$

1. compute a *leader* $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. compute a *challenger* $\theta_t = \arg\min_{\theta} f(\lambda_{t-1}, \hat{\theta}_t, \theta)$
   whose best arm is *not* $\hat{z}_t$
3. update the sampling distribution $\lambda_t$
4. sample $x_t \sim \lambda_t$, update $\hat{\theta}_{t+1}$

# Existing Optimal Approaches

## More recently …

**<u>Optimal Approaches:</u>** LinGame [Degenne, Menard, Shang, Valko '20]

**For** $t = 1,2,\cdots$

1. compute a *leader* $\hat{z}_t = \arg\max_{z \in \mathcal{Z}} z^\top \hat{\theta}_t$
2. compute a *challenger* $\theta_t = \arg\min_{\theta} f(\lambda_{t-1}, \hat{\theta}_t, \theta)$

   whose best arm is *not* $\hat{z}_t$ $\longrightarrow$ **Needs to enumerate $|\mathcal{Z}| \Rightarrow$ computationally expensive!**
3. update the sampling distribution $\lambda_t$
4. sample $x_t \sim \lambda_t$, update $\hat{\theta}_{t+1}$

# Existing Optimal Approaches

**More recently …**

<u>**Optimal Approaches:**</u> LinGame [Degenne, Menard, Shang, Valko '20]

**For** $t = 1,2,\cdots$

1. compute a *leader* $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. compute a *challenger* $\theta_t = \arg\min_{\theta} f(\lambda_{t-1}, \hat{\theta}_t, \theta)$

   whose best arm is *not* $\hat{z}_t$ $\longrightarrow$ **Needs to enumerate $|Z| \Rightarrow$ computationally expensive!**
3. update the sampling distribution $\lambda_t$
4. sample $x_t \sim \lambda_t$, update $\hat{\theta}_{t+1}$

**Can we achieve the best of both worlds?**

# Existing Optimal Approaches

## More recently …

**Optimal Approaches:** LinGame [Degenne, Menard, Shang, Valko '20]

**For** $t = 1,2,\cdots$

1. compute a *leader* $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. compute a *challenger* $\theta_t = \arg\min_{\theta} f(\lambda_{t-1}, \hat{\theta}_t, \theta)$

   whose best arm is *not* $\hat{z}_t$ $\longrightarrow$ **Needs to enumerate $|Z| \Rightarrow$ computationally expensive!**
3. update the sampling distribution $\lambda_t$
4. sample $x_t \sim \lambda_t$, update $\hat{\theta}_{t+1}$

**Can we achieve the best of both worlds?**

**Our contribution:** **an *optimal* algorithm that *only* requires argmax oracle and sampling!**

# Our Algorithm

# Our Algorithm

**Like Thompson sampling, we maintain a distribution $p$ over $\theta \in \Theta$**

# Our Algorithm

**Like Thompson sampling, we maintain a distribution $p$ over $\theta \in \Theta$**

For $t = 1,2,\cdots$

# Our Algorithm

**Like Thompson sampling, we maintain a distribution $p$ over $\theta \in \Theta$**

**For** $t = 1,2,\cdots$

1. compute the *leader* $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$

# Our Algorithm

**Like Thompson sampling, we maintain a distribution $p$ over $\theta \in \Theta$**

**For** $t = 1, 2, \cdots$

1. compute the *leader* $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. sample a *challenger* $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$

# Our Algorithm

**Like Thompson sampling, we maintain a distribution $p$ over $\theta \in \Theta$**

**For** $t = 1,2,\cdots$

1. compute the *leader* $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. sample a *challenger* $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$
3. sample "the most informative arm" $x_t \sim \lambda_t$, observe $y_t$

# Our Algorithm

**Like Thompson sampling, we maintain a distribution $p$ over $\theta \in \Theta$**

**For** $t = 1,2,\cdots$

1. compute the *leader* $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. sample a *challenger* $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$
3. sample "the most informative arm" $x_t \sim \lambda_t$, observe $y_t$
4. update $\hat{\theta}_{t+1}$, $\lambda_{t+1}$ and $p_{t+1}$

# Our Algorithm

**Like Thompson sampling, we maintain a distribution $p$ over $\theta \in \Theta$**

**For** $t = 1,2,\cdots$

1. compute the *leader* $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. sample a *challenger* $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$
3. sample "the most informative arm" $x_t \sim \lambda_t$, observe $y_t$
4. update $\hat{\theta}_{t+1}$, $\lambda_{t+1}$ and $p_{t+1}$

**Replaces searching over |Z| with sampling → computationally efficient**

# Our Algorithm

**Like Thompson sampling, we maintain a distribution $p$ over $\theta \in \Theta$**

**For** $t = 1,2,\cdots$

1. compute the *leader* $\hat{z}_t = \arg\max_{z \in \mathcal{Z}} z^\top \hat{\theta}_t$
2. sample a *challenger* $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$
3. sample "the most informative arm" $x_t \sim \lambda_t$, observe $y_t$
4. update $\hat{\theta}_{t+1}$, $\lambda_{t+1}$ and $p_{t+1}$

**Replaces searching over |Z| with sampling → computationally efficient**

**Need to design $\lambda$ and $p$ carefully!**

**PEPS:** Pure Exploration with Projection-Free Sampling

**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

for $t = 1,2,\cdots,T$

**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

for $t = 1,2,\cdots,T$

1. Compute $\hat{z}_t = \arg\max_{z \in \mathrm{Z}} z^\top \hat{\theta}_t$

**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

for $t = 1,2,\cdots,T$

1. Compute $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$

2. Sample $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$, $x_t \sim \lambda_t$

**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

for $t = 1,2,\cdots,T$

1. Compute $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$

2. Sample $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$, $x_t \sim \lambda_t$

3. Observe $y_t = x_t^\top \theta_\star + \epsilon_t$, update $\hat{\theta}_{t+1}$

**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

for $t = 1,2,\cdots,T$

1. Compute $\hat{z}_t = \arg\max_{z \in \mathrm{Z}} z^\top \hat{\theta}_t$

2. Sample $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$, $x_t \sim \lambda_t$

3. Observe $y_t = x_t^\top \theta_\star + \epsilon_t$, update $\hat{\theta}_{t+1}$

**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

for $t = 1,2,\cdots,T$

1. Compute $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$

2. Sample $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$, $x_t \sim \lambda_t$

3. Observe $y_t = x_t^\top \theta_\star + \epsilon_t$, update $\hat{\theta}_{t+1}$

4. Update $\lambda_{t+1,x} \leftarrow \lambda_{t,x} e^{\eta(x^\top(\theta_t - \hat{\theta}_t))^2}$

**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

for $t = 1,2,\cdots,T$

1. Compute $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. Sample $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$, $x_t \sim \lambda_t$
3. Observe $y_t = x_t^\top \theta_\star + \epsilon_t$, update $\hat{\theta}_{t+1}$
4. Update $\lambda_{t+1,x} \leftarrow \lambda_{t,x} e^{\eta(x^\top(\theta_t - \hat{\theta}_t))^2}$

