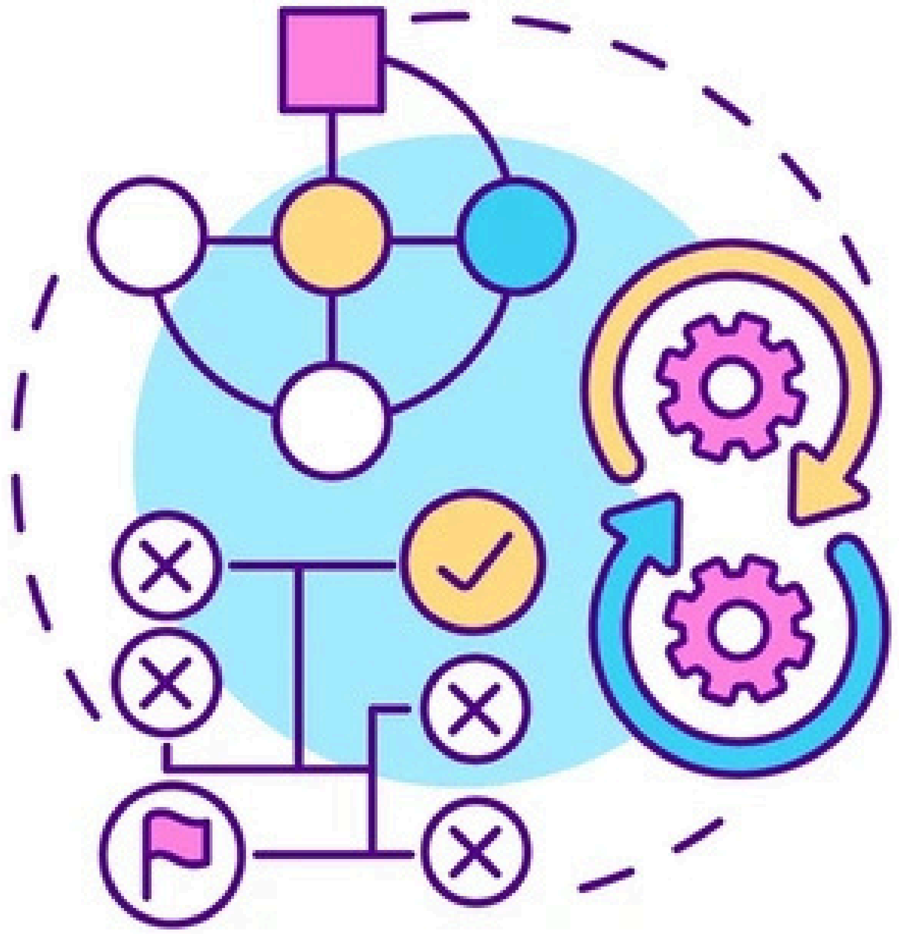


[RE] CROSS-VALIDATED OFF-POLICY EVALUATION

METE HARUN AKCAY

MASA CIRKOVIC

ALEXIS GBECKOR-KOVE

AGENDA

Cross-Validated Off-Policy Evaluation

Matej Cief[1,2], Branislav Kveton[3], Michal Kompan[2]

[1]Brno University of Technology
[2]Kempelen Institute of Intelligent Technologies
[3]Amazon*

Abstract

In this paper, we study the problem of estimator selection and hyper-parameter tuning in off-policy evaluation. Although cross-validation is the most popular method for model selection in supervised learning, off-policy evaluation relies mostly on theory-based approaches, which provide only limited guidance to practitioners. We show how to use cross-validation for off-policy evaluation. This challenges a popular belief that cross-validation in off-policy evaluation is not feasible. We evaluate our method empirically and show that it addresses a variety of use cases.

