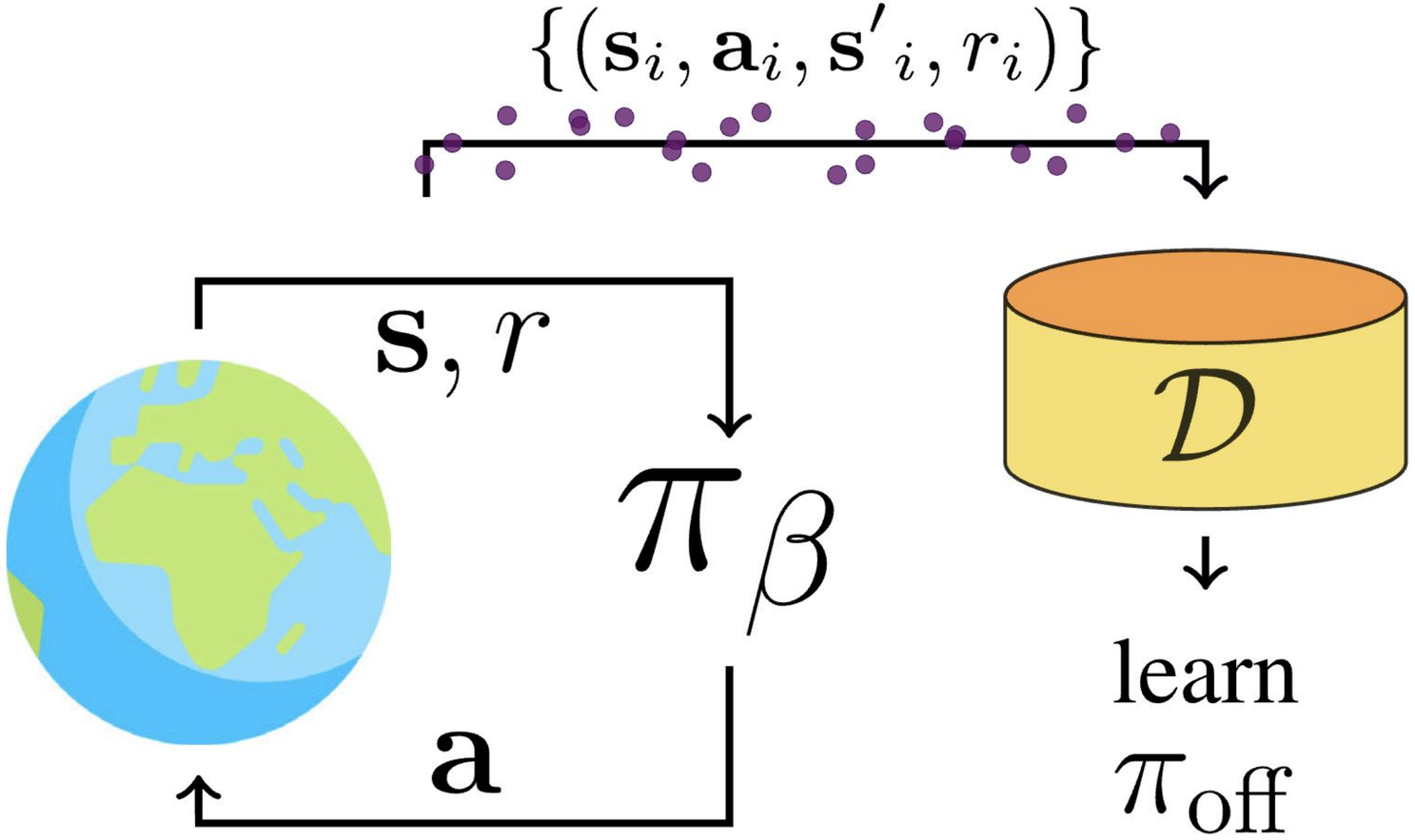
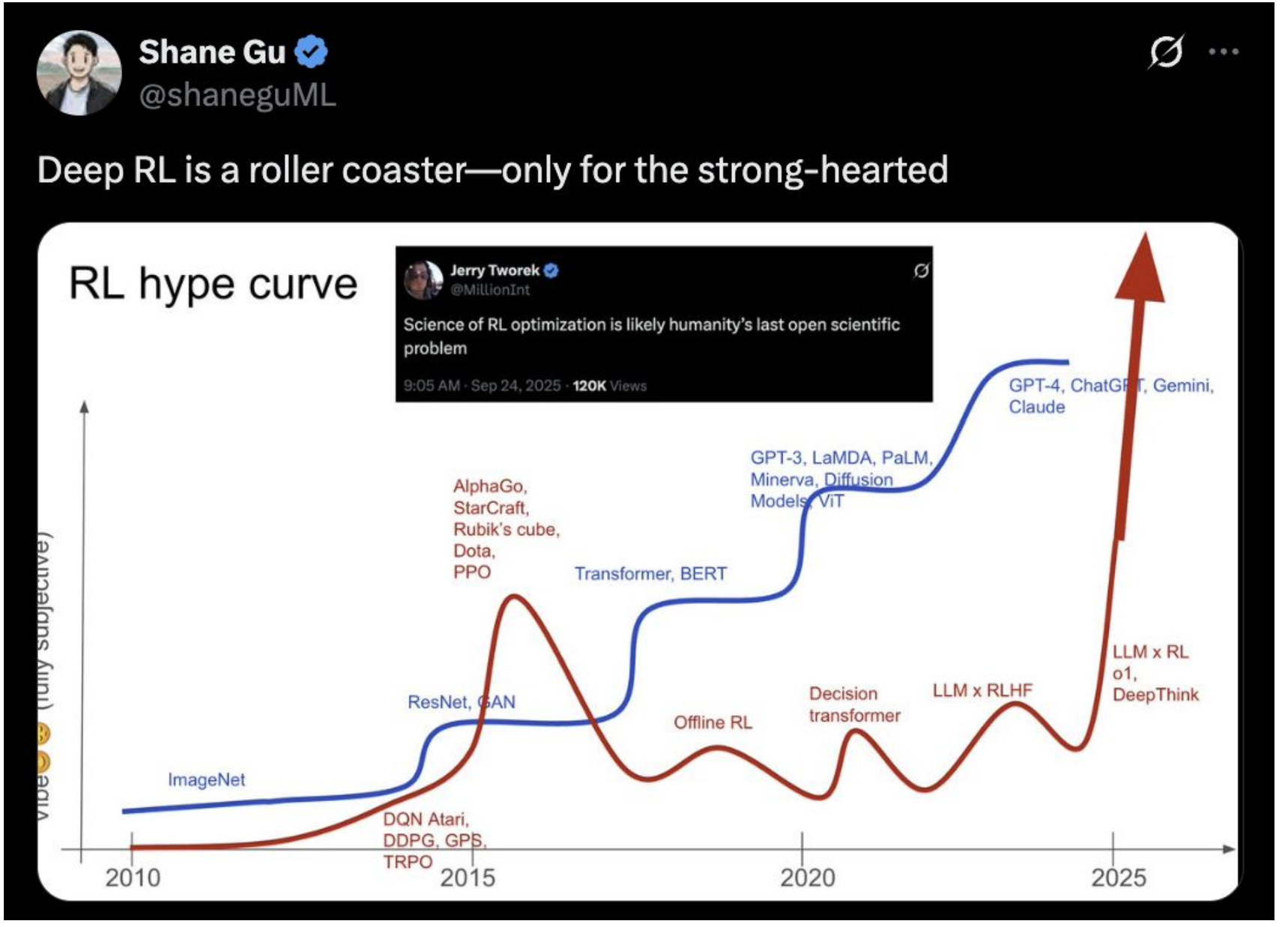
A Clean Slate for Offline RL

Matthew T. Jackson*
Uljad Berdica*
Jarek Liesen*
Shimon Whiteson
Jakob Foerster


$\{(\mathbf{s}_i, \mathbf{a}_i, \mathbf{s'}_i, r_i)\}$
$\mathbf{s}, r$
π_β
$\mathbf{a}$
$\mathcal{D}$
learn
π_{off}


Shane Gu
@shaneguML
Deep RL is a roller coaster—only for the strong-hearted
RL hype curve
Jerry Tworek
@MillionInt
Science of RL optimization is likely humanity's last open scientific problem
9:05 AM · Sep 24, 2025 · 120K Views
vibe (fully subjective)
ImageNet
ResNet, GAN
AlphaGo,
StarCraft,
Rubik's cube,
Dota,
PPO
DQN Atari,
DDPG, GPS,
TRPO
Transformer, BERT
GPT-3, LaMDA, PaLM,
Minerva, Diffusion
Models, ViT
Offline RL
Decision
transformer
GPT-4, ChatGPT, Gemini,
Claude
LLM x RLHF
LLM x RL
o1,
DeepThink
2010
2015
2020
2025