**pull arm $x$ that maximizes the difference $|x^\top(\theta_t - \hat{\theta}_t)|$**

**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

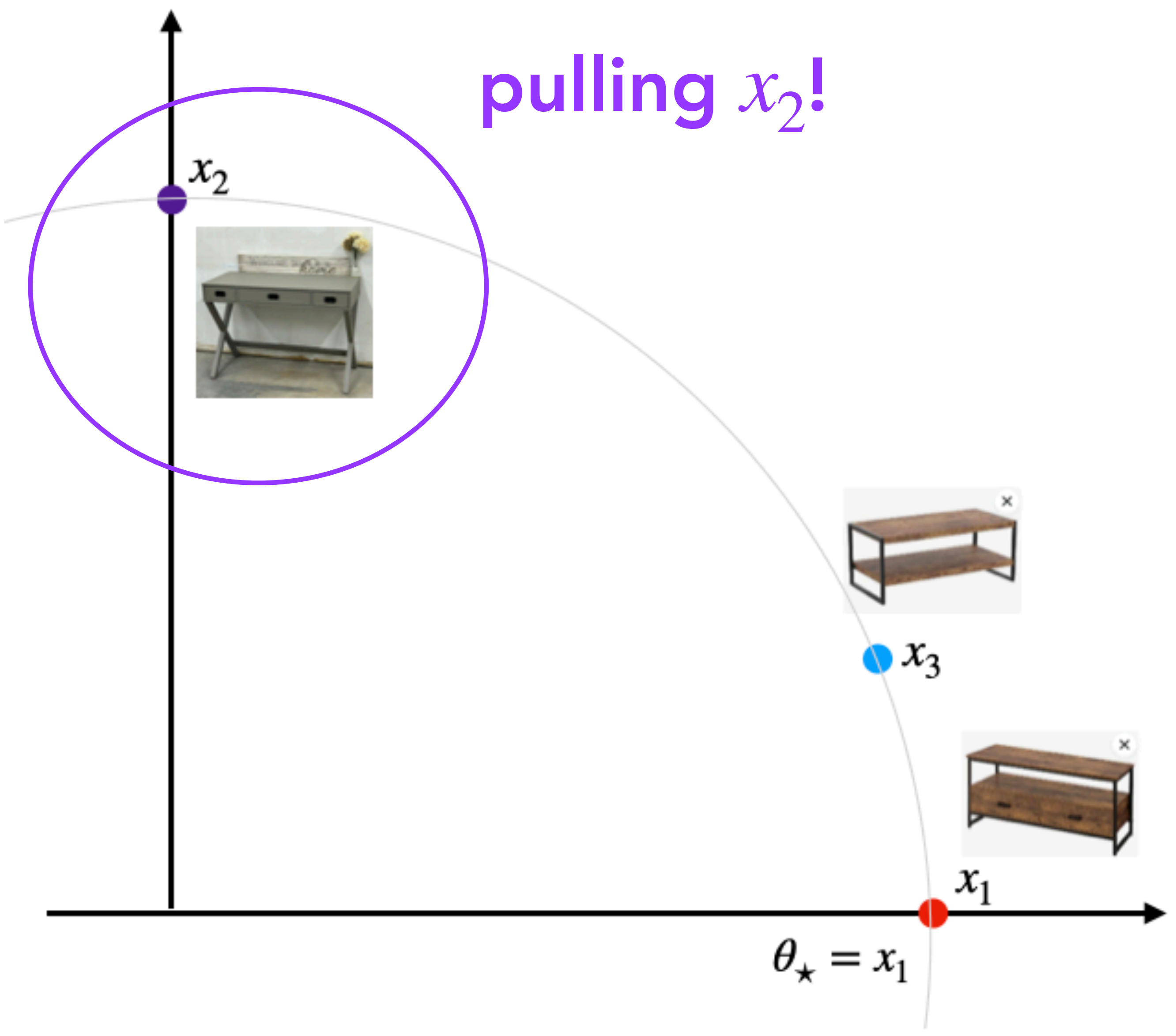
for $t = 1,2,\cdots,T$

1. Compute $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$

2. Sample $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$, $x_t \sim \lambda_t$

3. Observe $y_t = x_t^\top \theta_\star + \epsilon_t$, update $\hat{\theta}_{t+1}$

4. Update $\lambda_{t+1,x} \leftarrow \lambda_{t,x} e^{\eta(x^\top(\theta_t - \hat{\theta}_t))^2}$



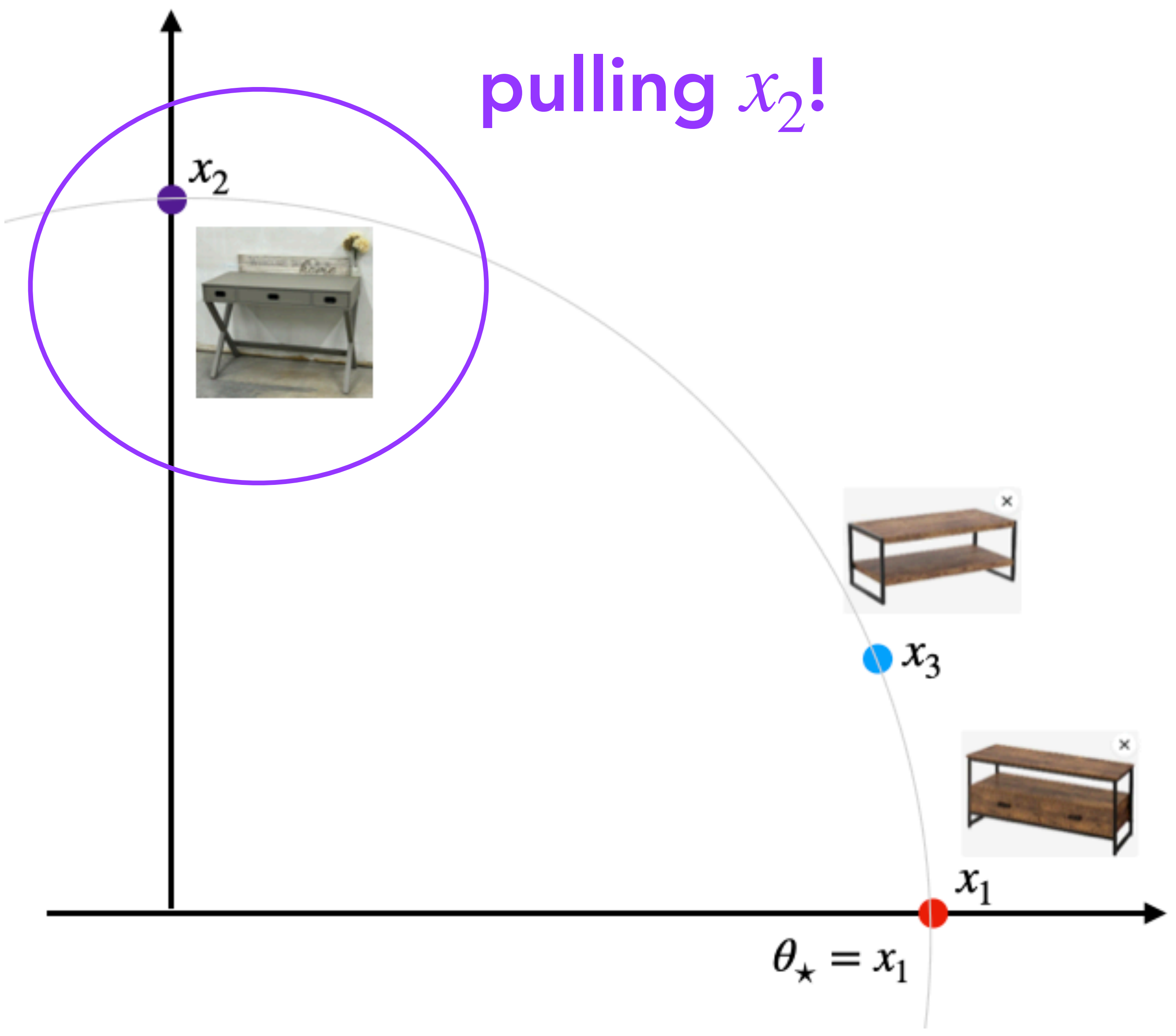

**pull arm $x$ that maximizes the difference $|x^\top(\theta_t - \hat{\theta}_t)|$**

**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

for $t = 1, 2, \cdots, T$

1. Compute $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. Sample $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$, $x_t \sim \lambda_t$
3. Observe $y_t = x_t^\top \theta_\star + \epsilon_t$, update $\hat{\theta}_{t+1}$
4. Update $\lambda_{t+1,x} \leftarrow \lambda_{t,x} e^{\eta(x^\top(\theta_t - \hat{\theta}_t))^2}$
5. Update $p_{t+1} = N(\hat{\theta}_{t+1}, (\sum_{s=1}^{t} x_s x_s^\top)^{-1})$




pull arm $x$ that maximizes the difference $|x^\top(\theta_t - \hat{\theta}_t)|$

**PEPS:** Pure Exploration with Projection-Free Sampling

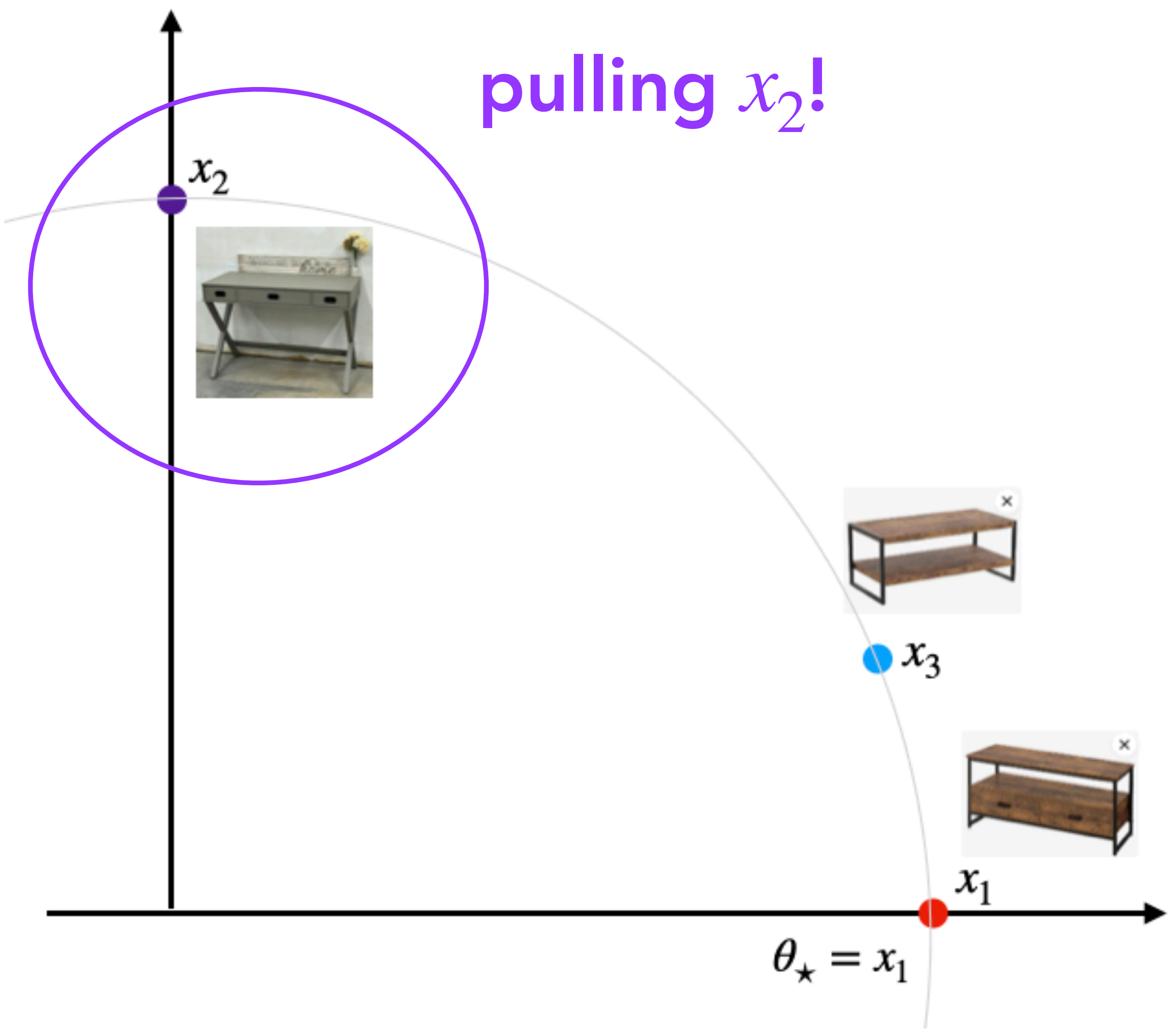
Input: X, Z, T, $\eta$

for $t = 1, 2, \cdots, T$

1. Compute $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. Sample $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$, $x_t \sim \lambda_t$
3. Observe $y_t = x_t^\top \theta_\star + \epsilon_t$, update $\hat{\theta}_{t+1}$
4. Update $\lambda_{t+1,x} \leftarrow \lambda_{t,x} e^{\eta(x^\top(\theta_t - \hat{\theta}_t))^2}$
5. Update $p_{t+1} = N(\hat{\theta}_{t+1}, (\sum_{s=1}^{t} x_s x_s^\top)^{-1})$