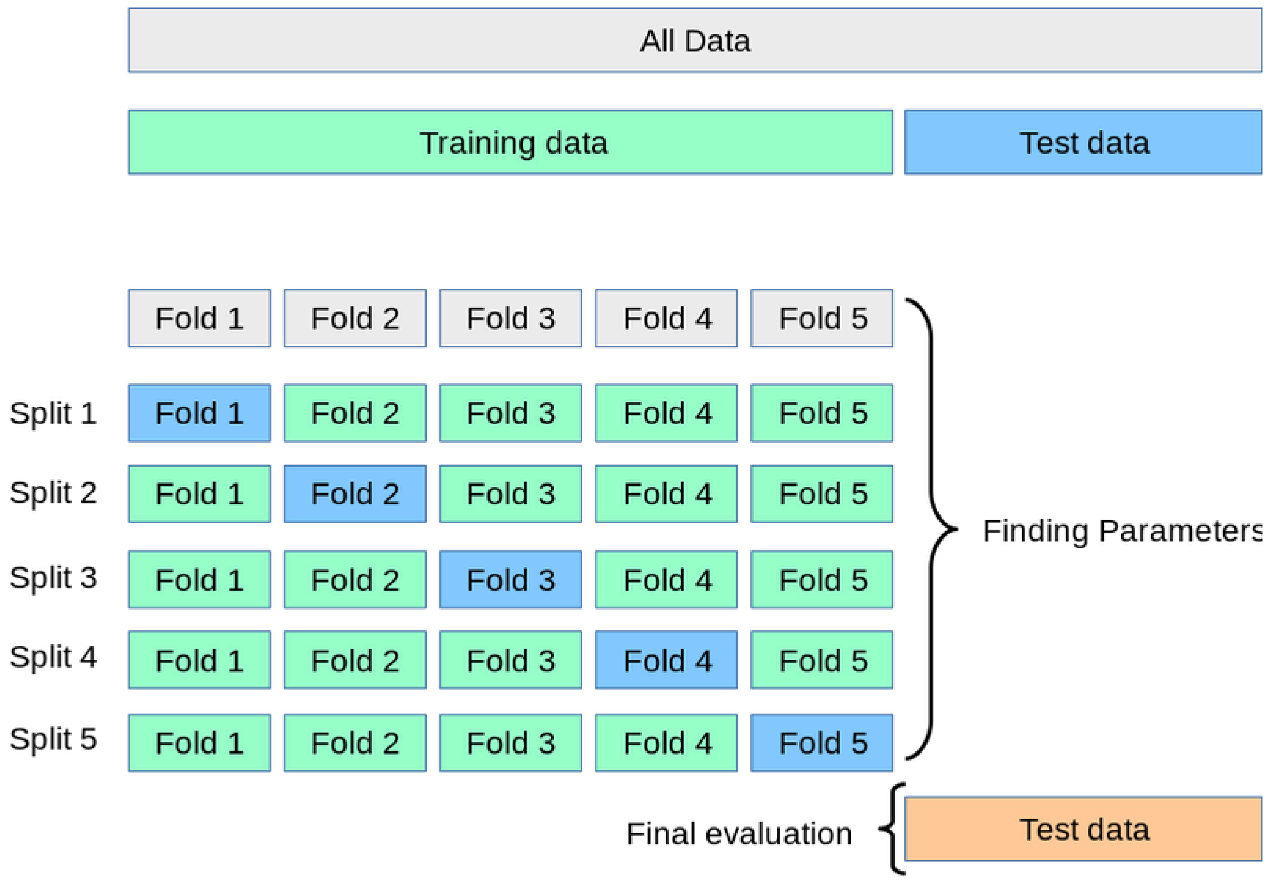
INTRODUCTION

Off-policy evaluation is a framework for estimating the performance of a policy without deploying it online.