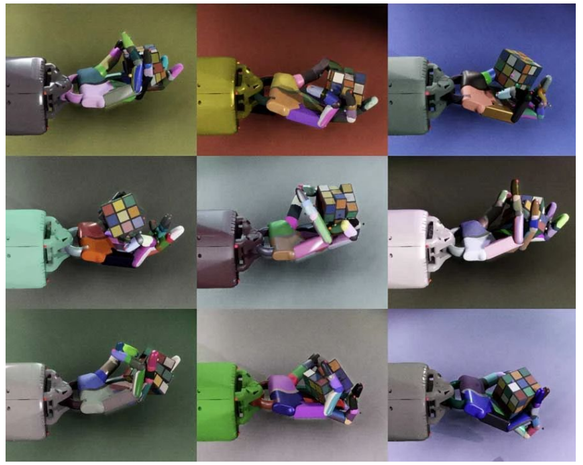
Conditions for Offline RL

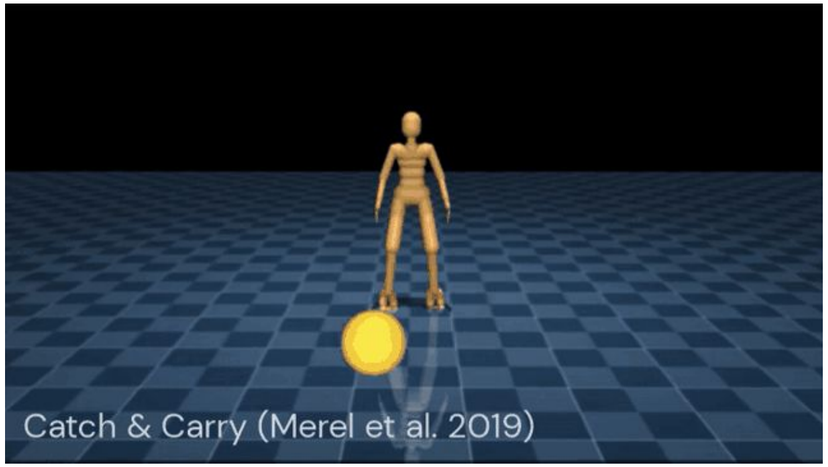
1. Large, **non-expert** dataset

2. **Limited** environment interactions

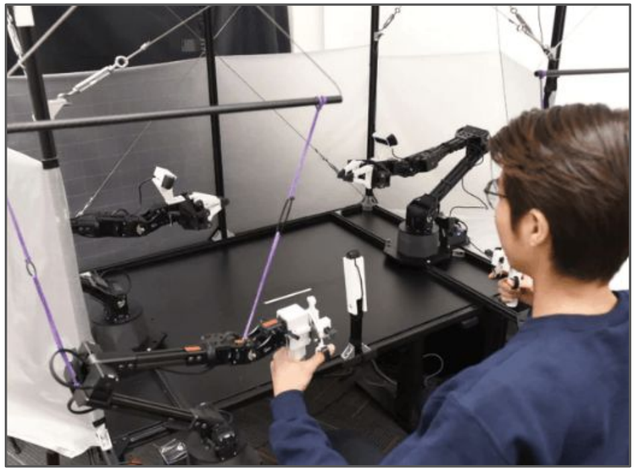
Approaches to Robotics

2010s - Sim-to-real





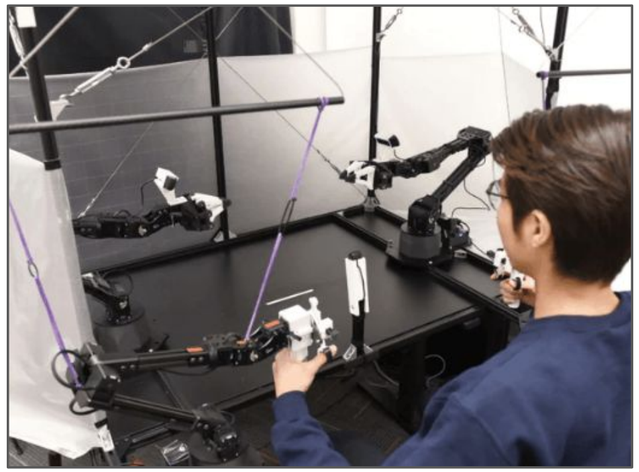
2020s - Teleoperation



Conditions for Offline RL

1. Large, **non-expert** dataset

2. **Limited** environment interactions



Offline RL is Chronically Online

Task Name	BC	BC-10%	TD3+BC	AWAC	CQL	IQL	ReBRAC	SAC-N	EDAC	DT
halfcheetah-medium-v2	42.40 ± 0.19	42.46 ± 0.70	48.10 ± 0.18	50.02 ± 0.27	47.04 ± 0.22	48.31 ± 0.22	64.04 ± 0.68	**68.20 ± 1.28**	67.70 ± 1.04	42.20 ± 0.26
halfcheetah-medium-replay-v2	35.66 ± 2.33	23.59 ± 6.95	44.84 ± 0.59	45.13 ± 0.88	45.04 ± 0.27	44.46 ± 0.22	51.18 ± 0.31	60.70 ± 1.01	**62.06 ± 1.10**	38.91 ± 0.50
halfcheetah-medium-expert-v2	55.95 ± 7.35	90.10 ± 2.45	90.78 ± 6.04	95.00 ± 0.61	95.63 ± 0.42	94.74 ± 0.52	103.80 ± 2.95	98.96 ± 9.31	**104.76 ± 0.64**	91.55 ± 0.95
hopper-medium-v2	53.51 ± 1.76	55.48 ± 7.30	60.37 ± 3.49	63.02 ± 4.56	59.08 ± 3.77	67.53 ± 3.78	**102.29 ± 0.17**	40.82 ± 9.91	101.70 ± 0.28	65.10 ± 1.61
hopper-medium-replay-v2	29.81 ± 2.07	70.42 ± 8.66	64.42 ± 21.52	98.88 ± 2.07	95.11 ± 5.27	97.43 ± 6.39	94.98 ± 6.53	**100.33 ± 0.78**	99.66 ± 0.81	81.77 ± 6.87
hopper-medium-expert-v2	52.30 ± 4.01	111.16 ± 1.03	101.17 ± 9.07	101.90 ± 6.22	99.26 ± 10.91	107.42 ± 7.80	109.45 ± 2.34	101.31 ± 11.63	105.19 ± 10.08	**110.44 ± 0.33**
walker2d-medium-v2	63.23 ± 16.24	67.34 ± 5.17	82.71 ± 4.78	68.52 ± 27.19	80.75 ± 3.28	80.91 ± 3.17	85.82 ± 0.77	87.47 ± 0.66	**93.36 ± 1.38**	67.63 ± 2.54
walker2d-medium-replay-v2	21.80 ± 10.15	54.35 ± 6.34	85.62 ± 4.01	80.62 ± 3.58	73.09 ± 13.22	82.15 ± 3.03	84.25 ± 2.25	78.99 ± 0.50	**87.10 ± 2.78**	59.86 ± 2.73
walker2d-medium-expert-v2	98.96 ± 15.98	108.70 ± 0.25	110.03 ± 0.36	111.44 ± 1.62	109.56 ± 0.39	111.72 ± 0.86	111.86 ± 0.43	**114.93 ± 0.41**	114.75 ± 0.74	107.11 ± 0.96
Gym-MuJoCo avg	50.40	69.29	76.45	79.39	78.28	81.63	89.74	83.52	**92.92**	73.84
maze2d-umaze-v1	0.36 ± 8.69	12.18 ± 4.29	29.41 ± 12.31	65.65 ± 5.34	-8.90 ± 6.11	42.11 ± 0.58	106.87 ± 22.16	**130.59 ± 16.52**	95.26 ± 6.39	18.08 ± 25.42
maze2d-medium-v1	0.79 ± 3.25	14.25 ± 2.33	59.45 ± 36.25	84.63 ± 35.54	86.11 ± 9.68	34.85 ± 2.72	**105.11 ± 31.67**	88.61 ± 18.72	57.04 ± 3.45	31.71 ± 26.33
maze2d-large-v1	2.26 ± 4.39	11.32 ± 5.10	97.10 ± 25.41	**215.50 ± 3.11**	23.75 ± 36.70	61.72 ± 3.50	78.33 ± 61.77	204.76 ± 1.19	95.60 ± 22.92	35.66 ± 28.20
Maze2d avg	1.13	12.58	61.99	121.92	33.65	46.23	96.77	**141.32**	82.64	28.48
antmaze-umaze-v2	55.25 ± 4.15	65.75 ± 5.26	70.75 ± 39.18	56.75 ± 9.09	92.75 ± 1.92	77.00 ± 5.52	**97.75 ± 1.48**	0.00 ± 0.00	0.00 ± 0.00	57.00 ± 9.82
antmaze-umaze-diverse-v2	47.25 ± 4.09	44.00 ± 1.00	44.75 ± 11.61	54.75 ± 8.01	37.25 ± 3.70	54.25 ± 5.54	**83.50 ± 7.02**	0.00 ± 0.00	0.00 ± 0.00	51.75 ± 0.43
antmaze-medium-play-v2	0.00 ± 0.00	2.00 ± 0.71	0.25 ± 0.43	0.00 ± 0.00	65.75 ± 11.61	65.75 ± 11.71	**89.50 ± 3.35**	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
antmaze-medium-diverse-v2	0.75 ± 0.83	5.75 ± 9.39	0.25 ± 0.43	0.00 ± 0.00	67.25 ± 3.56	73.75 ± 5.45	**83.50 ± 8.20**	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
antmaze-large-play-v2	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	20.75 ± 7.26	42.00 ± 4.53	**52.25 ± 29.01**	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
antmaze-large-diverse-v2	0.00 ± 0.00	0.75 ± 0.83	0.00 ± 0.00	0.00 ± 0.00	20.50 ± 13.24	30.25 ± 3.63	**64.00 ± 5.43**	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
AntMaze avg	17.21	19.71	19.33	18.58	50.71	57.17	**78.42**	0.00	0.00	18.12
pen-human-v1	71.03 ± 6.26	26.99 ± 9.60	-3.88 ± 0.21	76.65 ± 11.71	13.71 ± 16.98	78.49 ± 8.21	**103.16 ± 8.49**	6.86 ± 5.93	5.07 ± 6.16	67.68 ± 5.48
pen-cloned-v1	51.92 ± 15.15	46.67 ± 14.25	5.13 ± 5.28	85.72 ± 16.92	1.04 ± 6.62	83.42 ± 8.19	**102.79 ± 7.84**	31.35 ± 2.14	12.02 ± 1.75	64.43 ± 1.43
pen-expert-v1	109.65 ± 7.28	114.96 ± 2.96	122.53 ± 21.27	**159.91 ± 1.87**	-1.41 ± 2.34	128.05 ± 9.21	152.16 ± 6.33	87.11 ± 48.95	-1.55 ± 0.81	116.38 ± 1.27
door-human-v1	2.34 ± 4.00	-0.13 ± 0.07	-0.33 ± 0.01	2.39 ± 2.26	**5.53 ± 1.31**	3.26 ± 1.83	-0.10 ± 0.01	-0.38 ± 0.00	-0.12 ± 0.13	4.44 ± 0.87
door-cloned-v1	-0.09 ± 0.03	0.29 ± 0.59	-0.34 ± 0.01	-0.01 ± 0.01	-0.33 ± 0.01	3.07 ± 1.75	0.06 ± 0.05	-0.33 ± 0.00	2.66 ± 2.31	**7.64 ± 3.26**
door-expert-v1	105.35 ± 0.09	104.04 ± 1.46	-0.33 ± 0.01	104.57 ± 0.31	-0.32 ± 0.02	**106.65 ± 0.25**	106.37 ± 0.29	-0.33 ± 0.00	106.29 ± 1.73	104.87 ± 0.39
hammer-human-v1	3.03 ± 3.39	-0.19 ± 0.02	1.02 ± 0.24	1.01 ± 0.51	0.14 ± 0.11	**1.79 ± 0.80**	0.24 ± 0.24	0.24 ± 0.00	0.28 ± 0.18	1.28 ± 0.15
hammer-cloned-v1	0.55 ± 0.16	0.12 ± 0.08	0.25 ± 0.01	1.27 ± 2.11	0.30 ± 0.01	1.50 ± 0.69	**5.00 ± 3.75**	0.14 ± 0.09	0.19 ± 0.07	1.82 ± 0.55
hammer-expert-v1	126.78 ± 0.64	121.75 ± 7.67	3.11 ± 0.03	127.08 ± 0.13	0.26 ± 0.01	128.68 ± 0.33	**133.62 ± 0.27**	25.13 ± 43.25	28.52 ± 49.00	117.45 ± 6.65
relocate-human-v1	0.04 ± 0.03	-0.14 ± 0.08	-0.29 ± 0.01	0.45 ± 0.53	0.06 ± 0.03	0.12 ± 0.04	**0.16 ± 0.30**	-0.31 ± 0.01	-0.17 ± 0.17	0.05 ± 0.01
relocate-cloned-v1	-0.06 ± 0.01	-0.00 ± 0.02	-0.30 ± 0.01	-0.01 ± 0.03	-0.29 ± 0.01	0.04 ± 0.01	**1.66 ± 2.59**	-0.01 ± 0.10	0.17 ± 0.35	0.16 ± 0.09
relocate-expert-v1	107.58 ± 1.20	97.90 ± 5.21	-1.73 ± 0.96	**109.52 ± 0.47**	-0.30 ± 0.02	106.11 ± 4.02	107.52 ± 2.28	-0.36 ± 0.00	71.94 ± 18.37	104.28 ± 0.42
Adroit avg	48.18	42.69	10.40	55.71	1.53	53.43	**59.39**	12.43	18.78	49.21
Total avg	37.95	43.06	37.16	62.01	37.61	61.92	**76.04**	44.16	43.65	48.31