pull arm $x$ that maximizes the difference $|x^\top(\theta_t - \hat{\theta}_t)|$

Posterior Update



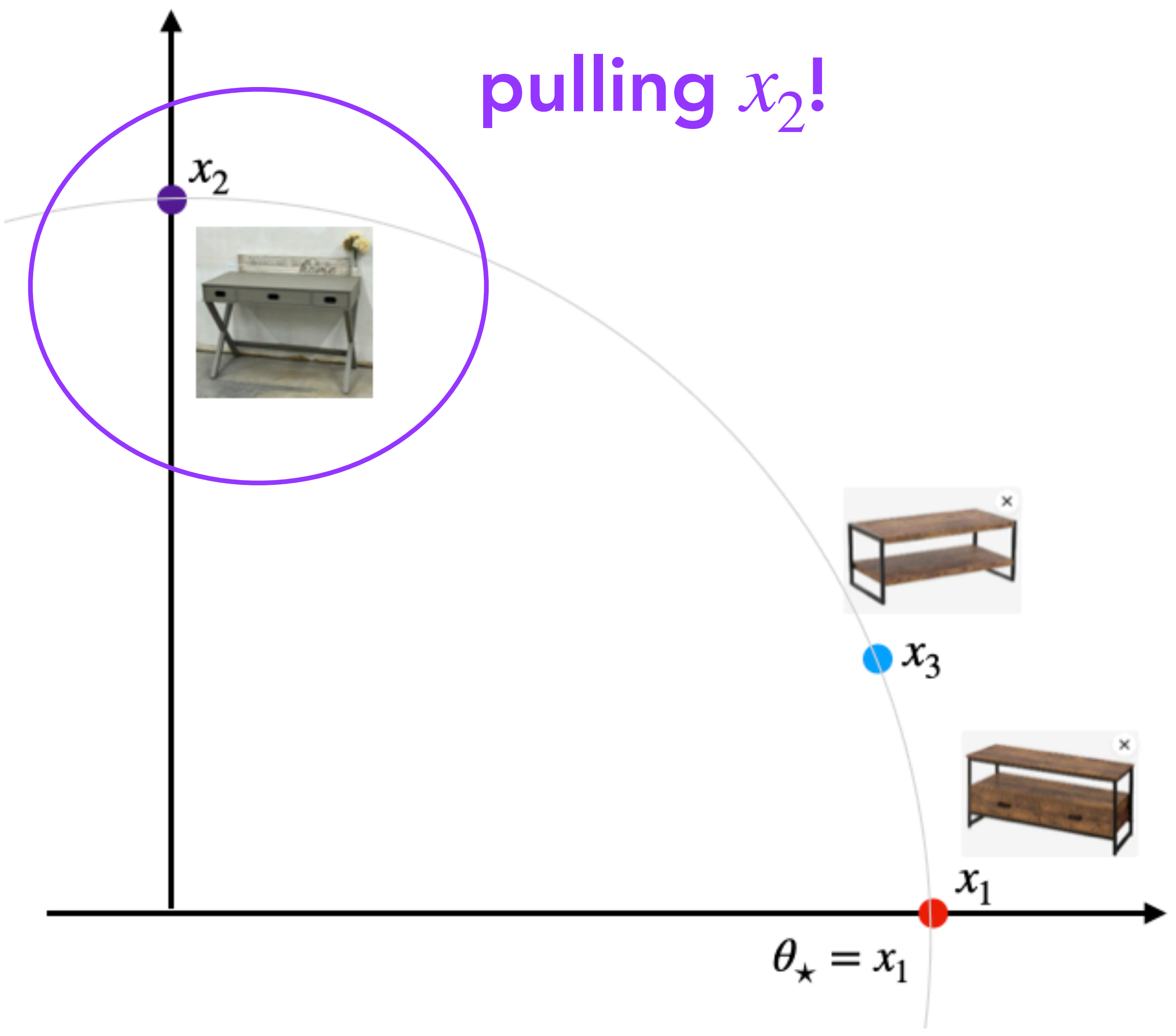
**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

for $t = 1,2,\cdots,T$

1. Compute $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. Sample $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$, $x_t \sim \lambda_t$
3. Observe $y_t = x_t^\top \theta_\star + \epsilon_t$, update $\hat{\theta}_{t+1}$
4. Update $\lambda_{t+1,x} \leftarrow \lambda_{t,x} e^{\eta (x^\top (\theta_t - \hat{\theta}_t))^2}$
5. Update $p_{t+1} = N(\hat{\theta}_{t+1}, (\sum_{s=1}^{t} x_s x_s^\top)^{-1})$

Return $\hat{z} = \arg\max_{z \in Z} z^\top \theta, \theta \sim p_T$




pull arm $x$ that maximizes the difference $|x^\top(\theta_t - \hat{\theta}_t)|$

Posterior Update

**PEPS:** Pure Exploration with Projection-Free Sampling

Input: X, Z, T, $\eta$

for $t = 1,2,\cdots,T$

1. Compute $\hat{z}_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$

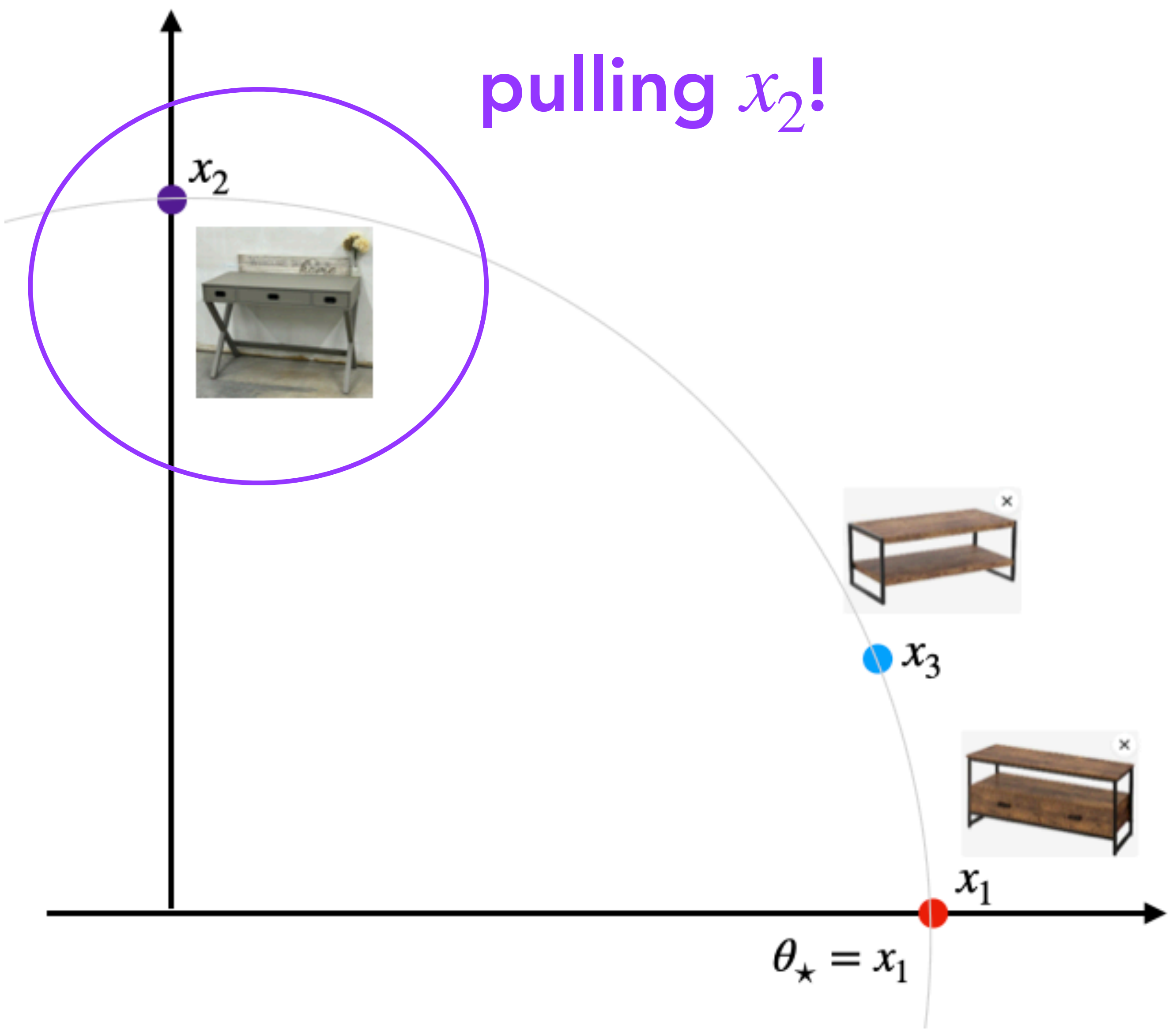
2. Sample $\theta_t \sim p_t$ whose best arm is not $\hat{z}_t$, $x_t \sim \lambda_t$

3. Observe $y_t = x_t^\top \theta_\star + \epsilon_t$, update $\hat{\theta}_{t+1}$

4. Update $\lambda_{t+1,x} \leftarrow \lambda_{t,x} e^{\eta(x^\top(\theta_t - \hat{\theta}_t))^2}$

5. Update $p_{t+1} = N(\hat{\theta}_{t+1}, (\sum_{s=1}^{t} x_s x_s^\top)^{-1})$ ← **Posterior Update**

Return $\hat{z} = \arg\max_{z \in Z} z^\top \theta, \theta \sim p_T$ ← **Final Recommendation**




**pull arm $x$ that maximizes the difference $|x^\top(\theta_t - \hat{\theta}_t)|$**

# Theoretical Guarantees: Asymptotic Optimality

# Theoretical Guarantees: Asymptotic Optimality

**Theorem (Li, Jamieson, Jain).** For some $\lambda \in \Delta_{\mathcal{X}}$ consider a procedure that draws $x_1, \cdots, x_T \sim \lambda$ and observes $y_t = \langle x_t, \theta_\star \rangle + \epsilon_t$ and computes $\widehat{z}_T = \arg\max_{z \in \mathcal{Z}} \langle z, \widehat{\theta}_T \rangle$ where $\hat{\theta}_T$ is the OLS estimate.