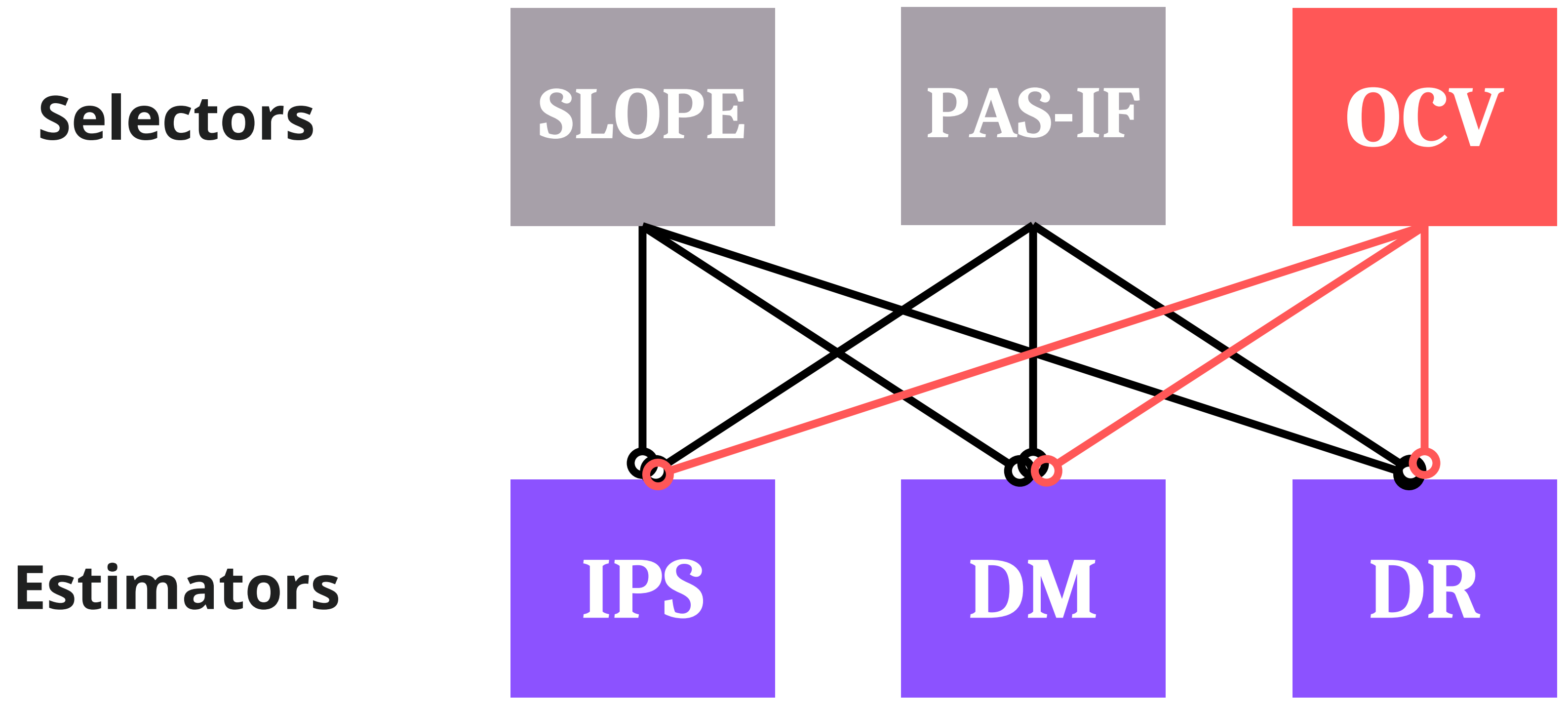
It is useful where online A/B testing is costly or too dangerous.

- recommendation systems
- medical treatments

SUPERVISED LEARNING



OFF-POLICY EVALUATION



SCOPE OF REPRODUCIBILITY

Main claims of the paper:

- Cross-validation-based estimator selection (OCV) can reliably choose a suitable estimator among IPS, DM and DR, demonstrating better performance in multiple datasets.
- OCV performs well even when the validation estimator does not directly match the best estimator.
- OCV serves as a general solution for hyper-parameter tuning and joint estimator selection, achieving comparable or superior performance to theory-based methods across various estimators.

Additional findings of the paper:

- Their improvements make standard cross-validation more stable.
- The validator used in cross-validation has to be unbiased to block the optimization objective shifting to prefer the estimators biased in the same direction.
- Cross-validation is computationally efficient.

METHODOLOGY

ENVIRONMENT SETUP

SOFTWARE REQUIREMENTS

Python v3.10+
pyyaml
latex

HARDWARE

8-core Intel i9-9900K
(16) @ 5.000GHz
32GB RAM
NVIDIA RTX 2080 Ti GPU

COMMANDS & CONFIGS

dr_strong.yaml
dr_weak.yaml
tuning.yaml
ablation
k_splits.yaml
ocv_dm.yaml

METHODOLOGY

DATASETS

- Nine (9) Datasets
- Each Dataset split into two subsets
  - -> Bandit feedback
  - -> Policy learning

DATASET	CLASSES	FEATURES	SAMPLE SİZE
Ecoli	8	7	336
Glass	6	9	214
Letter	26	16	20,000
Optdigits	10	64	5,620
Page-blocks	5	10	5,473
Pendigits	10	16	10,992
Satimage	6	36	6,435
Vehicle	4	18	846
Yeast	10	8	1,484

METHODOLOGY

ESTIMATORS

ESTIMATOR	SHORT DESCRIPTION
Inverse Propensity Score (IPS)	Unbiased, high variance
Direct Method (DM)	Lower variance, potential bias
Doubly Robust (DR)	Combines IPS and DM to reduce variance
SLOPE	Estimator selection based on variance ordering
PAS-IF	Creates surrogate policies from logged data
OCV	Cross-validation-based estimator selection