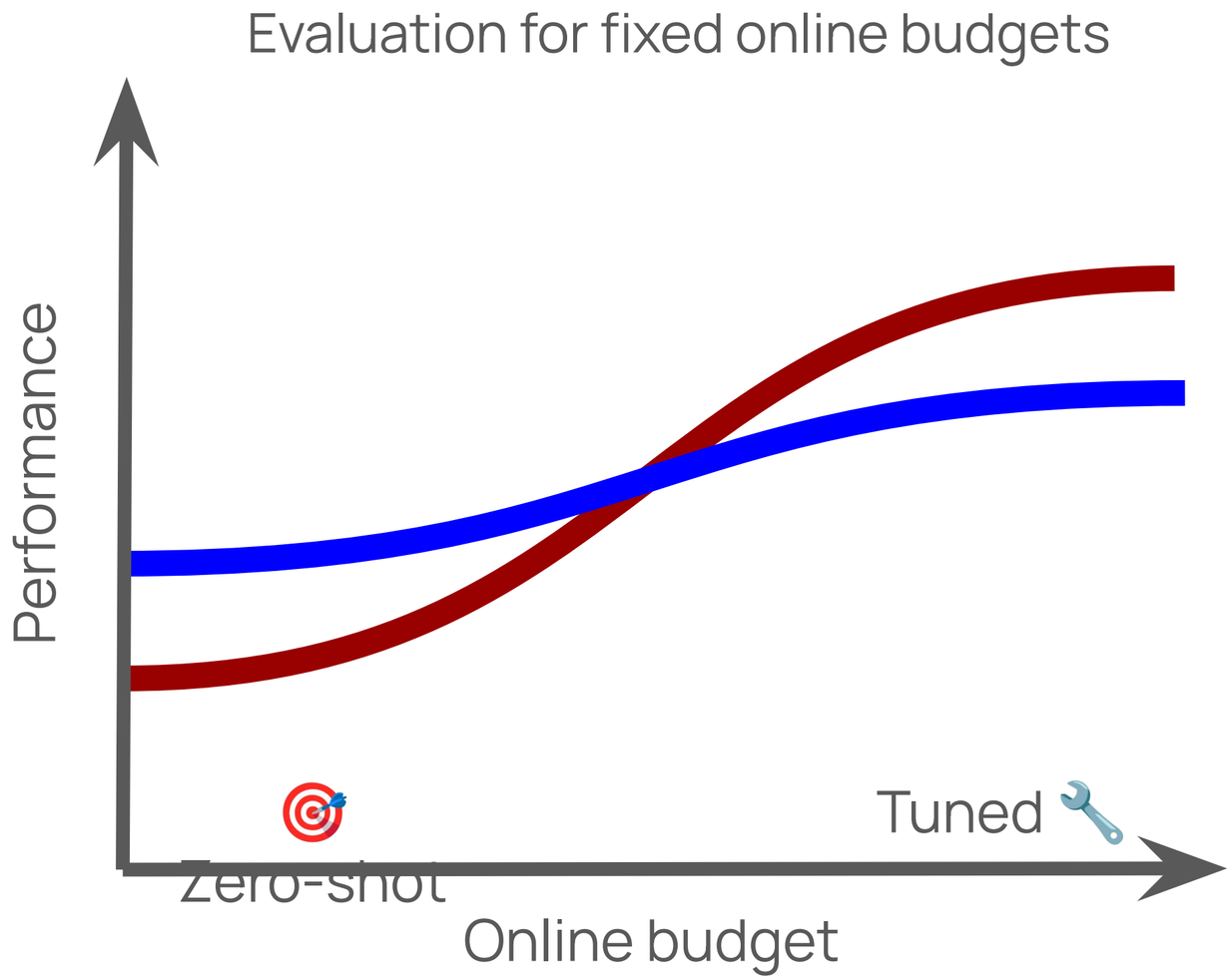
Dataset Performance

Table 8: Environment specific hyperparameters.

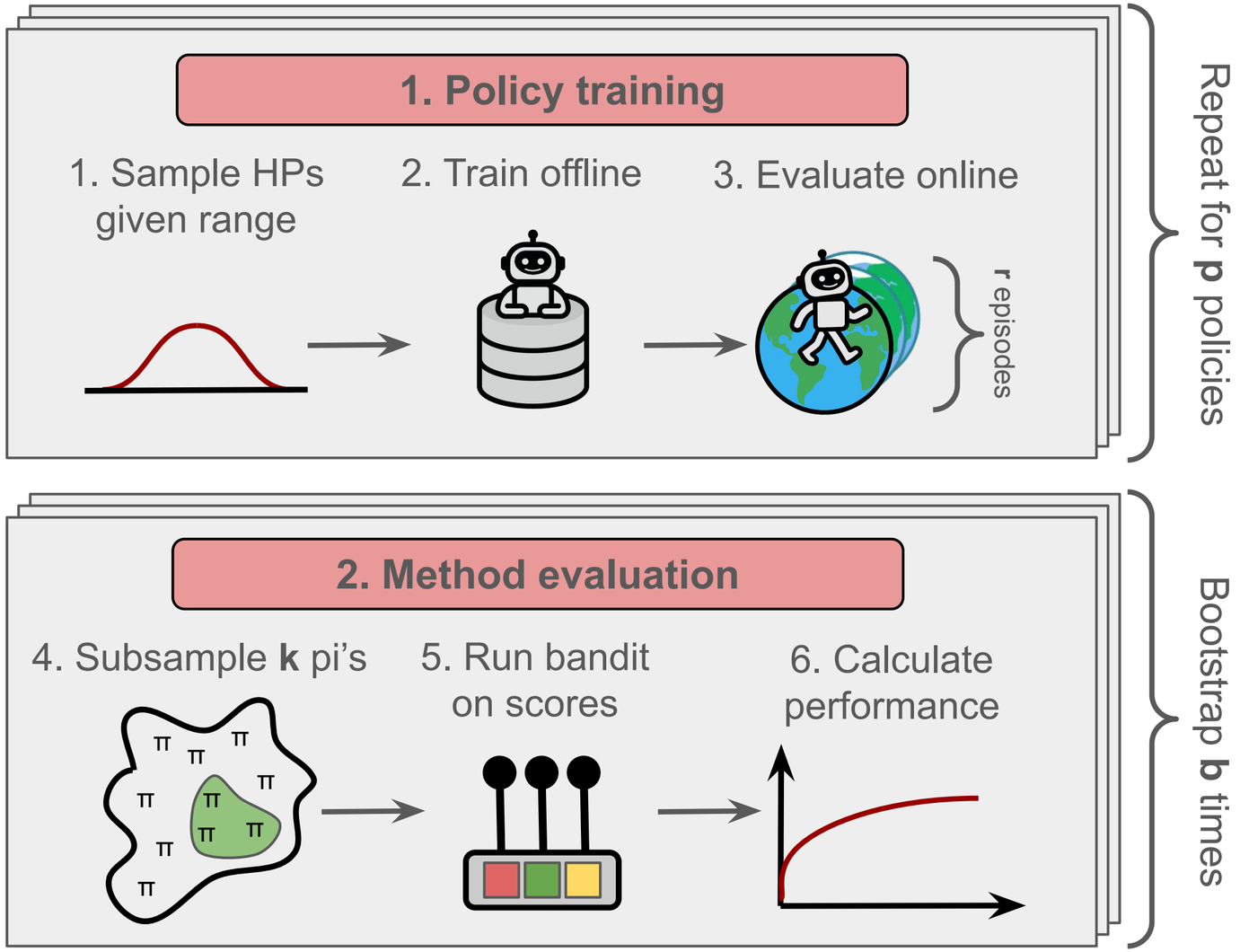
Task Name	**SAC-N** (N)	**LB-SAC** (N, LayerNorm)	**EDAC** (N, η)
halfcheetah-random	10	2, False	10, 0.0
halfcheetah-medium	10	4, False	10, 1.0
halfcheetah-expert	10	6, False	10, 1.0
halfcheetah-medium-expert	10	8, False	10, 5.0
halfcheetah-medium-replay	10	4, False	10, 1.0
halfcheetah-full-replay	10	4, False	10, 1.0
hopper-random	500	25, False	50, 1.0
hopper-medium	500	25, True	50, 1.0
hopper-expert	500	50, False	50, 1.0
hopper-medium-expert	200	40, False	50, 1.0
hopper-medium-replay	200	20, False	50, 1.0
hopper-full-replay	200	25, False	50, 1.0
walker2d-random	20	15, False	10, 1.0
walker2d-medium	20	10, False	10, 1.0
walker2d-expert	100	30, False	10, 5.0
walker2d-medium-expert	20	10, False	10, 5.0
walker2d-medium-replay	20	10, False	10, 1.0
walker2d-full-replay	20	4, False	10, 1.0
ant-medium	50	10, True	10, 5.0

