Information-theoretic Online Memory Selection for Continual Learning

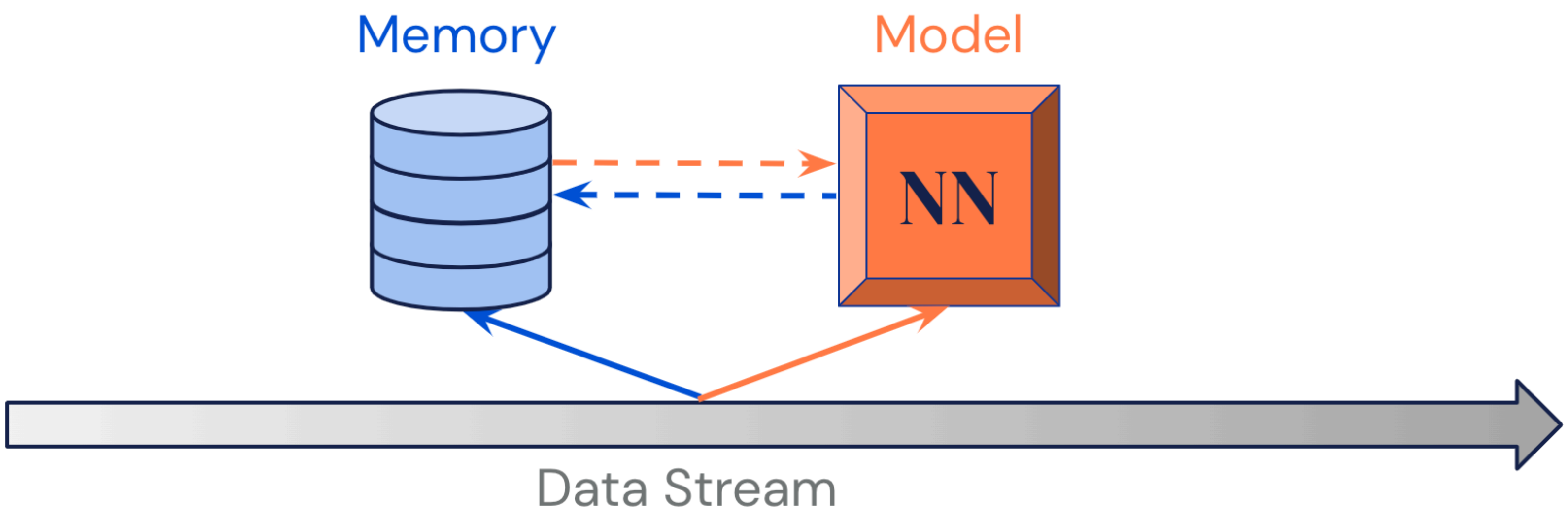
Shengyang Sun, Daniele Calandriello, Clara Huiyi Hu, Ang Li, Michalis Titsias

Online Memory Selection

- Selecting a representative memory is critical to replay-based continual learning methods.

- The agent updates both the memory and the model based on the instant observation,

$$(f_\theta, \mathcal{M}) \longleftarrow (f_\theta, \mathcal{M}, (\mathbf{x}_\star, y_\star))$$



Online Memory Selection

- Challenges
  - The purely online constraint calls for both effectiveness and efficiency.