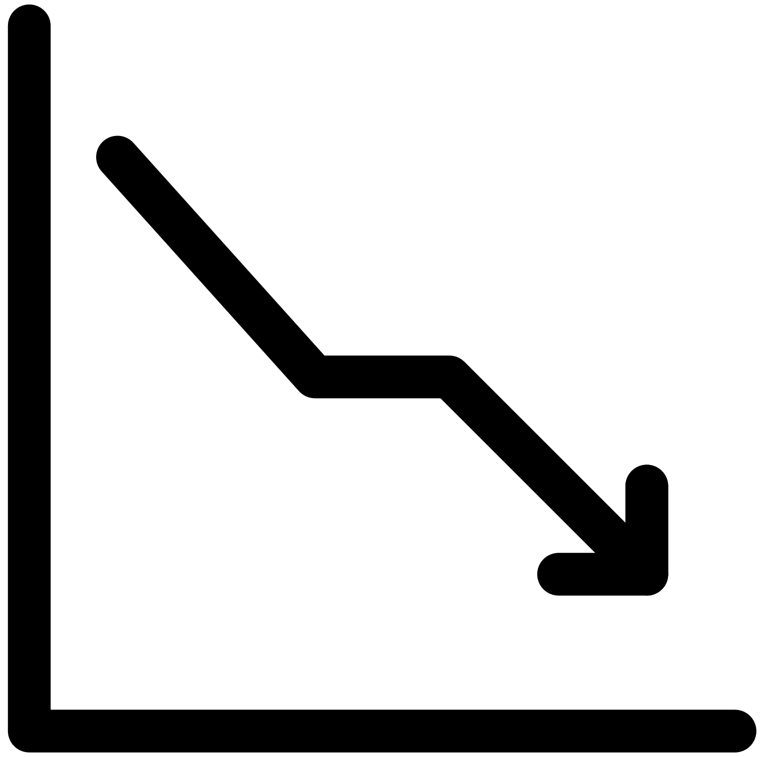
METHODOLOGY

HYPERPARAMETER TUNING

HYPERPARAMETER	DESCRIPTION
IPS Clipping Constant	Clipping threshold for variance reduction in IPS.
DM Regression Model	Regression model choice impacting DM's performance
DR Combination	Balance between IPS and DM to achieve optimal variance reduction