Dataset-specific Hyperparameters

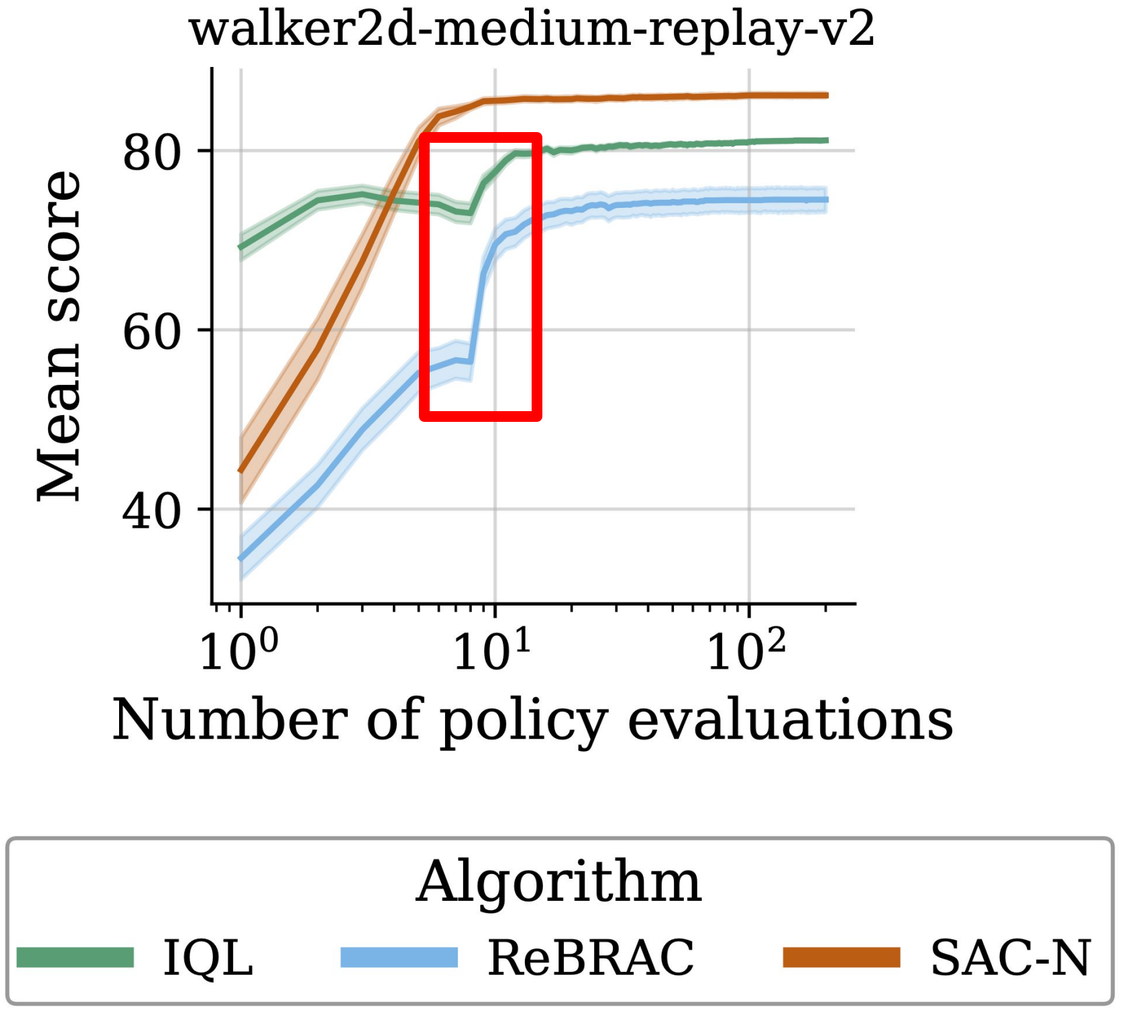
Elucidating Offline RL



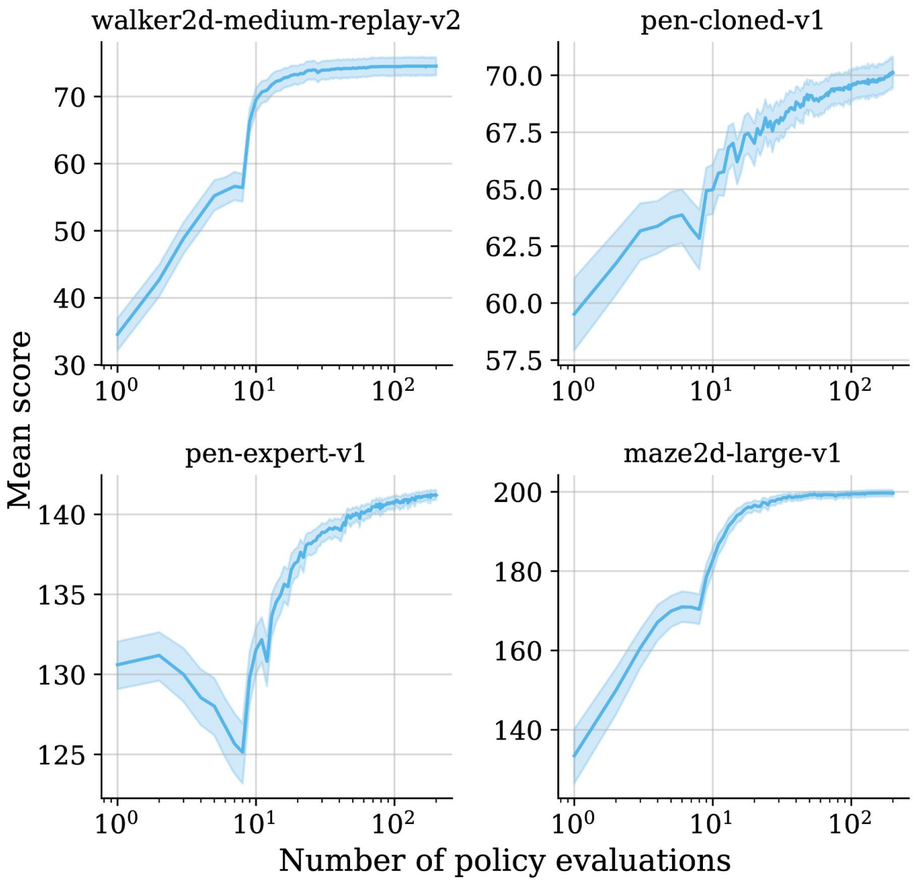
Proposed Evaluation Procedure

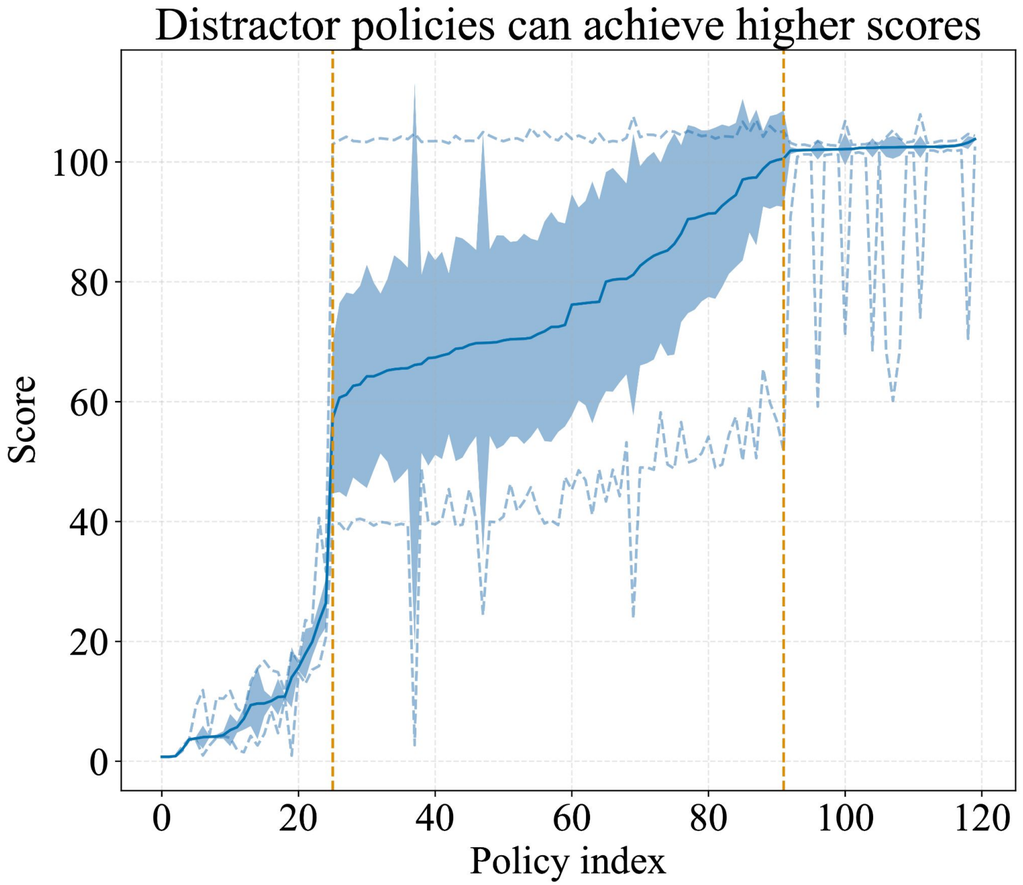


Example Evaluation

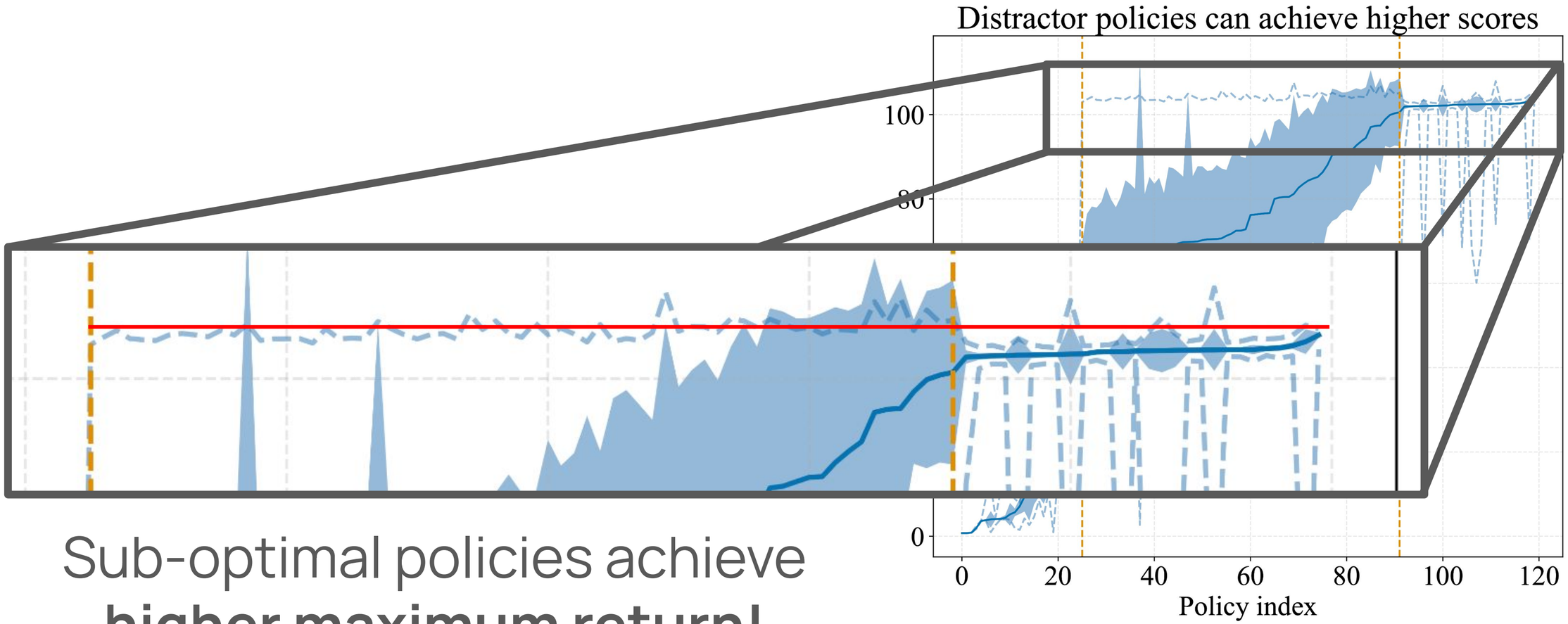


The Distractor Policy Phenomenon