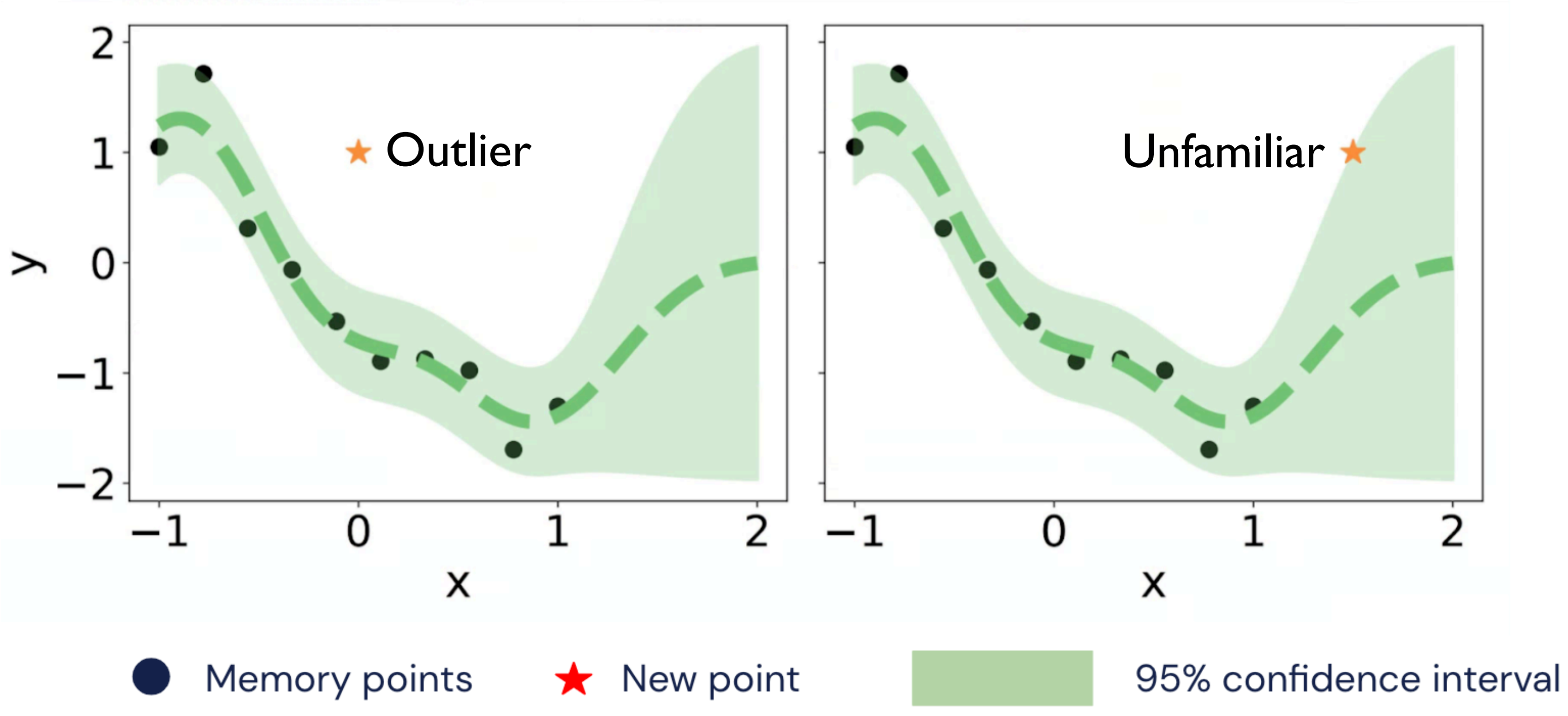
  - To select a representative memory needs to deal with data imbalance.

- $\mathcal{M}$ should contain the most information about the underlying function.