METHODOLOGY

EVALUATION METRICS



Mean Squared Error

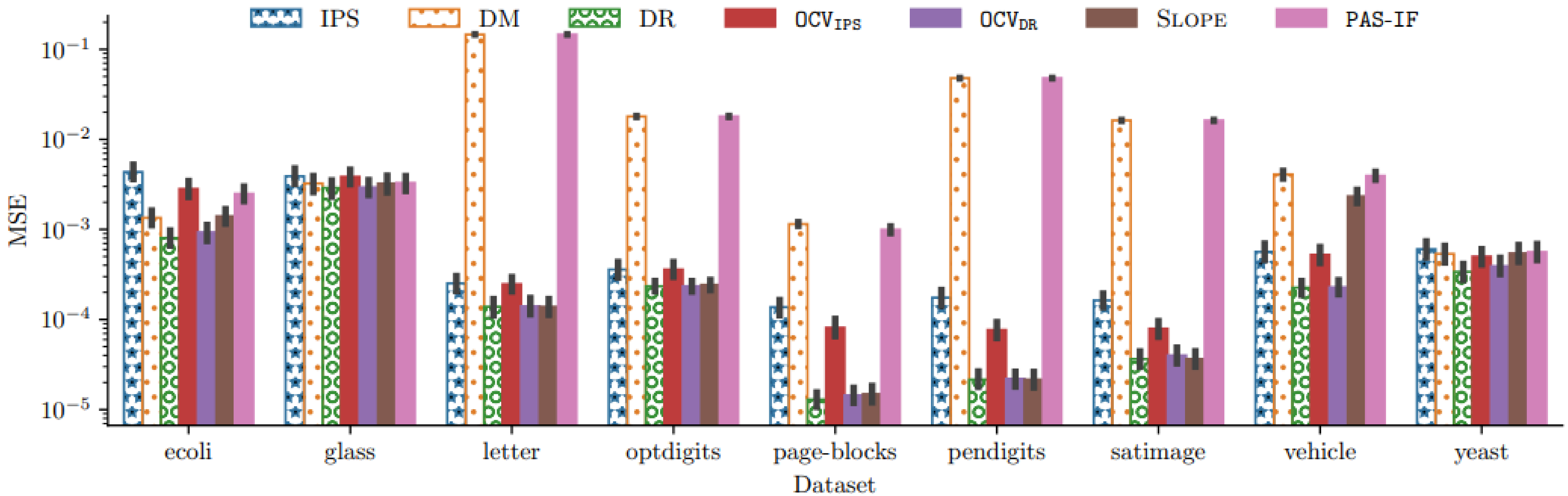
Runtime

Ranged from 6 to 77 hours

RESULTS

1. **Cross-validation consistently chooses a good estimator**

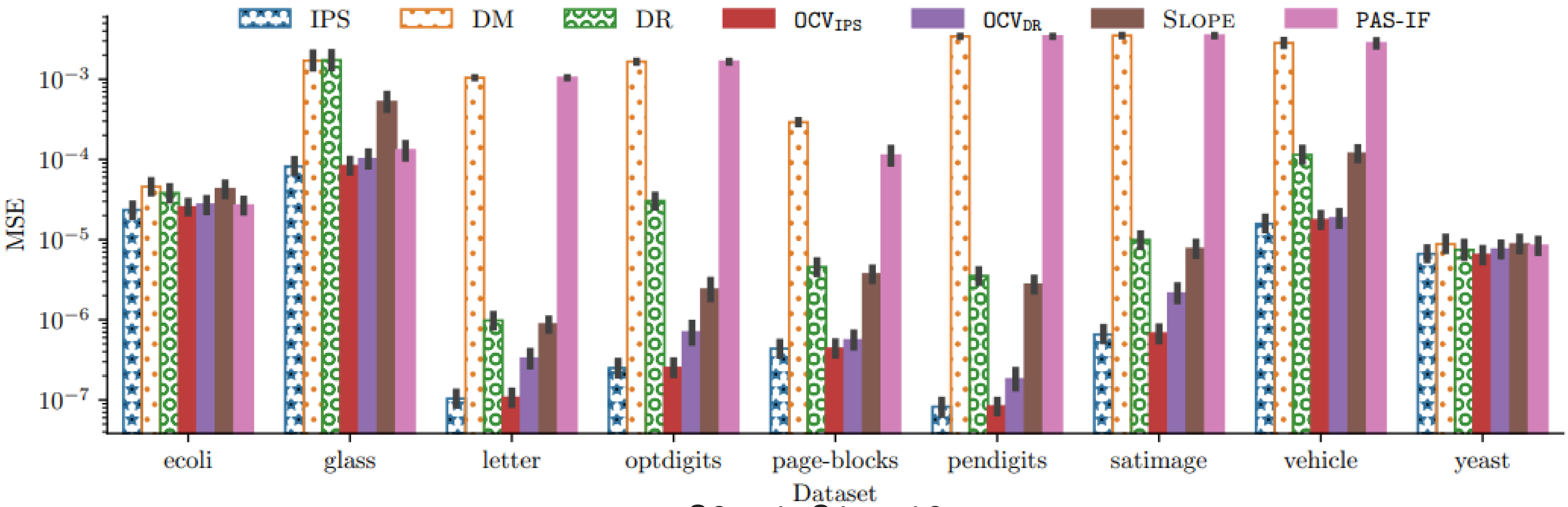
- Took 77h45min to reproduce
- OCVdr significantly outperforms others on page-blocks and vehicle
- It is never significantly worse performing
- Both OCVs outperform the other two
- PAS-IF chose DM which is a biased validator thus it chooses biased estimator (DM)



RESULTS

2. **Cross-validation with DR performs well even when DR performs poorly**

- OCV dr performs well just because DR is also the best estimator proven FALSE
- Temperature of the target policy changed to -10
- Both OCVs outperform the other two
- Slope performs poorly here