Denote $\Theta^c_{z_\star} = \{\theta : z_\star \neq \arg\max_{z \in \mathcal{Z}} z^\top \theta\}$, then for any $\lambda \in \Delta_{\mathcal{X}}$

$$\limsup_{T \to \infty} -\frac{1}{T} \log \mathbb{P}_{\theta_\star, x_t \sim \lambda}(\hat{z}_t \neq z_\star) \leq \tau_\star$$

$$\tau_\star = \max_{\lambda \in \Delta_{\mathcal{X}}} \min_{\theta \in \Theta^c_{z_\star}} \|\theta_\star - \theta\|^2_{A(\lambda)}$$

**worst-case KL divergence**

# Theoretical Guarantees: Asymptotic Optimality

**Theorem (Li, Jamieson, Jain).** For some $\lambda \in \Delta_{\mathcal{X}}$ consider a procedure that draws $x_1, \cdots, x_T \sim \lambda$ and observes $y_t = \langle x_t, \theta_\star \rangle + \epsilon_t$ and computes $\hat{z}_T = \arg\max_{z \in \mathcal{Z}} \langle z, \hat{\theta}_T \rangle$ where $\hat{\theta}_T$ is the OLS estimate. Denote $\Theta^c_{z_\star} = \{\theta : z_\star \neq \arg\max_{z \in \mathcal{Z}} z^\top \theta\}$, then for any $\lambda \in \Delta_{\mathcal{X}}$

$$\limsup_{T \to \infty} -\frac{1}{T} \log \mathbb{P}_{\theta_\star, x_t \sim \lambda}(\hat{z}_t \neq z_\star) \leq \tau_\star$$

$$\tau_\star = \max_{\lambda \in \Delta_{\mathcal{X}}} \min_{\theta \in \Theta^c_{z_\star}} \|\theta_\star - \theta\|^2_{A(\lambda)}$$

worst-case KL divergence

**Theorem (Li, Jamieson, Jain)** Set $\eta = O(1/\sqrt{T})$, and assume $\Theta$ is bounded. Then with probability 1

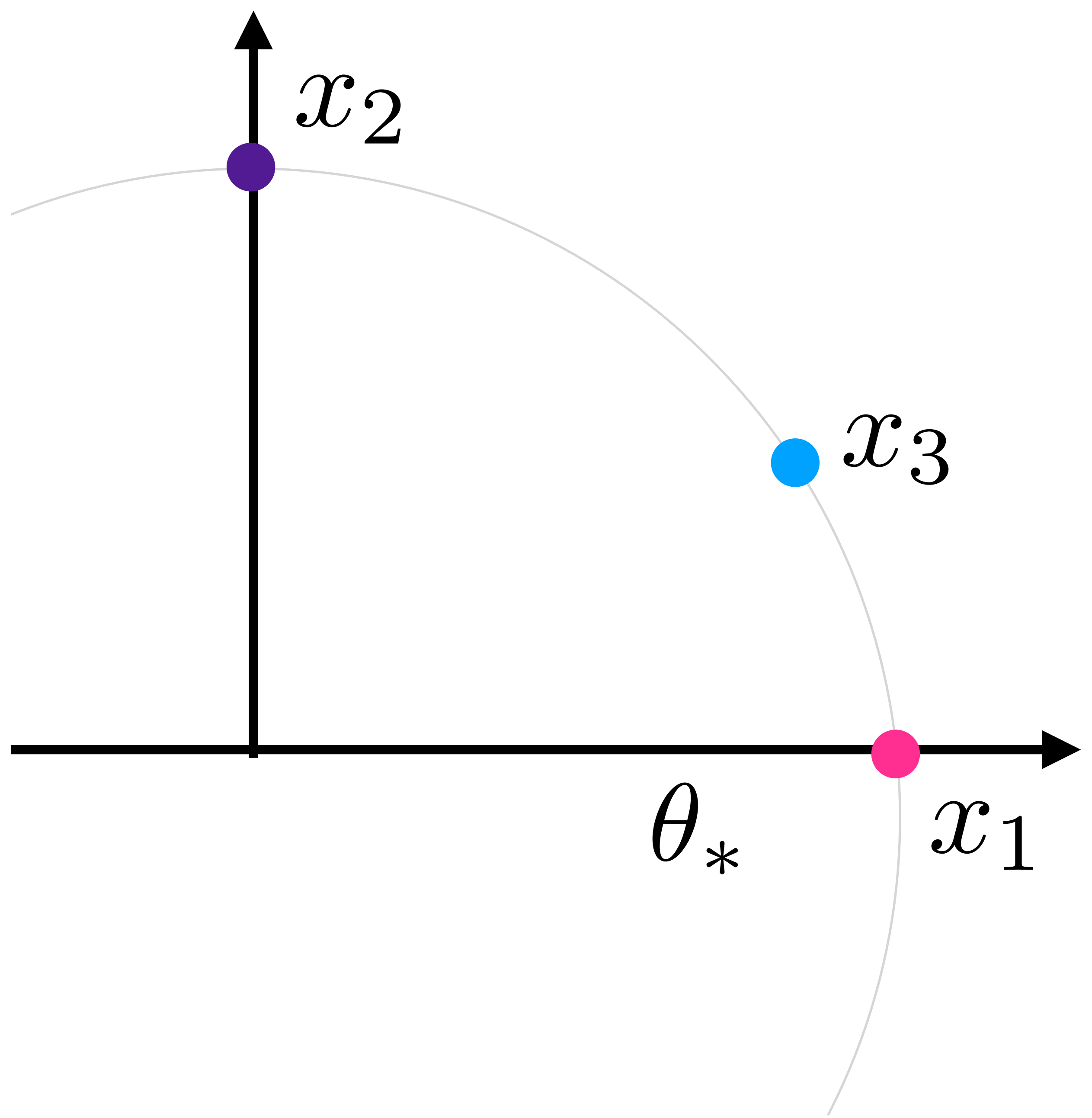
$$\lim_{T \to \infty} -\frac{1}{T} \log \mathbb{P}_{\theta \sim p_T}(\hat{z}_T \neq z_\star) = \tau_\star$$

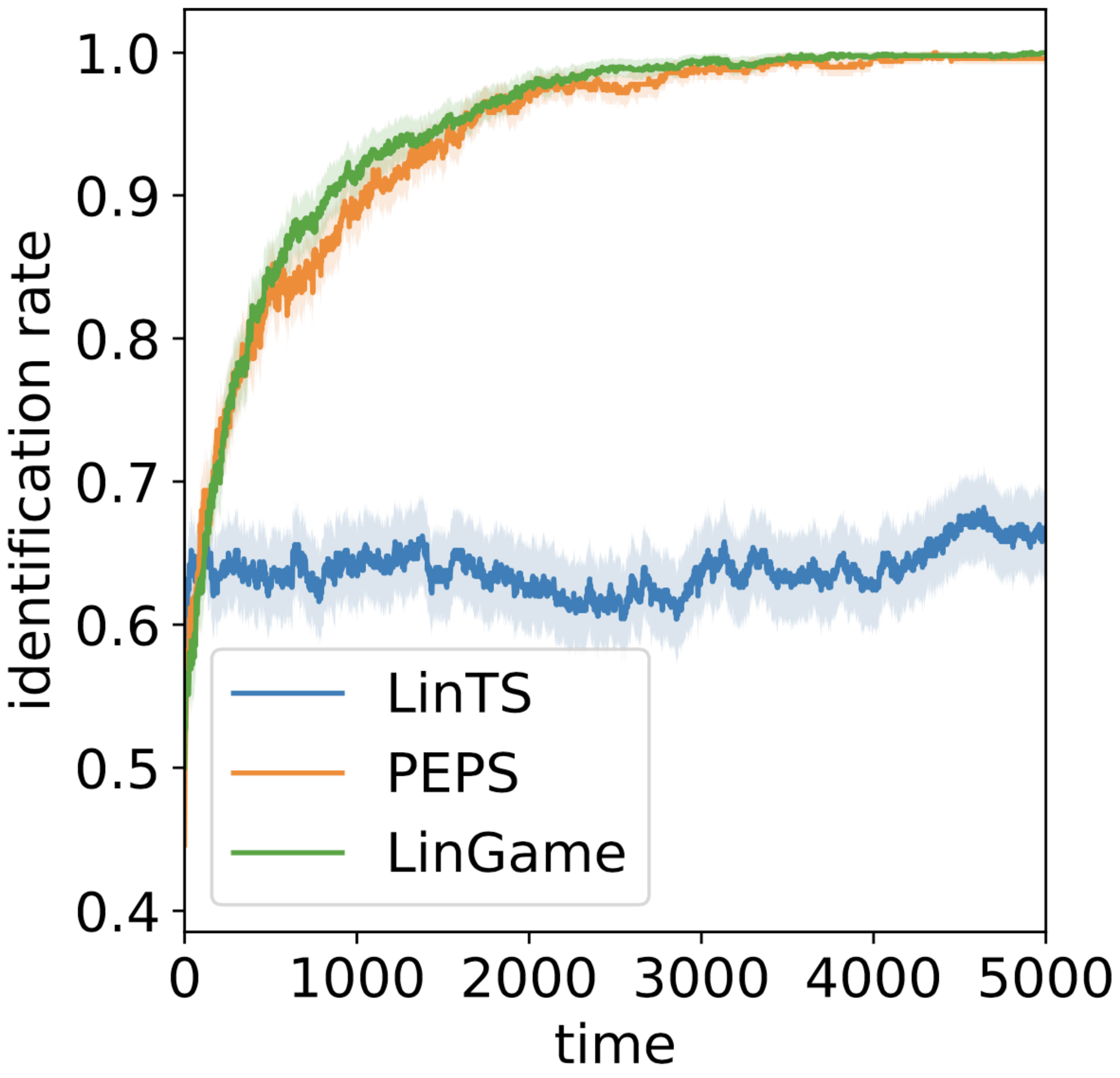
# Experiments: Soare Instance

Hard instance (Soare et al 2014)