Memorable Information Criterion

- Surprise: find "surprising" data points

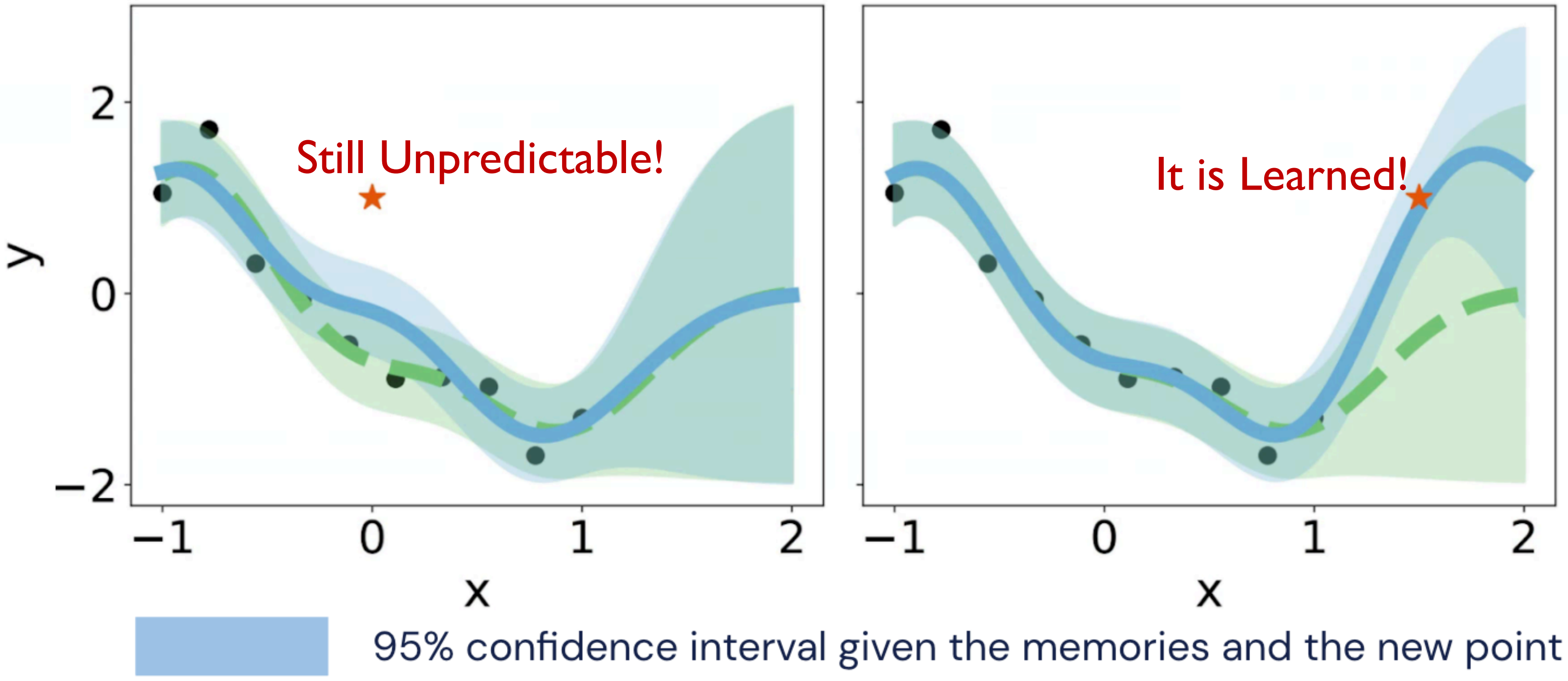
$$s_{surp}\big((x_\star, y_\star); \mathcal{M}\big) = -\log p(y_\star | y_\mathcal{M})$$



Memorable Information Criterion

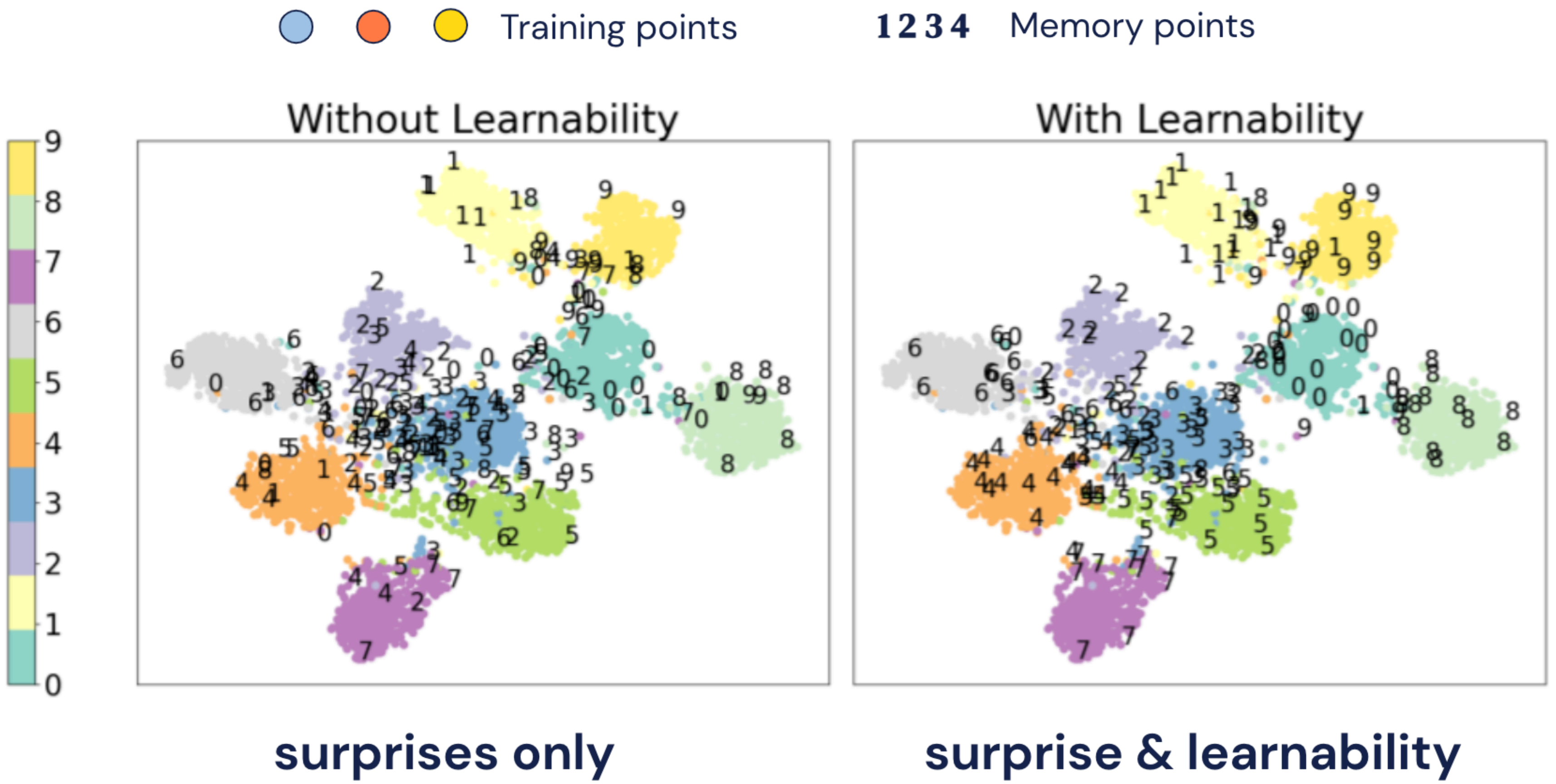
- Learnability: avoid "harmful" outliers