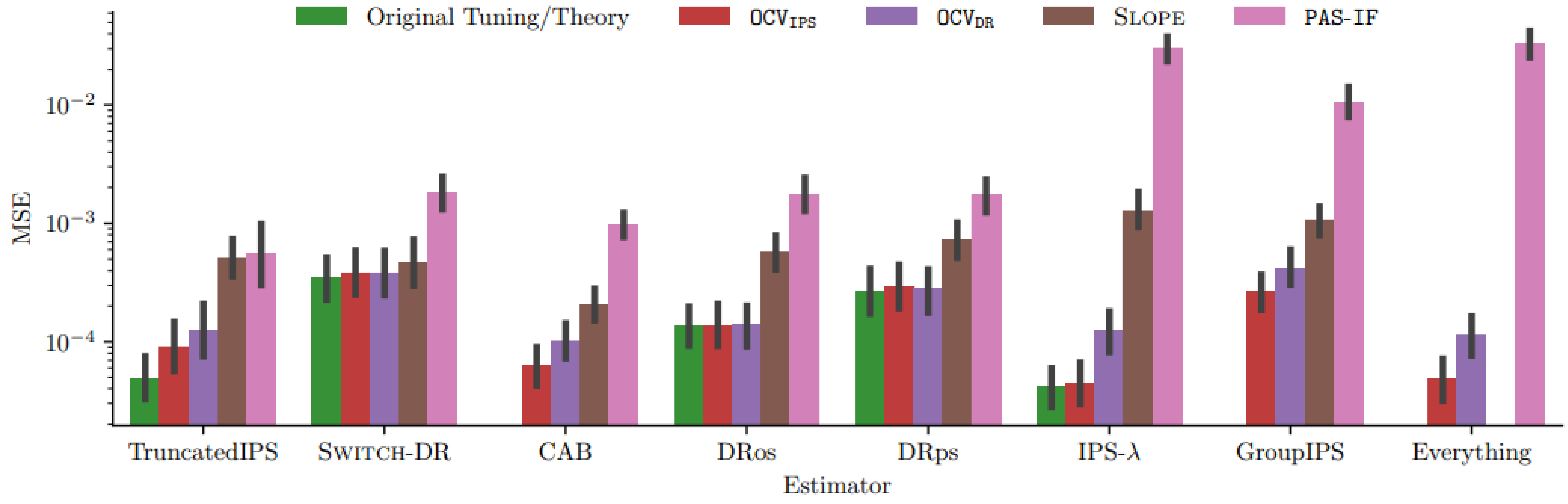
$\beta 0 = 1, \beta 1 = -10$

RESULTS

?

3. **OCV provides a robust solution for hyper-parameter tuning and estimator selection**

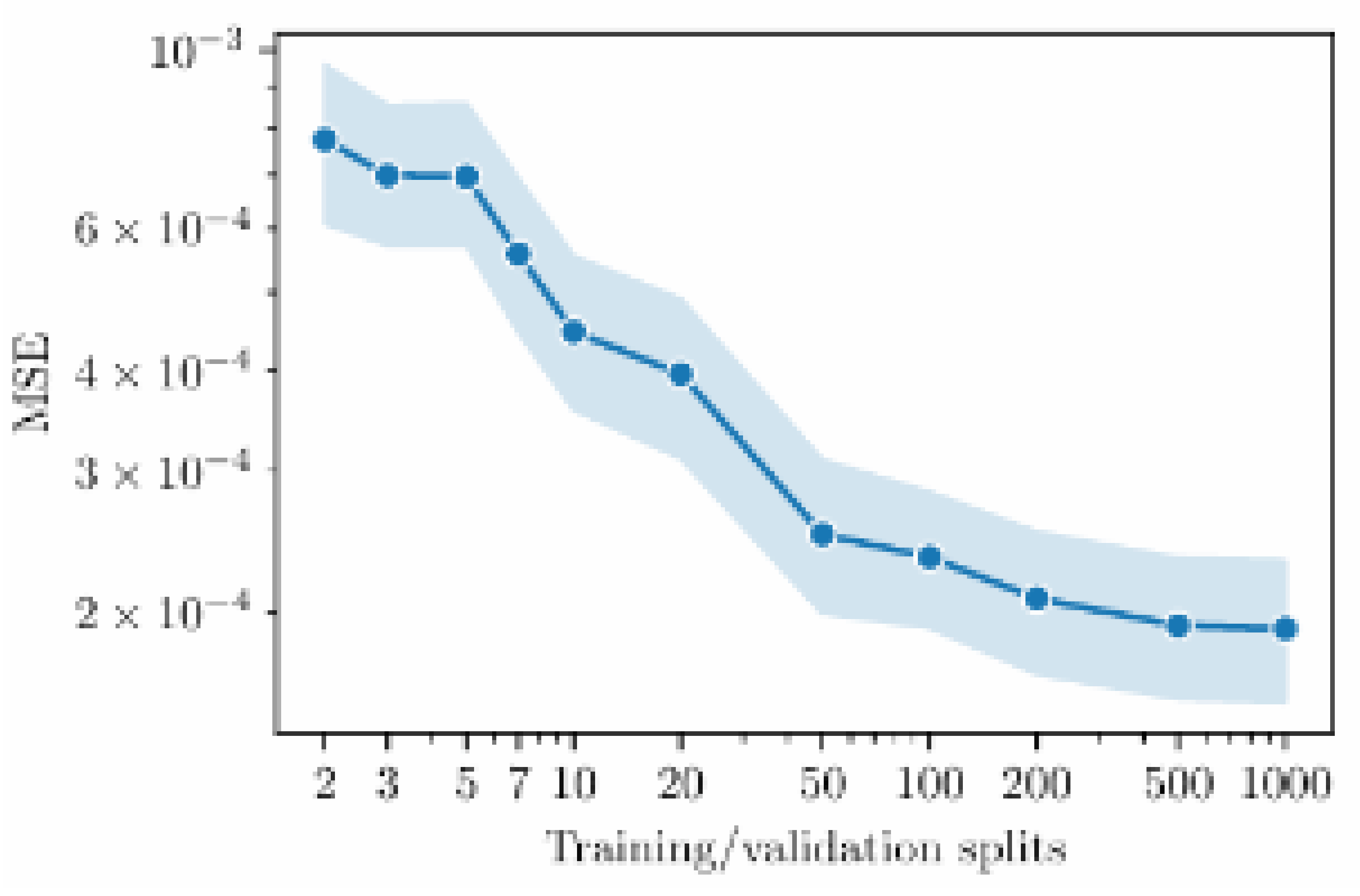
- We couldn't reproduce
- Hyper-parameter tunnning of 7 different estimators
- Authors' proposed tuning is the best
- OCVs follow
- Everything refers to joint estimator selection and hyper-parameter tuning