$$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \ldots, d$$

$$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$$



# Experiments: Top-K

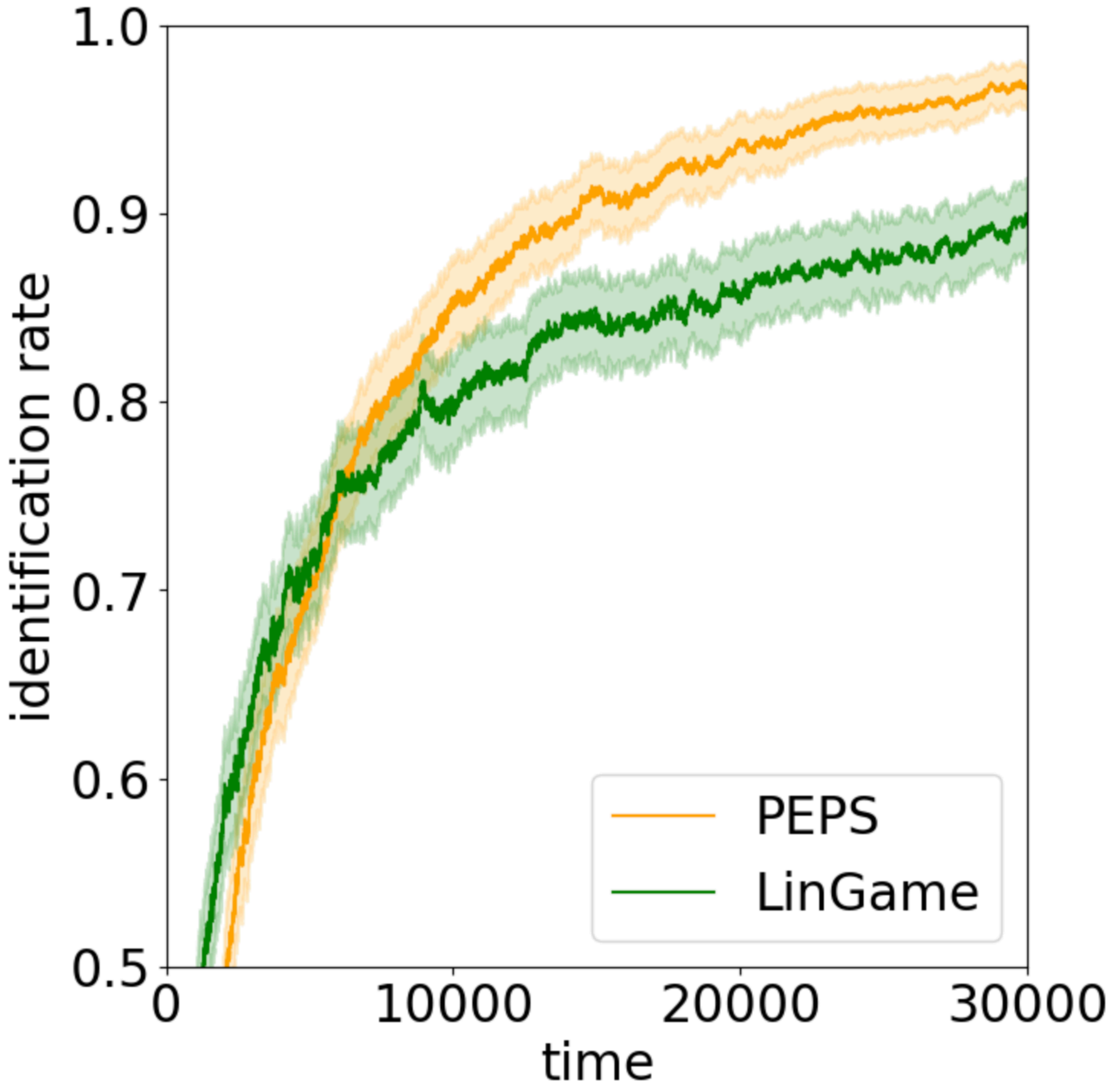
**<u>Goal:</u> identify Top-K arms in X**

$X = \{e_i\}_{i=1}^{12} \subset \mathbb{R}^{12}$

$Z = \left\{ e_{i_1} + e_{i_2} + e_{i_3} : i_1, i_2, i_3 \in \binom{[12]}{3} \right\} \subset \mathbb{R}^{12}$

⇒ identifying Top-3 arms in X is equivalent to identifying the top-one in Z!

$$\theta = [1, .95, .90, \cdots, 1 - .05i, \cdots]$$



# Collaborators

UNIVERSITY *of* WASHINGTON

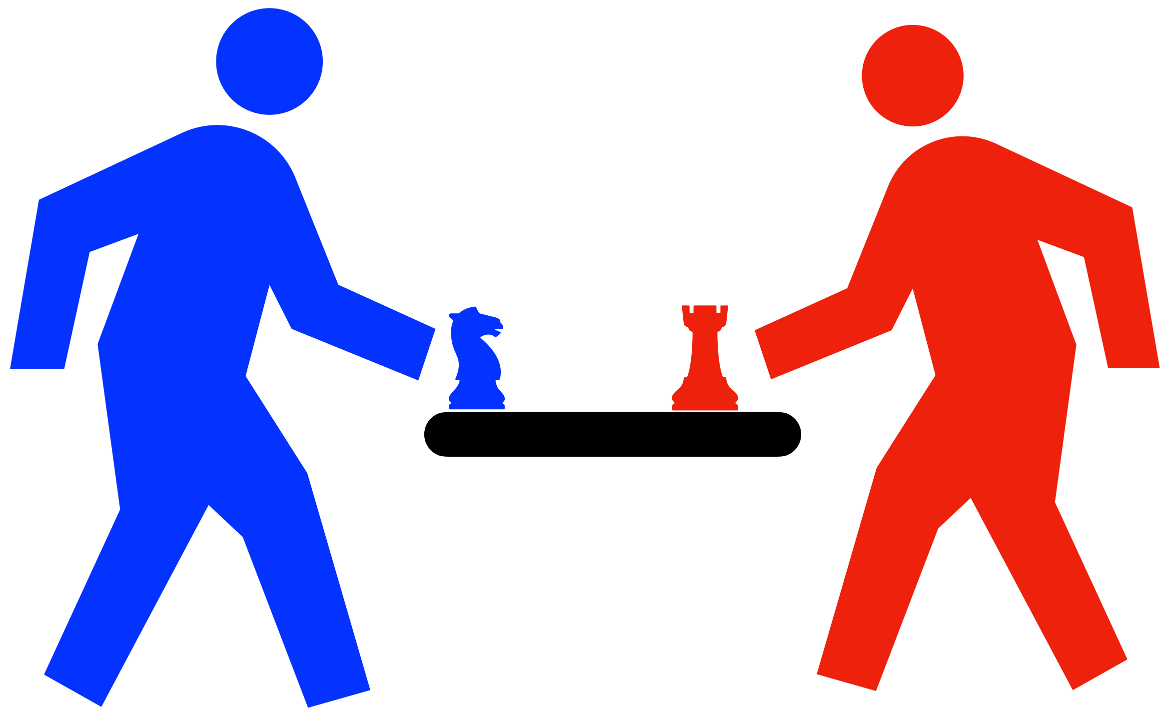
Lalit Jain

Kevin Jamieson

**Thanks!**

# Saddle Point problems

$$\tau_\star = \max_{\lambda \in \Delta_X} \min_{\theta \in \Theta^c_{z_\star}} \|\theta_\star - \theta\|^2_{A(\lambda)} =: \max_{\lambda \in \Delta_X} \min_{\theta \in \Theta^c_{z_\star}} f(\lambda, \theta)$$ $\Rightarrow$ **convex-concave**



**Two-player, zero-sum convex-concave game**

**In each round** $t = 1,2,\cdots$

- max-learner plays $\lambda_t$
- min-learner plays $\theta_t$

# Two player games

**Exponential Weights + Best Response**

Recall $f(\lambda, \theta) = \|\theta_\star - \theta\|^2_{A(\lambda)}$

For $t = 1,2,3,\cdots$

1. **Min-Player:** $\theta_t = \arg\min_{\theta \in \Theta^c_{z_\star}} f(\lambda_t, \theta)$
2. **Max-Player:** **Update** $\lambda_{t+1,x} \propto \lambda_{t,x} e^{\eta[\nabla_\lambda f(\lambda, \theta_t)]_x}$

$$\|\theta_\star - \theta_t\|^2_{A(\lambda_t)} = \min_{\theta \in \Theta^c_{z_\star}} \|\theta_\star - \theta\|^2_{A(\lambda_t)}$$

$$\frac{1}{T}\sum_{t=1}^T \|\theta_\star - \theta_t\|^2_{A(\lambda_t)} \approx \max_{\lambda \in \triangle_\mathcal{X}} \frac{1}{T}\sum_{t=1}^T \|\theta_\star - \theta_t\|^2_{A(\lambda)}$$

**Combining the two above together gives**

$$\tau_\star - \min_{\theta \in \Theta^c_{z_\star}} \|\theta - \theta_\star\|^2_{A(\frac{1}{T}\sum_{t=1}^T \lambda_t)} \le o(1)$$

$\bar{\lambda}_T = \frac{1}{T}\sum_{t=1}^T \lambda_t$ **is an approximate saddle point**

# Computing the Best Response

$$\theta_t = \arg\min_{\theta \in \Theta^c_{z_\star}} \|\theta_\star - \theta\|^2_{A(\lambda_t)}$$

$$= \boxed{\arg\min_{z \neq z_\star, z \in \mathcal{Z}}} \boxed{\arg\min_{\theta \in \Theta_z} \|\theta_\star - \theta\|^2_{A(\lambda_t)}}$$

**Let** $\Theta_z = \{\theta : z = \arg\max_{z \in \mathcal{Z}} z^\top \theta\}$

**projection onto** $\Theta_z$

**search over** $\mathcal{Z}$

**hard to compute when** $\mathcal{Z}$ **is large!**

However, Thompson sampling does not need projections!

Question: can we achieve this?

# Revisiting the Lower Bound

$$\tau_\star = \max_{\lambda \in \Delta_X} \min_{\theta \in \Theta^c_{z_\star}} \|\theta_\star - \theta\|^2_{A(\lambda)}$$

**Challenge: Computing the Min**

$$= \max_{\lambda \in \Delta_X} \min_{p \in \Delta(\Theta^c_{z_\star})} \mathbb{E}_{\theta \sim p}[\|\theta_\star - \theta\|^2_{A(\lambda)}]$$

**Replace projection with a distribution over alternatives**

**<u>Idea: Two player zero-sum game</u>**

1. Max-player: Exponential Weights on $\lambda \in \Delta_X$
2. Min-player: Posterior Updates on $p \in \Delta(\Theta^c_{z_\star})$

# Our Algorithm

Input: X, Z, T, $\eta$

for $t = 1,2,\cdots,T$

1. Compute $z_t = \arg\max_{z \in Z} z^\top \hat{\theta}_t$
2. Sample $\theta_t \sim p_t$, $x_t \sim \lambda_t$
3. Observe $y_t = x_t^\top \theta_\star + \epsilon_t$, update $\hat{\theta}_{t+1}$
4. Update max-player $\lambda_{t+1,x} \leftarrow \lambda_{t,x} e^{-\eta \|\theta_t - \hat{\theta}_t\|^2_{xx^\top}}$
5. Update min-player $p_{t+1,\theta} \propto p_{t,\theta} e^{-\eta \|\theta - \hat{\theta}_t\|^2_{x_t x_t^\top}}$

**Stochastic Gradient**

Distribution over alternatives!