The Distractor Policy Phenomenon

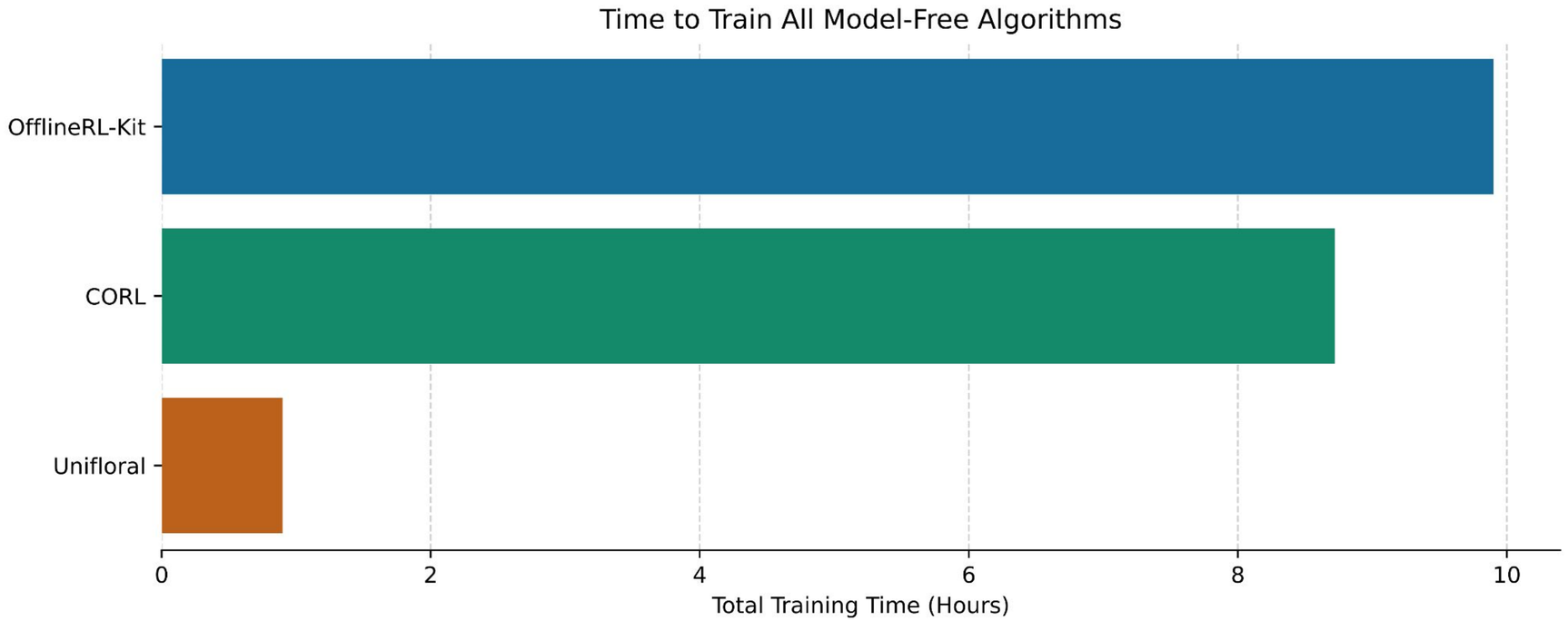


Sub-optimal policies achieve
higher maximum return!

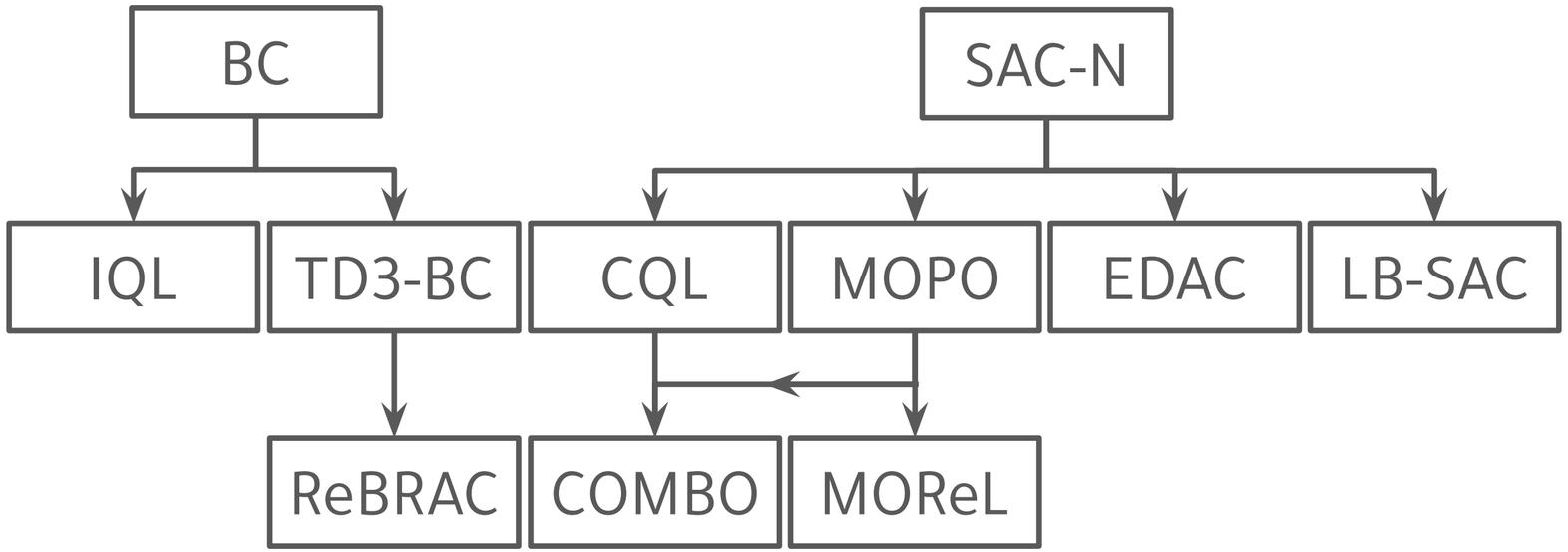
The Unifloral Library

Model-free: BC, TD3-BC, ReBRAC, IQL, SAC-N, LB-SAC, CQL, EDAC
Model-based: MOPO, MoReL, COMBO