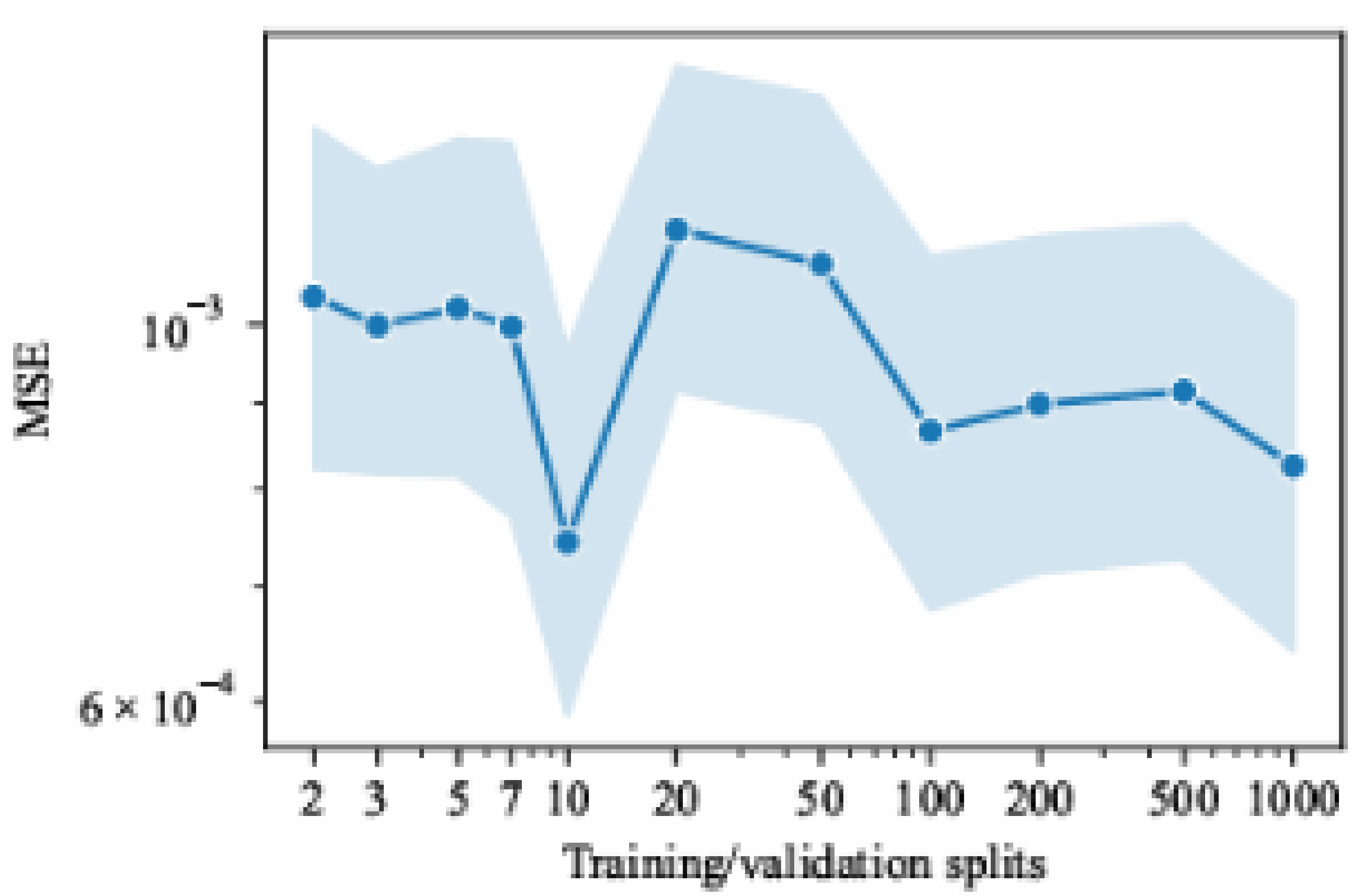
RESULTS

4. **Improvements make standard cross-validation more stable**

- "There is an error limit towards which our method converges with increasing K."
- Potentially holds true