# Exponential Weights as Posterior Sampling

$$p_{t+1,\theta} \propto p_{t,\theta} e^{-\|\theta - \hat{\theta}_t\|^2_{x_t x_t^\top}}$$

$$\propto e^{-\sum_{s=1}^{t} \|\theta - \hat{\theta}_s\|^2_{x_s x_s^\top}}$$

**if $\hat{\theta}_s$ is not changing much between rounds**

$$\propto N(\hat{\theta}_t, (\sum_{s=1}^{t} x_s x_s^\top)^{-1})$$

**(restricted to $\Theta^c_{\hat{z}_t}$)**

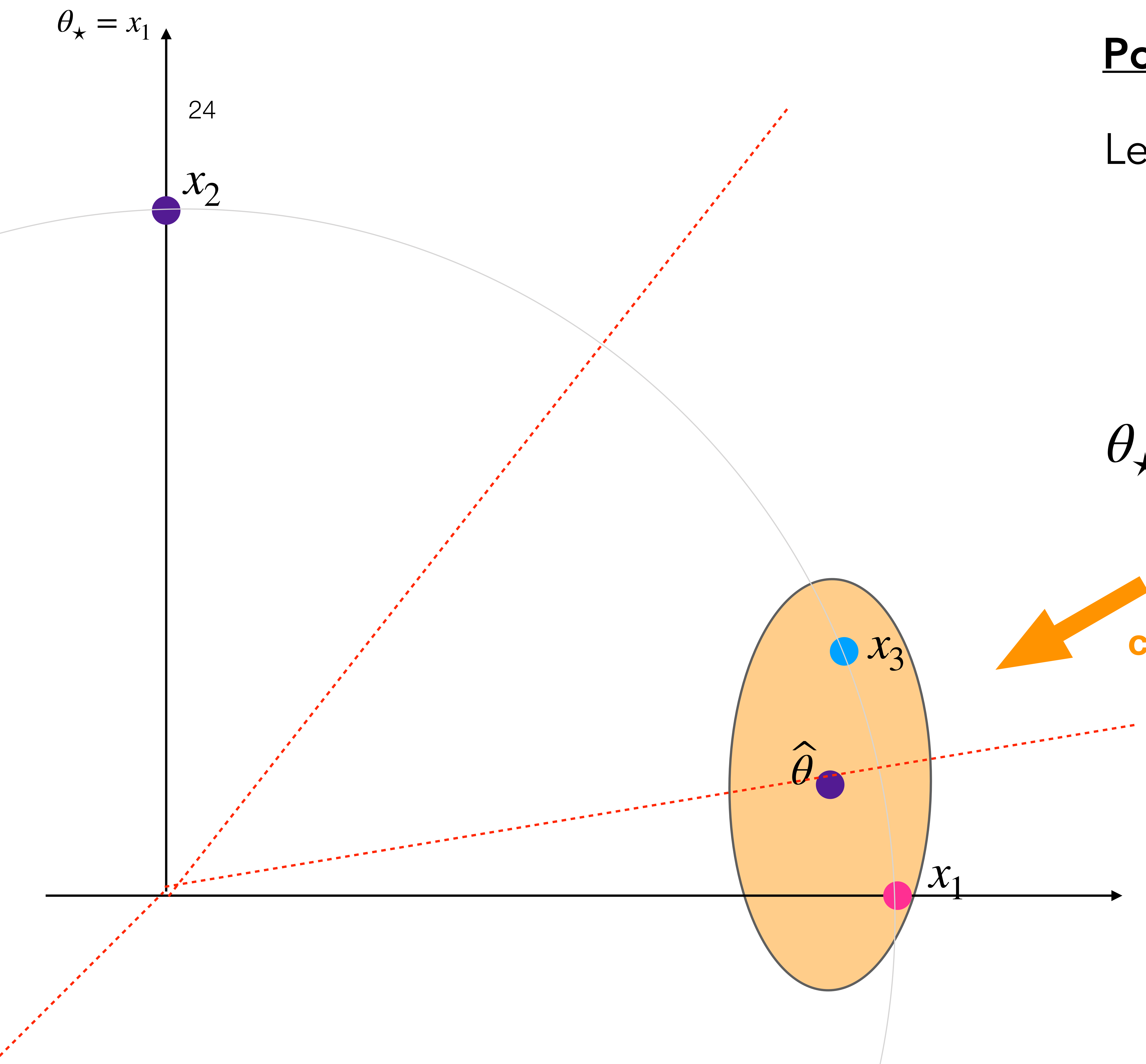
Sampling from $p_t$ can be (approximately) done using rejection sampling from a Gaussian!

Don't need to maintain $p_{t,\theta}, \theta \in \Theta^c_{\hat{z}_t}$

# Sub-Optimality of TS for BAI

$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \ldots, d$

$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$



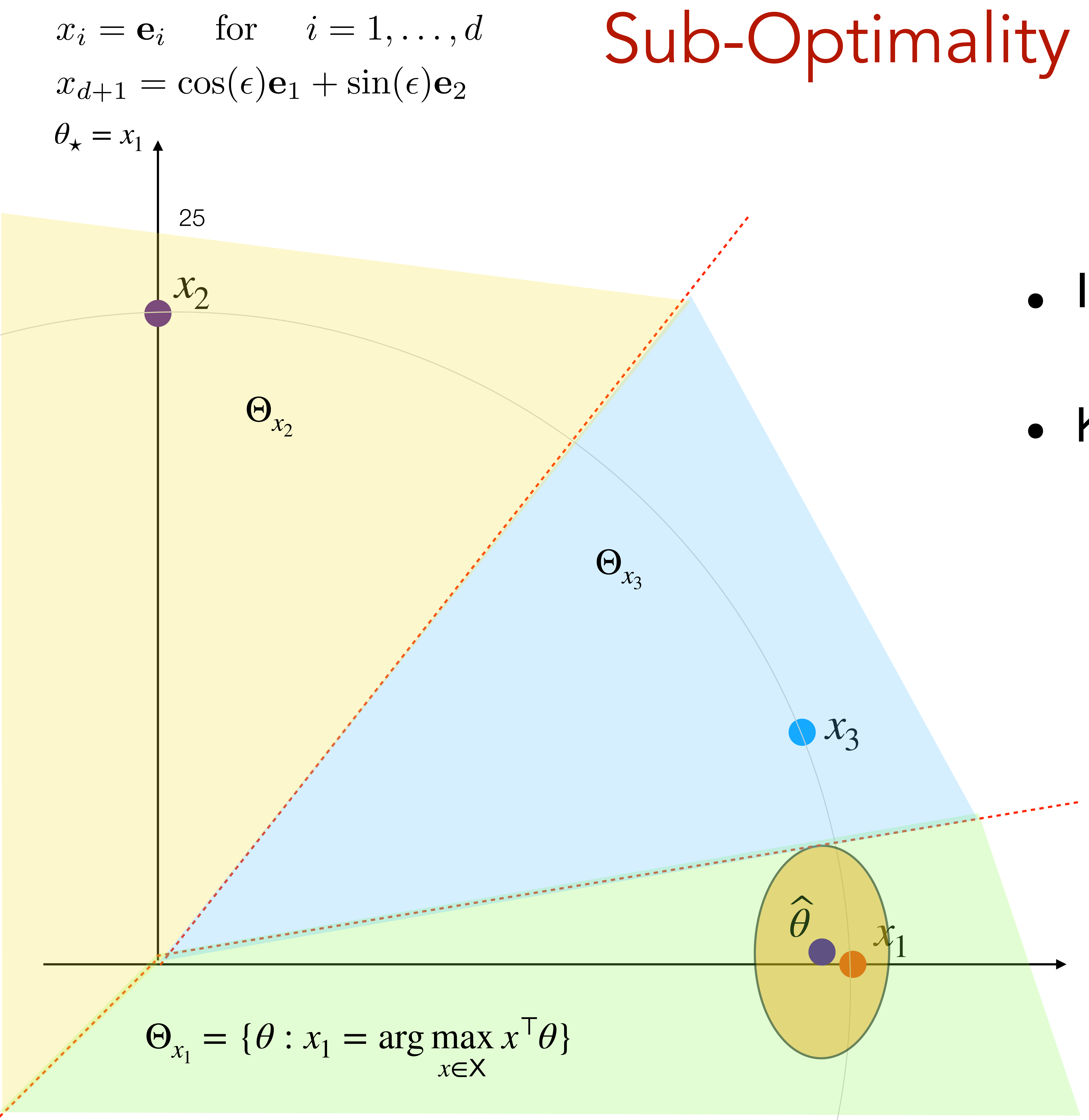

**Posterior**

Let $V_t = \sum_{s=1}^{t} x_s x_s^\top$, then $\Pi_t = N\left(\hat{\theta}_t, V_t^{-1}\right)$

$$\theta_\star \in \mathrm{S} = \left\{\theta : \|\hat{\theta}_t - \theta\|_{V_t}^2 \le O(\sqrt{d \log(t/\delta)})\right\}$$