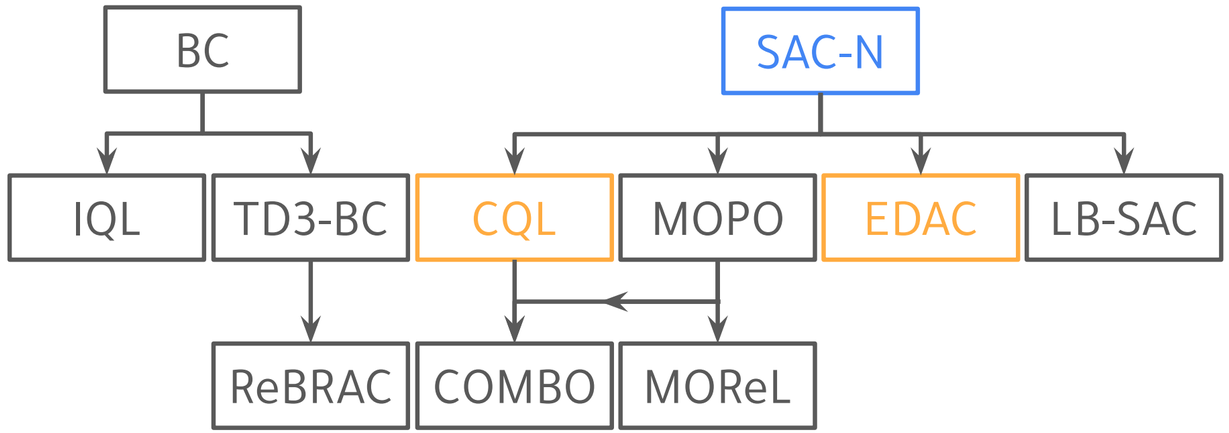
All in JAX!



The Unifloral Library



The Unifloral Library



SAC-N

sac_n.py

```python
# --- Update critics ---
@jax.value_and_grad
def _q_loss_fn(params):
    q_pred = q_apply_fn(params, batch.obs, batch.action)
    return jnp.square((q_pred - jnp.expand_dims(target, -1))).sum(-1).mean()

critic_loss, critic_grad = _q_loss_fn(agent_state.vec_q.params)
updated_q = agent_state.vec_q.apply_gradients(grads=critic_grad)
agent_state = agent_state._replace(vec_q=updated_q)
```

EDAC

edac.py

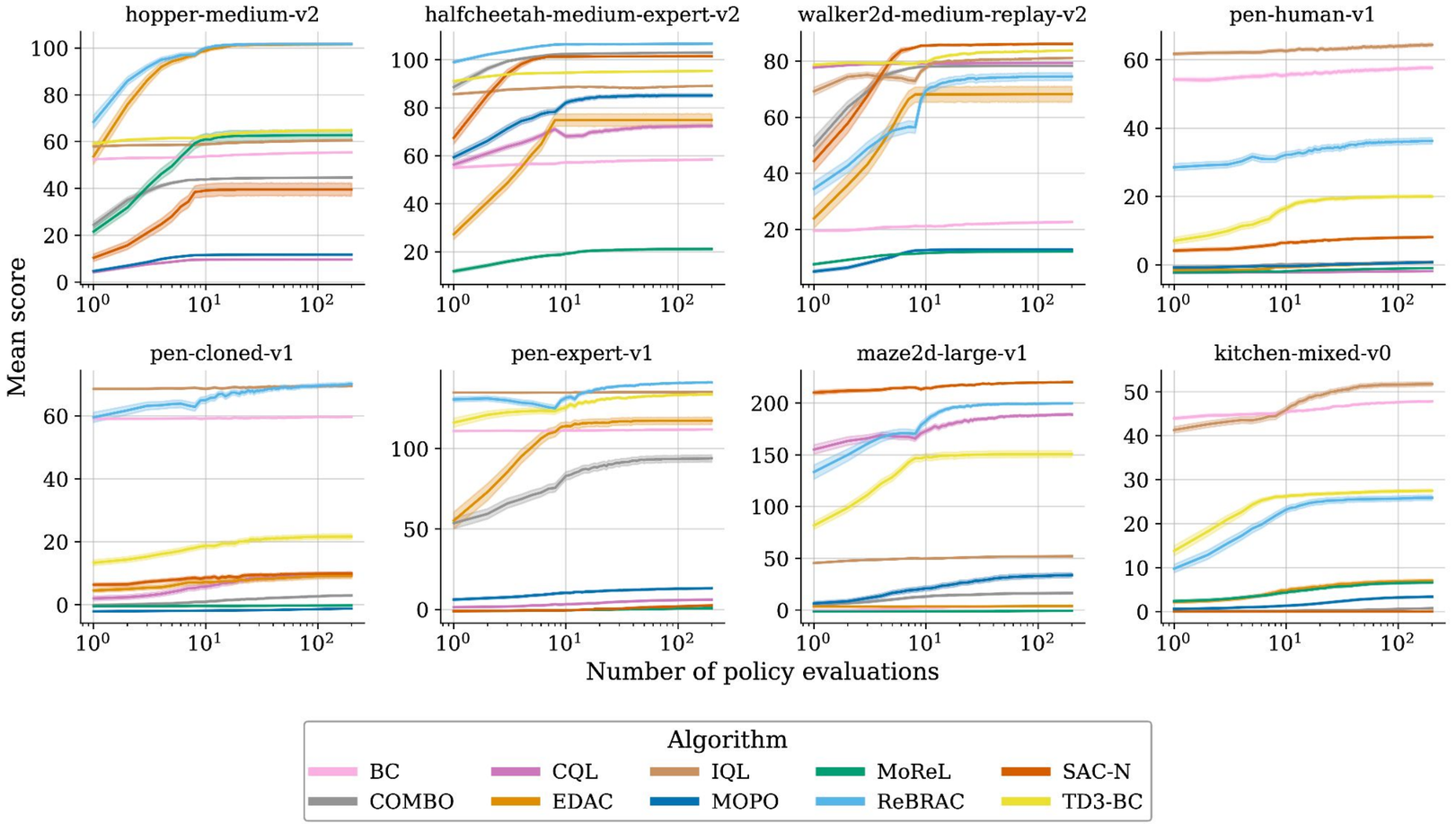
```python
# --- Update critics ---
@partial(jax.value_and_grad, has_aux=True)
def _q_loss_fn(params):
    value_loss = jnp.square((q_pred - jnp.expand_dims(target, -1)))
    value_loss = value_loss.sum(-1).mean()
    diversity_loss = jax.vmap(_diversity_loss_fn)(batch.obs, batch.action)
    diversity_loss = (1 / (args.num_critics - 1)) * diversity_loss.mean()
    critic_loss = value_loss + args.eta * diversity_loss
    return critic_loss, (value_loss, diversity_loss)

(critic_loss, (value_loss, diversity_loss)), critic_grad = _q_loss_fn(
    agent_state.vec_q.params)
updated_q = agent_state.vec_q.apply_gradients(grads=critic_grad)
agent_state = agent_state._replace(vec_q=updated_q)
```