Original results

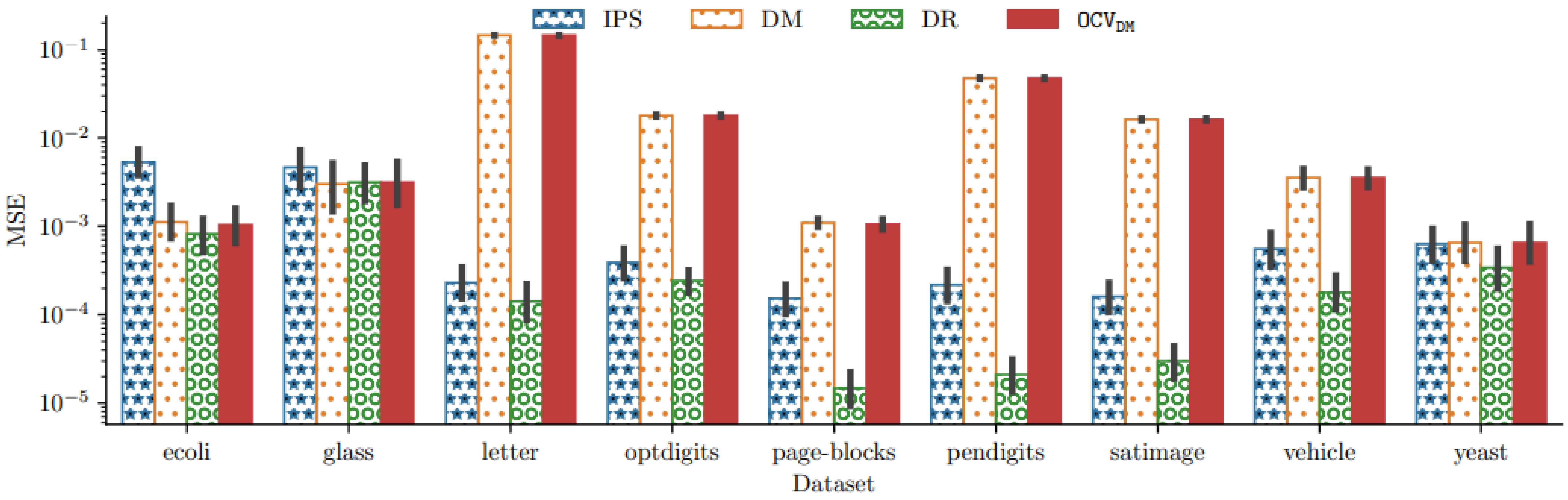


Reproduced results

RESULTS

5. **The validator used in cross validation has to be unbiased**

- When DM, a biased estimator, is used as the validator of OCV, the selector always selects DM, because it is biased towards itself



RESULTS

5. Cross-validation is computationally efficient

- Average computational cost of a single policy evaluation from Figure 1 when doing K=10
- Duration is different due to different hardware
- Order remains the same

Method	OCV_{IPS}	OCV_{DR}	SLOPE	PAS-IF
Time	0.06s	0.13s	0.005s	13.91s

```
Average time for each method
Estimator
\ensuremath{\tt PAS{\text -}IF}    51.22s
\ensuremath{\tt OCV_{DR}}           0.56s
\ensuremath{\tt OCV_{IPS}}          0.27s
\textsc{Slope}                      0.05s
```

DISCUSSION ~ WHAT WAS EASY

Well-Structured Repository:

- The GitHub repository was logically organized and easy to navigate.
- Separate scripts were provided for each experiment.

Clear Documentation:

- Detailed instructions were included on how to run the code.
- Configuration files were provided for each experiment, streamlining the reproduction process.

Minimal Adjustments Required:

- Only minor changes (disabling latex in figures) were necessary to reproduce the results.

Independent Experiment Reproduction:

- Each experiment was self-contained, making it easy to reproduce systematically without interdependencies.

DISCUSSION ~ WHAT WAS DIFFICULT

Understanding Theoretical Components:

- The theorems discussed in the paper required a strong mathematical background to comprehend.

Issues with Code:

- One experiment could not be reproduced from scratch due to a code error.
- A workaround involved using the saved output file provided in the repository, limiting the ability to fully verify the experiment independently.

High Computational Cost:

- One experiments took up to 78 hours due to intensive cross-validation and tuning. On average, an experiment took 15 hours.

QUESTIONS