confidence set for $\theta_*$

# Sub-Optimality of TS for BAI

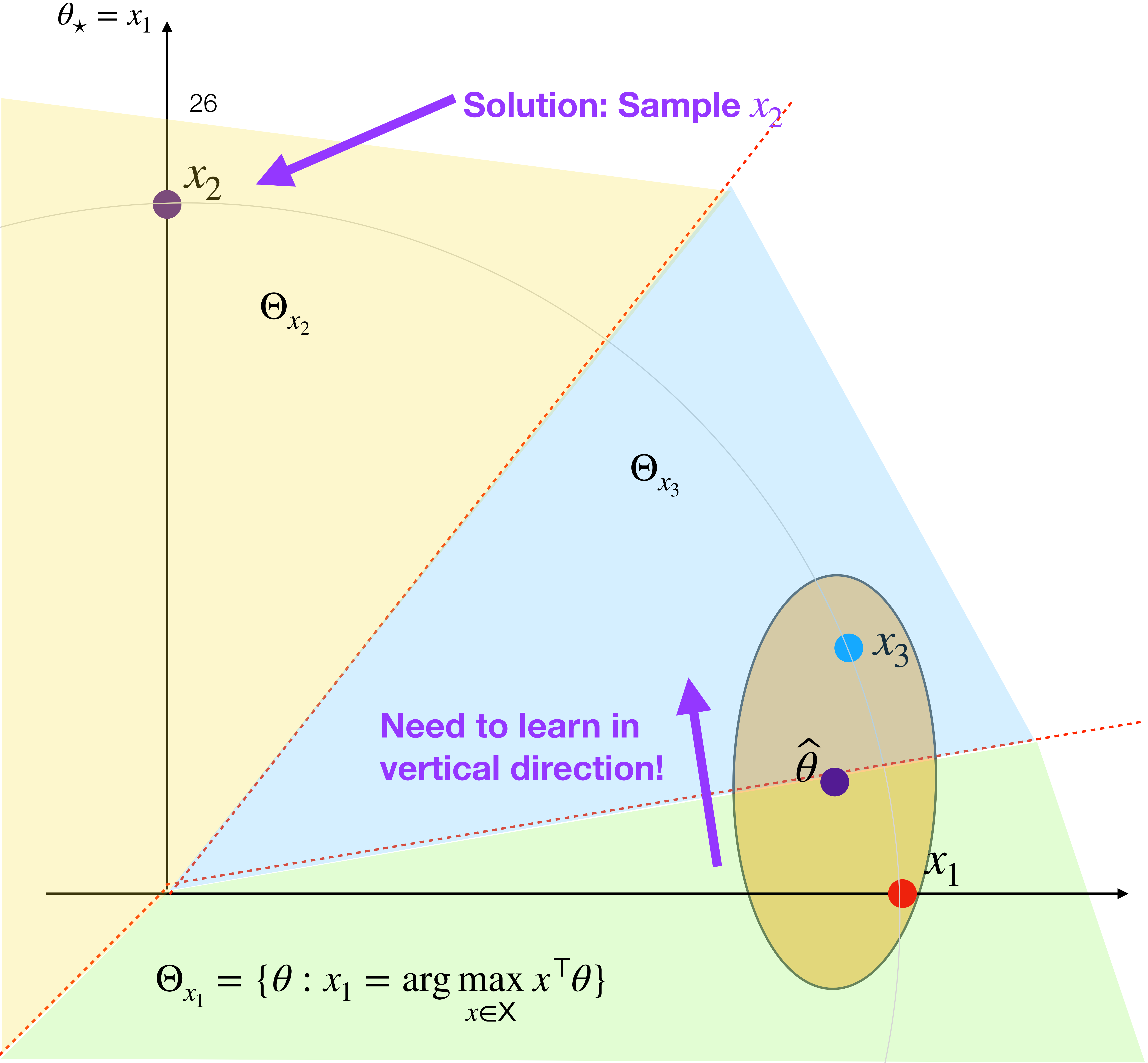



- If $S \subset \Theta_{x_1}$, returns $x_1$ as the best arm
- Key: distinguish between $x_1$ and $x_3$!

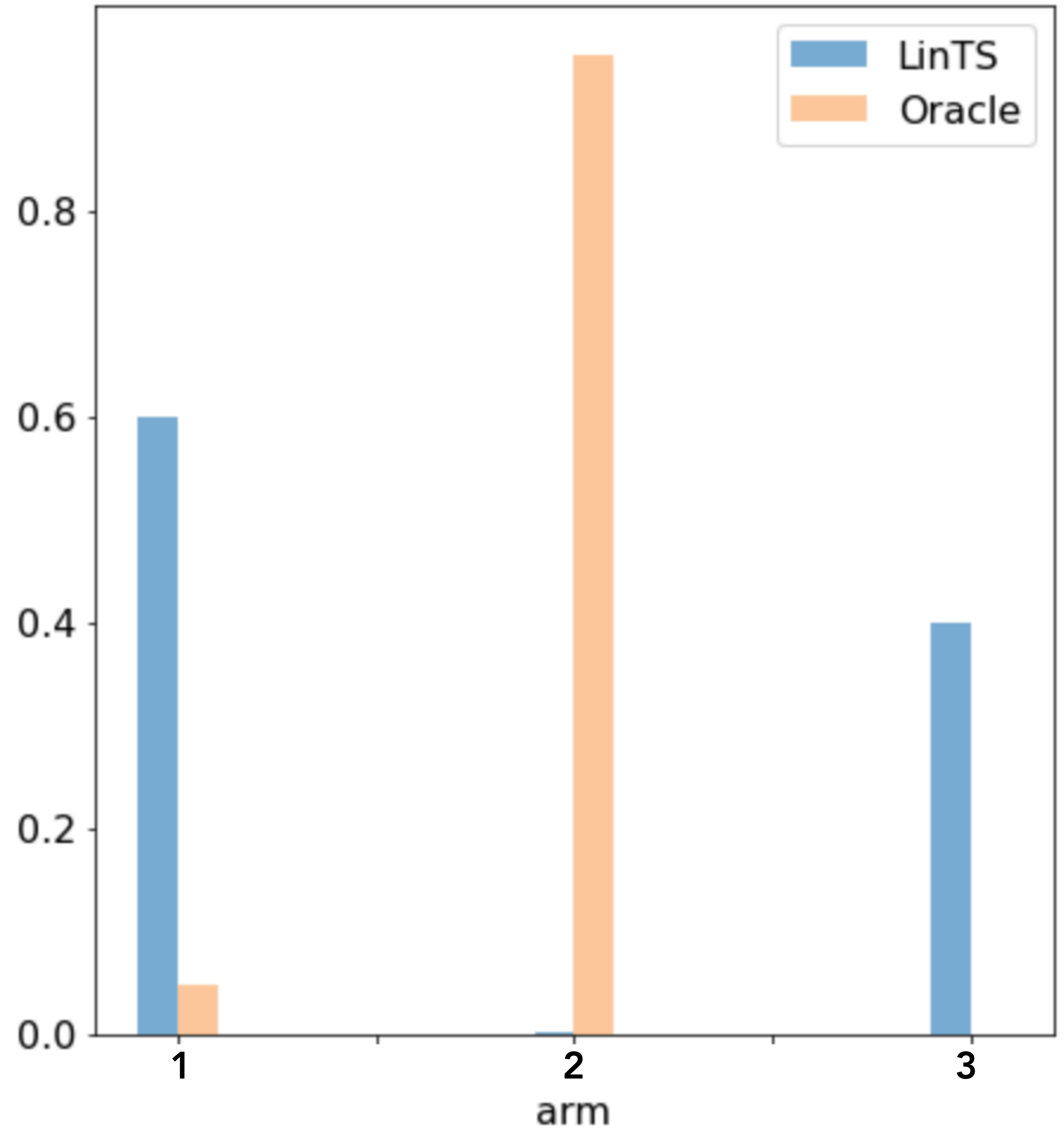
# Sub-Optimality of TS for BAI

$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \ldots, d$

$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$



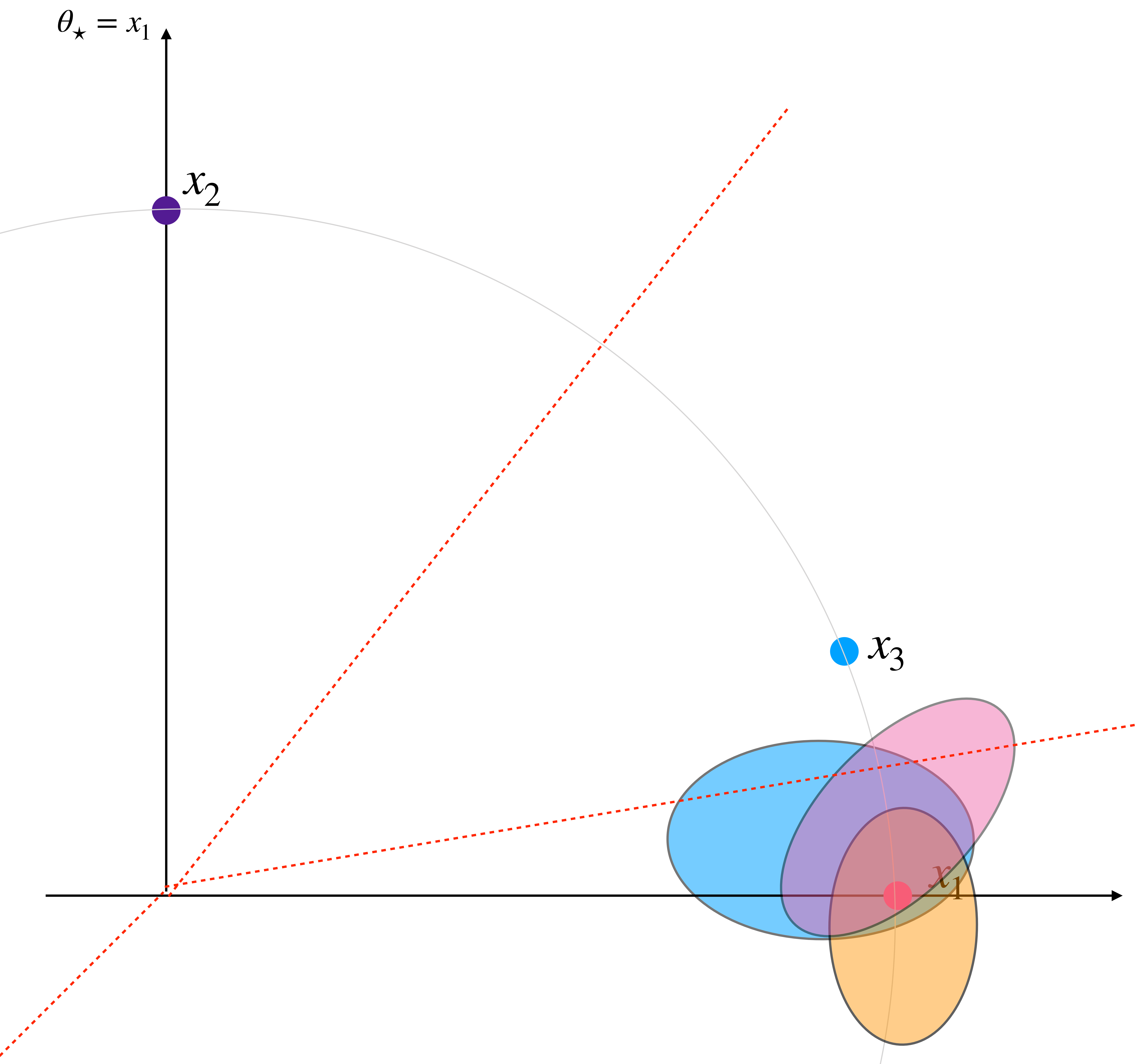

- However, Thompson sampling tends to pull arm $x_1$ or $x_3$ much more than $x_2$