CQL

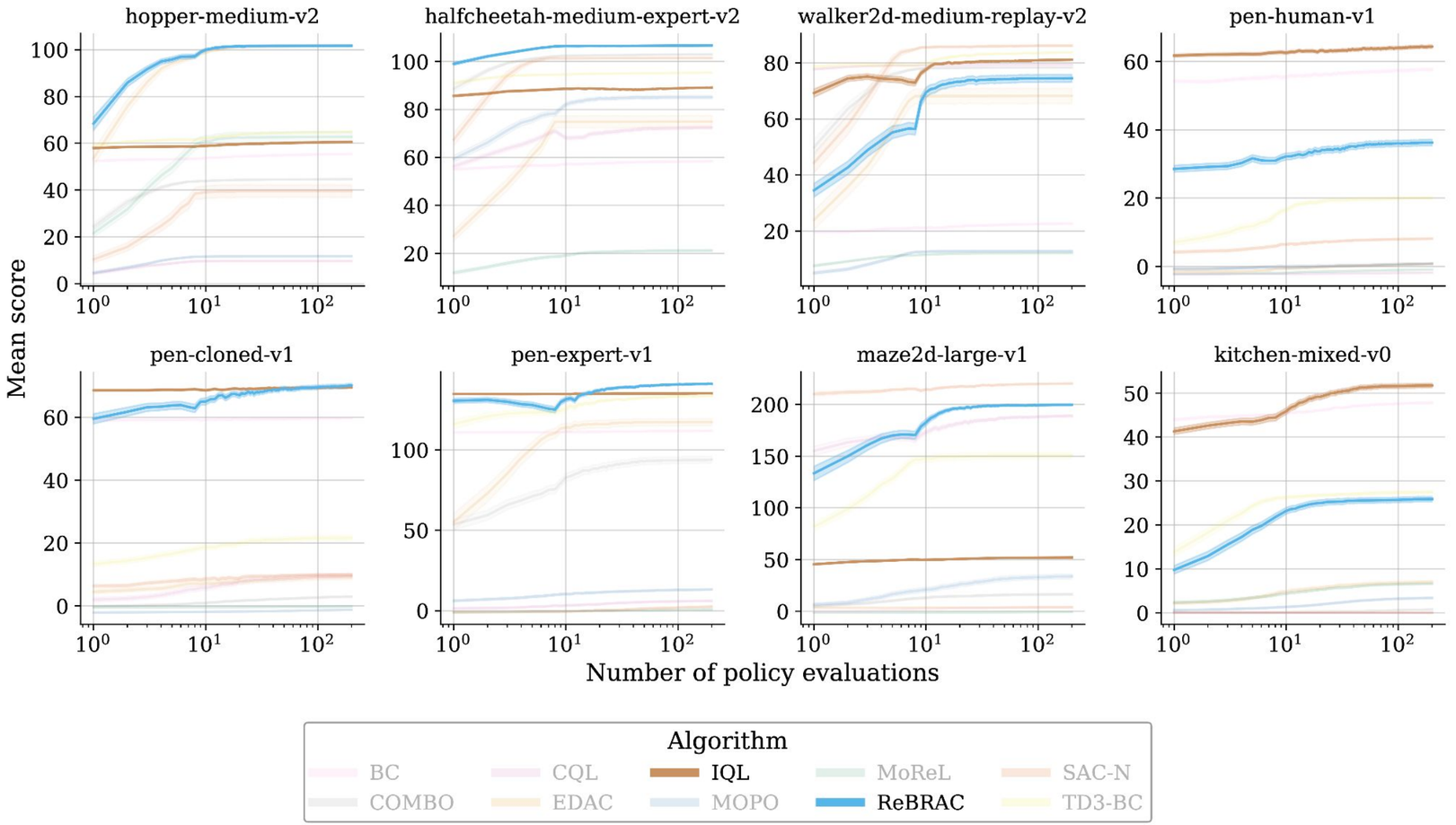
cql.py

```python
# --- Update critics ---
@jax.value_and_grad
def _q_loss_fn(params):
    q_pred = q_apply_fn(params, batch.obs, batch.action)
    critic_loss = jnp.square((q_pred - jnp.expand_dims(target, -1)))
    critic_loss = critic_loss.sum(-1).mean()
    rand_q = q_apply_fn(params, batch.obs, cql_random_actions)
    pi_q = q_apply_fn(params, batch.obs, pi_actions)
    next_pi_q = q_apply_fn(params, batch.next_obs, pi_next_actions)
    all_qs = jnp.concatenate([rand_q, pi_q, next_pi_q, q_pred], axis=1)
    q_ood = jax.scipy.special.logsumexp(all_qs / args.cql_temperature, axis=1)
    q_ood = jax.lax.stop_gradient(q_ood * args.cql_temperature)
    q_diff = (jnp.expand_dims(q_ood, 1) - q_pred).mean()
    min_q_loss = q_diff * args.cql_min_q_weight

    critic_loss += min_q_loss.mean()
    return critic_loss

critic_loss, critic_grad = _q_loss_fn(agent_state.vec_q.params)
updated_q = agent_state.vec_q.apply_gradients(grads=critic_grad)
agent_state = agent_state._replace(vec_q=updated_q)
```

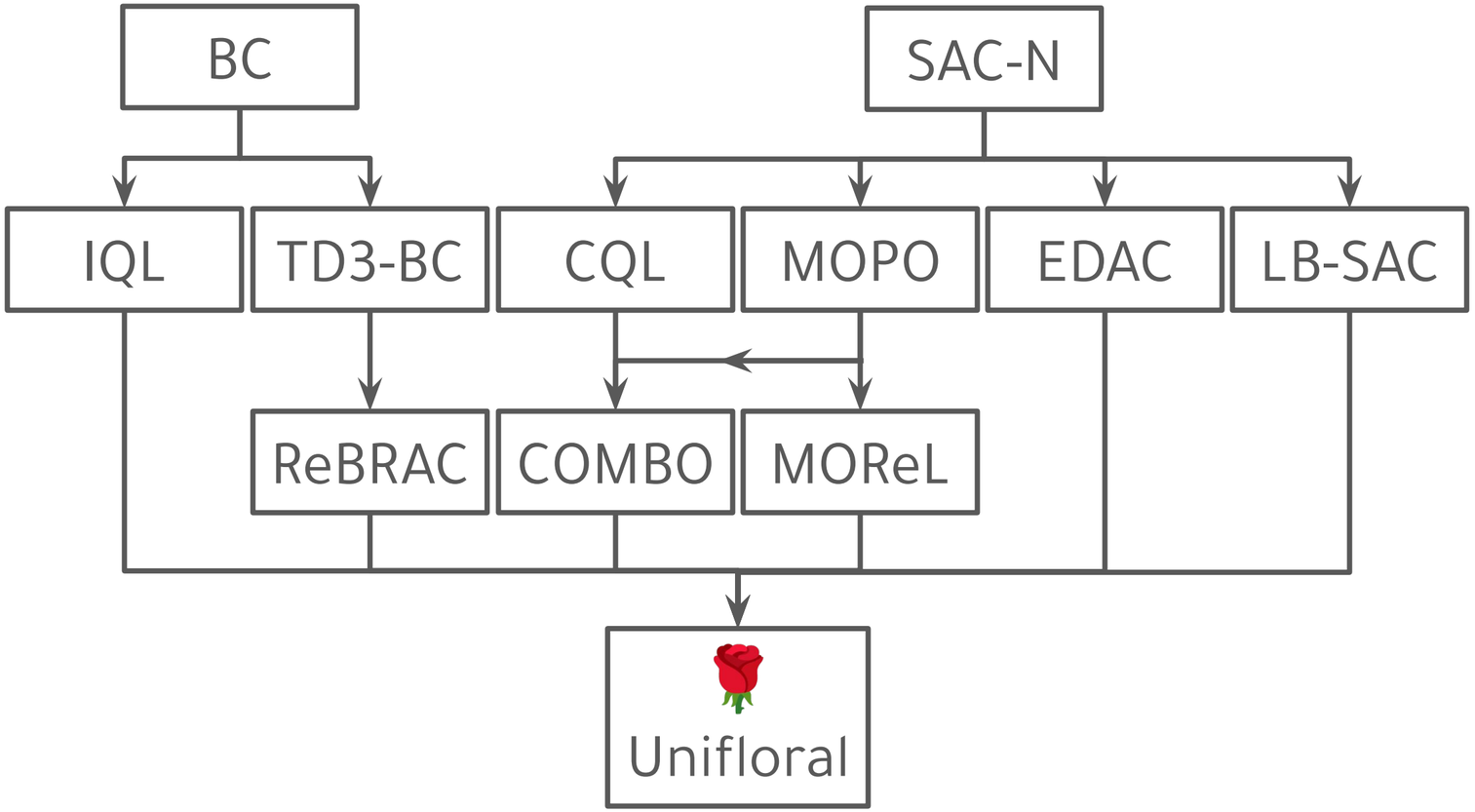
Revisiting the Field



Revisiting the Field



Simplifying the Design Space



Simplifying the Design Space

BC

bc.yaml

```yaml
lr: 3e-4
actor_lr: 3e-4
lr_schedule: constant
batch_size: 256
gamma: 0.99
polyak_step_size: 0.005
norm_obs: true
actor_num_layers: 2
actor_layer_width: 256
actor_ln: false
deterministic: true
deterministic_eval: false
use_tanh_mean: true
use_log_std_param: false
log_std_min: -5.0
log_std_max: 2.0
num_critics: 2
critic_num_layers: 2
critic_layer_width: 256
critic_ln: false
actor_bc_coef: 1.0
actor_q_coef: 0.0
use_q_target_in_actor: false
normalize_q_loss: false
aggregate_q: first
use_awr: false
awr_temperature: 1.0
awr_exp_adv_clip: 100.0
num_critic_updates_per_step: 1
critic_bc_coef: 0.0
diversity_coef: 0.0
policy_noise: 0.0
noise_clip: 0.0
use_target_actor: false
use_value_target: false
value_expectile: 0.8
use_entropy_loss: false
ent_coef_init: 1.0
actor_entropy_coef: 0.0
critic_entropy_coef: 0.0
```

TD3-BC

td3_bc.yaml

```yaml
lr: 3e-4
actor_lr: 3e-4
lr_schedule: constant
batch_size: 256
gamma: 0.99
polyak_step_size: 0.005
norm_obs: true
actor_num_layers: 2
actor_layer_width: 256
actor_ln: false
deterministic: true
deterministic_eval: false
use_tanh_mean: true
use_log_std_param: false
log_std_min: -5.0
log_std_max: 2.0
num_critics: 2
critic_num_layers: 2
critic_layer_width: 256
critic_ln: false
actor_bc_coef: 1.0
actor_q_coef: [1.0, 2.0, 2.5, 3.0, 4.0]
use_q_target_in_actor: false
normalize_q_loss: true
aggregate_q: first
use_awr: false
awr_temperature: 1.0
awr_exp_adv_clip: 100.0
num_critic_updates_per_step: 2
critic_bc_coef: 0.0
diversity_coef: 0.0
policy_noise: 0.2
noise_clip: 0.5
use_target_actor: true
use_value_target: false
value_expectile: 0.8
use_entropy_loss: false
ent_coef_init: 1.0
actor_entropy_coef: 0.0
critic_entropy_coef: 0.0
```