$$s_{\text{learn}}\big((x_\star, y_\star); \mathcal{M}\big) = \log p(y_\star | y_\star, y_\mathcal{M})$$



Memorable Information Criterion

- Surprise finds unfamiliar points.
- Learnability avoids outliers.



Information-theoretic Reservoir Sampling

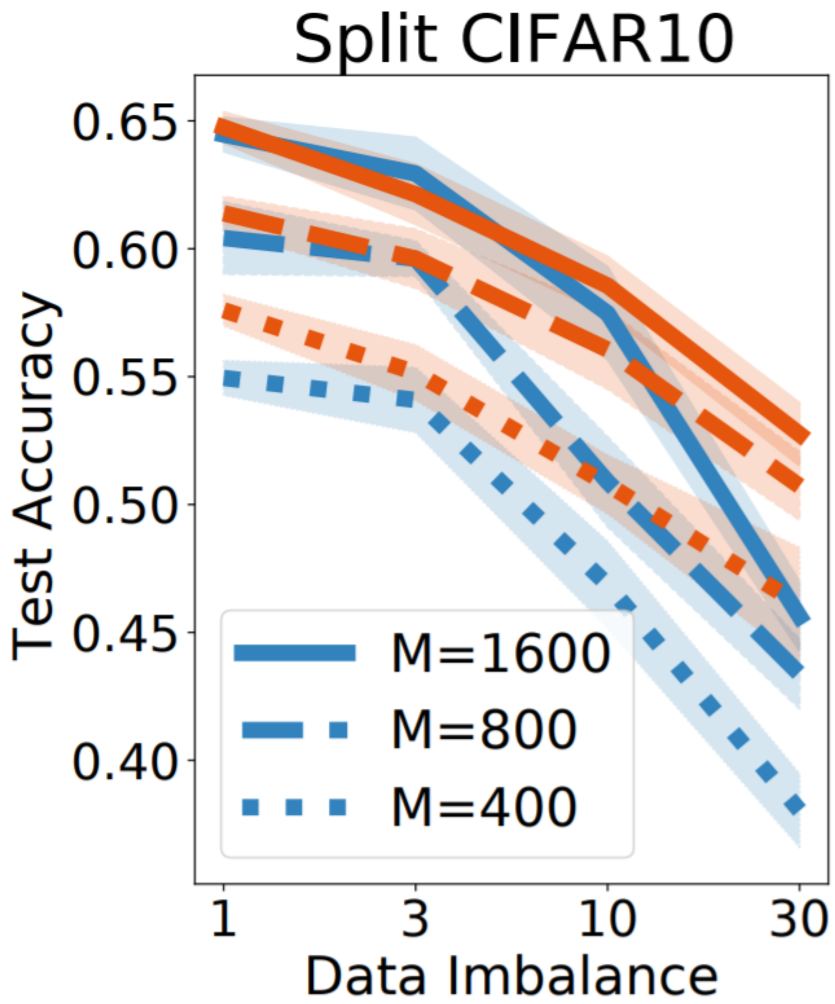
- InfoRS: a modification of Reservoir Sampling (Vitter, 1985) to select informative points only.