# Lower Bound: Oracle Strategy

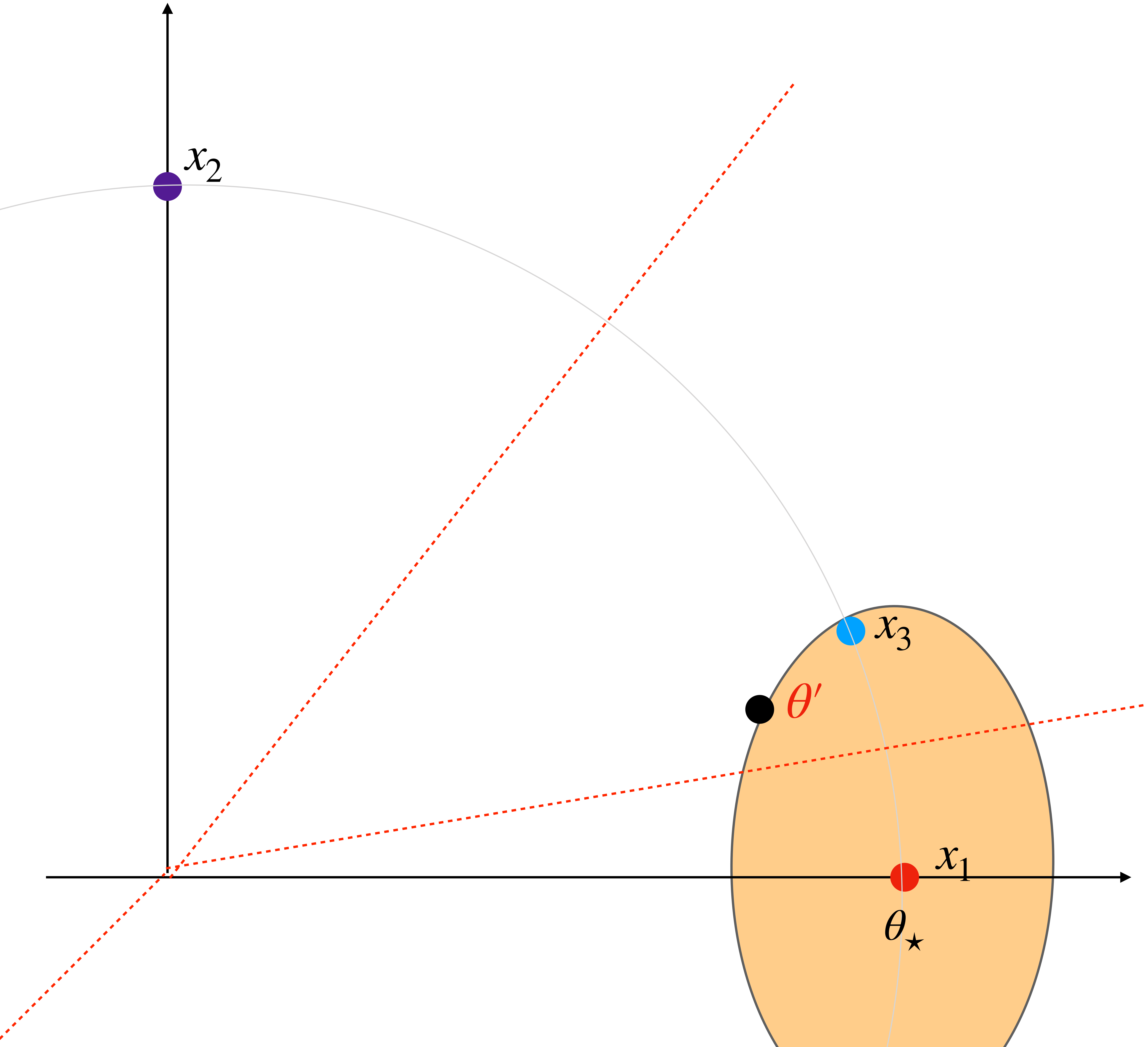
$x_i = \mathbf{e}_i \quad \text{for} \quad i = 1, \dots, d$

$x_{d+1} = \cos(\epsilon)\mathbf{e}_1 + \sin(\epsilon)\mathbf{e}_2$




**Planning Problem:** What is the best sampling distribution to quickly shrink the posterior into the correct region?

# Lower Bound: Oracle Strategy




Want to reject the possibility that $\theta'$ is the true parameter

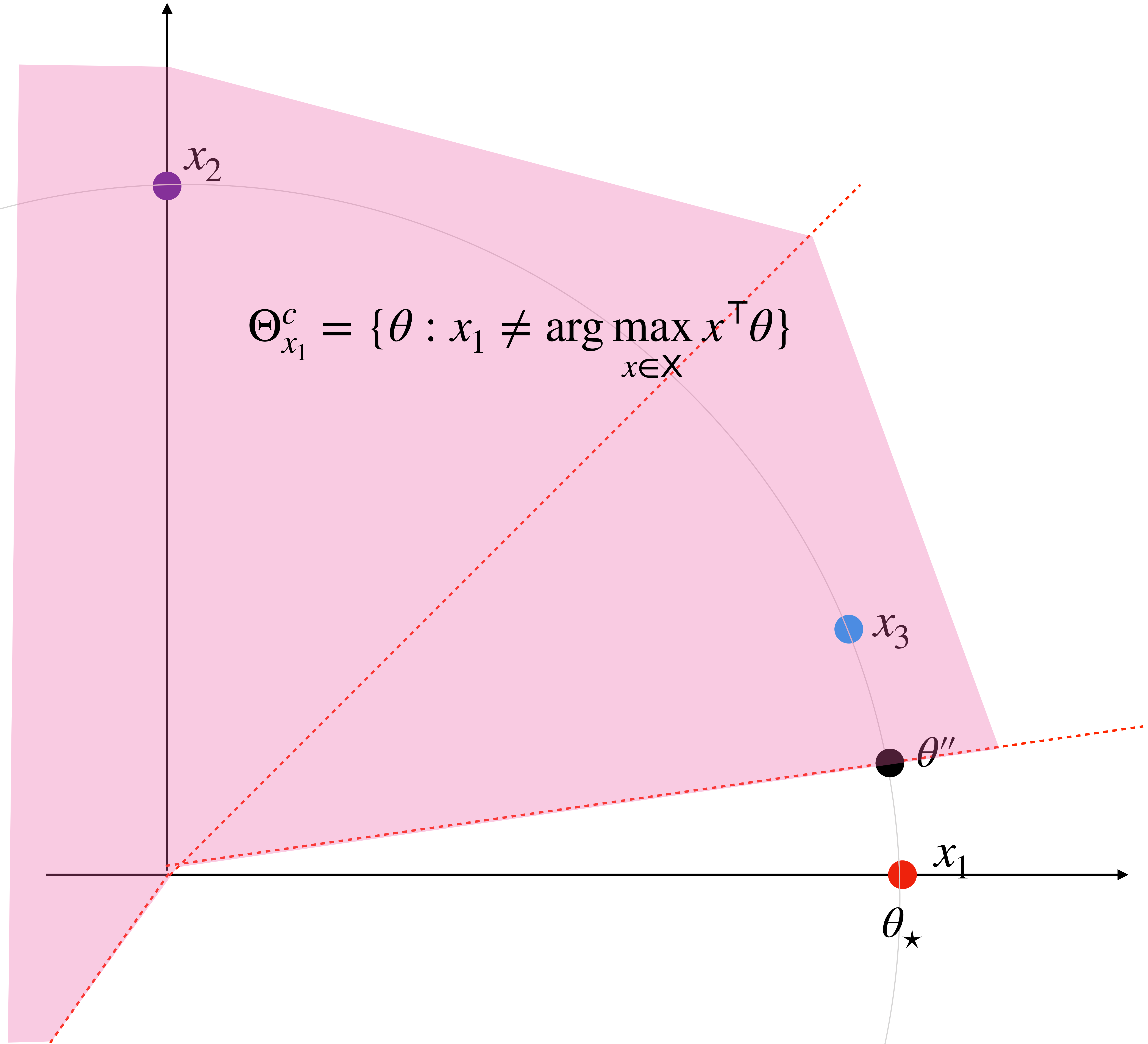
$\Rightarrow$ distinguish between $N(\theta', V_t^{-1})$ and $N(\theta_\star, V_t^{-1})$

Design samples $X = [x_1, \cdots, x_t]$

$$\max_X \|\theta_\star - \theta'\|_{V_t}^2$$

**KL Divergence**

# Lower Bound: Oracle Strategy




<u>Choose</u> samples $X = [x_1, \cdots, x_t]$

$$\max_{X} \min_{\theta \in \Theta_{x_1}^c} \|\theta_\star - \theta\|_{V_t}^2$$

$$\Theta_{x_1} = \{\theta : x_1 = \arg\max_{x \in \mathcal{X}} x^\top \theta\}$$

$\Rightarrow$ The sampling distribution $\lambda \in \triangle_{\mathcal{X}}$ should be the (argmax) solution to

$$\tau_* = \max_{\lambda \in \Delta_{\mathcal{X}}} \min_{\theta \in \Theta_{x_1}^c} \|\theta_\star - \theta\|_{A(\lambda)}^2$$

Where $A(\lambda) = \sum_{x \in \mathcal{X}} \lambda_x x x^\top$

# Experiments: Sphere Instance

$\mathsf{X} \subset B^6, |\mathsf{X}| = 20$

$\theta_* = x + .01(x' - x)$