Algorithm 1 Information-theoretic Reservoir Sampling (InfoRS)

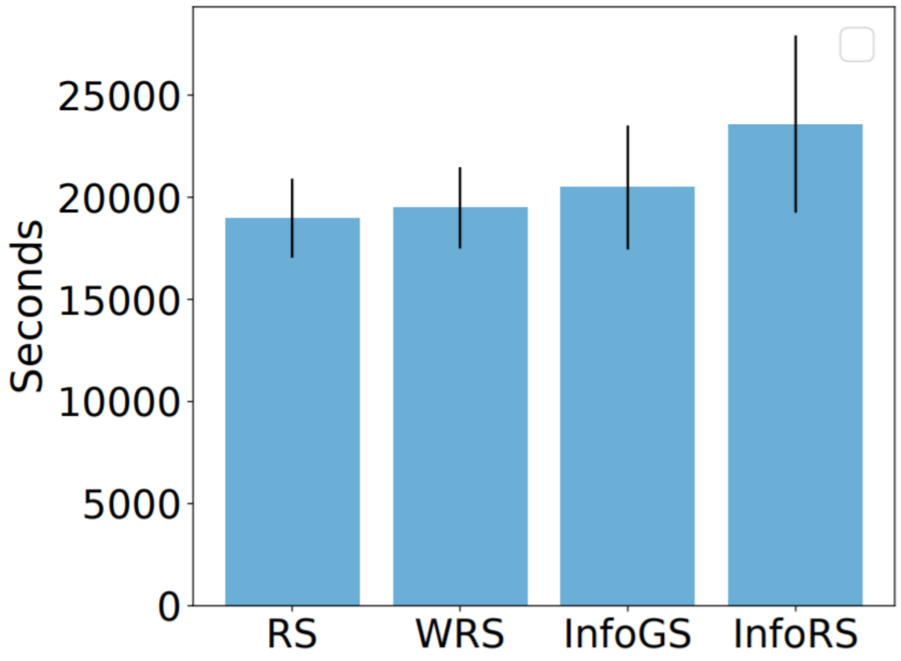
Input: Memory $\mathcal{M}$ and matrices $\mathbf{A}_{\mathcal{M}}^{-1}, \mathbf{b}_{\mathcal{M}}$, the batch $\mathcal{B}$, the predictor parameter $\boldsymbol{\theta}$.
Input: The reservoir count n and the budget M.
Input: Running mean and stddev for the MIC: $\hat{\mu}_i, \hat{\sigma}_i$. The thresholding ratio γ_i.
Update the predictor parameter $\boldsymbol{\theta}$ based on $\mathcal{M}$ and $\mathcal{B}$.[3] // Predictor Update
Update the features for the memory points used in replay, and update $\mathbf{A}_{\mathcal{M}}^{-1}, \mathbf{b}_{\mathcal{M}}$ accordingly.
for $(\mathbf{x}_\star, y_\star)$ in $\mathcal{B}$ **do**
    if $|\mathcal{M}| < M$ **or** $\mathrm{MIC}_\eta((\mathbf{x}_\star, y_\star); \mathcal{M}) \geq \hat{\mu}_i + \hat{\sigma}_i * \gamma_i$ // Information Thresholding
        Update $\mathcal{M}, n \longleftarrow$ **ReservoirSampling**$(\mathcal{M}, M, n, (\mathbf{x}_\star, y_\star))$. // Memory Update
        Update $\mathbf{A}_{\mathcal{M}}^{-1}, \mathbf{b}_{\mathcal{M}}$ based on the Sherman-Morrison formula if $\mathcal{M}$ is updated.
    Update $\hat{\mu}_i, \hat{\sigma}_i$ using the criterion $\mathrm{MIC}_\eta((\mathbf{x}_\star, y_\star); \mathcal{M})$. // Running Moments Update
return Buffer $\mathcal{M}$ and $\mathbf{A}_{\mathcal{M}}^{-1}, \mathbf{b}_{\mathcal{M}}$. The reservoir count n and statistics $\hat{\mu}_i, \hat{\sigma}_i$. The updated $\boldsymbol{\theta}$.

Continual Learning Experiments

Improved robustness against data imbalance



Minimal Computational Overhead



Thanks!

Improved robustness against data imbalance in online memory selection