Simplifying the Design Space

BC

bc.yaml

```yaml
lr: 3e-4
actor_lr: 3e-4
lr_schedule: constant
batch_size: 256
gamma: 0.99
polyak_step_size: 0.005
norm_obs: true
actor_num_layers: 2
actor_layer_width: 256
actor_ln: false
deterministic: true
deterministic_eval: false
use_tanh_mean: true
use_log_std_param: false
log_std_min: -5.0
log_std_max: 2.0
num_critics: 2
critic_num_layers: 2
critic_layer_width: 256
critic_ln: false
actor_bc_coef: 1.0
actor_q_coef: 0.0
use_q_target_in_actor: false
normalize_q_loss: false
aggregate_q: first
use_awr: false
awr_temperature: 1.0
awr_exp_adv_clip: 100.0
num_critic_updates_per_step: 1
critic_bc_coef: 0.0
diversity_coef: 0.0
policy_noise: 0.0
noise_clip: 0.0
use_target_actor: false
use_value_target: false
value_expectile: 0.8
use_entropy_loss: false
ent_coef_init: 1.0
actor_entropy_coef: 0.0
critic_entropy_coef: 0.0
```

TD3-BC

td3_bc.yaml

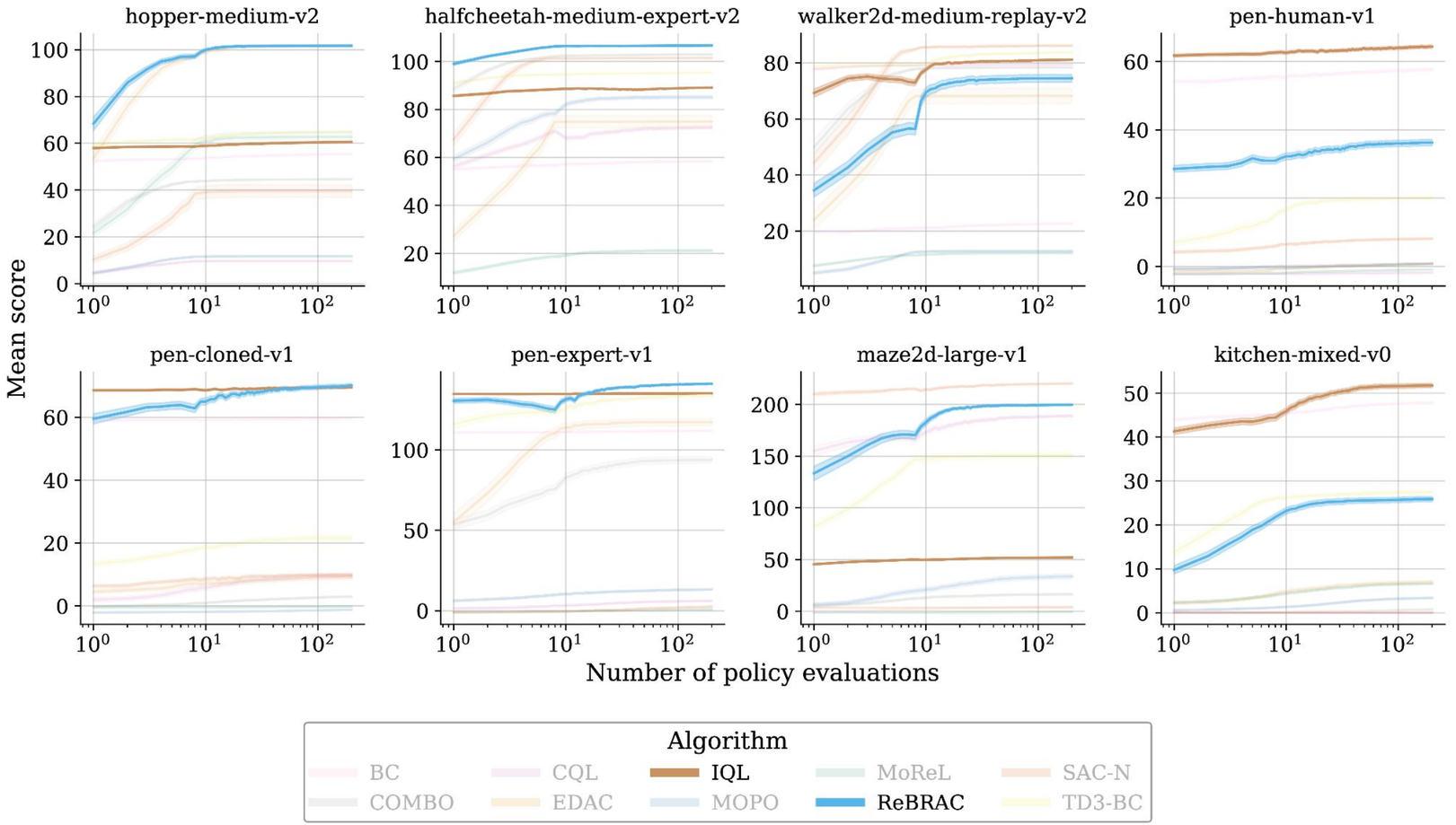
```yaml
lr: 3e-4
actor_lr: 3e-4
lr_schedule: constant
batch_size: 256
gamma: 0.99
polyak_step_size: 0.005
norm_obs: true
actor_num_layers: 2
actor_layer_width: 256
actor_ln: false
deterministic: true
deterministic_eval: false
use_tanh_mean: true
use_log_std_param: false
log_std_min: -5.0
log_std_max: 2.0
num_critics: 2
critic_num_layers: 2
critic_layer_width: 256
critic_ln: false
actor_bc_coef: 1.0
actor_q_coef: [1.0, 2.0, 2.5, 3.0, 4.0]
use_q_target_in_actor: false
normalize_q_loss: true
aggregate_q: first
use_awr: false
awr_temperature: 1.0
awr_exp_adv_clip: 100.0
num_critic_updates_per_step: 2
critic_bc_coef: 0.0
diversity_coef: 0.0
policy_noise: 0.2
noise_clip: 0.5
use_target_actor: true
use_value_target: false
value_expectile: 0.8
use_entropy_loss: false
ent_coef_init: 1.0
actor_entropy_coef: 0.0
critic_entropy_coef: 0.0
```

ReBRAC

rebrac.yaml

```yaml
lr: 1e-3
actor_lr: 1e-3
lr_schedule: constant
batch_size: 1024
gamma: 0.99
polyak_step_size: 0.005
norm_obs: false
actor_num_layers: 3
actor_layer_width: 256
actor_ln: true
deterministic: true
deterministic_eval: false
use_tanh_mean: true
use_log_std_param: false
log_std_min: -5.0
log_std_max: 2.0
num_critics: 2
critic_num_layers: 3
critic_layer_width: 256
critic_ln: true
actor_bc_coef: [0.0005, 0.001, 0.002, 0.003, 0.03, 0.1, 0.3, 1.0]
actor_q_coef: 1.0
use_q_target_in_actor: false
normalize_q_loss: true
aggregate_q: min
use_awr: false
awr_temperature: 1.0
awr_exp_adv_clip: 100.0
num_critic_updates_per_step: 2
critic_bc_coef: [0, 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.1]
diversity_coef: 0.0
policy_noise: 0.2
noise_clip: 0.5
use_target_actor: true
use_value_target: false
value_expectile: 0.8
use_entropy_loss: false
ent_coef_init: 1.0
actor_entropy_coef: 0.0
critic_entropy_coef: 0.0
```

Novel Algorithms in Unifloral - TD3-AWR



Novel Algorithms in Unifloral - TD3-AWR

IQL (AWR)

iql.yaml

```yaml
lr: 3e-4
actor_lr: 3e-4
lr_schedule: cosine
batch_size: 256
gamma: 0.99
polyak_step_size: 0.005
norm_obs: true
actor_num_layers: 2
actor_layer_width: 256
actor_ln: false
deterministic: false
deterministic_eval: true
use_tanh_mean: true
use_log_std_param: true
log_std_min: -20.0
log_std_max: 2.0
num_critics: 2
critic_num_layers: 2
critic_layer_width: 256
critic_ln: false
actor_bc_coef: 1.0
actor_q_coef: 0.0
use_q_target_in_actor: false
normalize_q_loss: false
aggregate_q: min
use_awr: true
awr_temperature: [0.5, 3.0, 10.0]
awr_exp_adv_clip: 100.0
num_critic_updates_per_step: 1
critic_bc_coef: 0.0
diversity_coef: 0.0
policy_noise: 0.0
noise_clip: 0.0
use_target_actor: false
use_value_target: false
value_expectile: [0.5, 0.7, 0.9]
use_entropy_loss: false
ent_coef_init: 1.0
actor_entropy_coef: 0.0
critic_entropy_coef: 0.0
```

+

ReBRAC

rebrac.yaml

```yaml
lr: 1e-3
actor_lr: 1e-3
lr_schedule: constant
batch_size: 1024
gamma: 0.99
polyak_step_size: 0.005
norm_obs: false
actor_num_layers: 3
actor_layer_width: 256
actor_ln: true
deterministic: true
deterministic_eval: false
use_tanh_mean: true
use_log_std_param: false
log_std_min: -5.0
log_std_max: 2.0
num_critics: 2
critic_num_layers: 3
critic_layer_width: 256
critic_ln: true
actor_bc_coef: [0.0005, 0.001, 0.002, 0.003, 0.03, 0.1, 0.3, 1.0]
actor_q_coef: 1.0
use_q_target_in_actor: false
normalize_q_loss: true
aggregate_q: min
use_awr: false
awr_temperature: 1.0
awr_exp_adv_clip: 100.0
num_critic_updates_per_step: 2
critic_bc_coef: [0, 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.1]
diversity_coef: 0.0
policy_noise: 0.2
noise_clip: 0.5
use_target_actor: true
use_value_target: false
value_expectile: 0.8
use_entropy_loss: false
ent_coef_init: 1.0
actor_entropy_coef: 0.0
critic_entropy_coef: 0.0
```

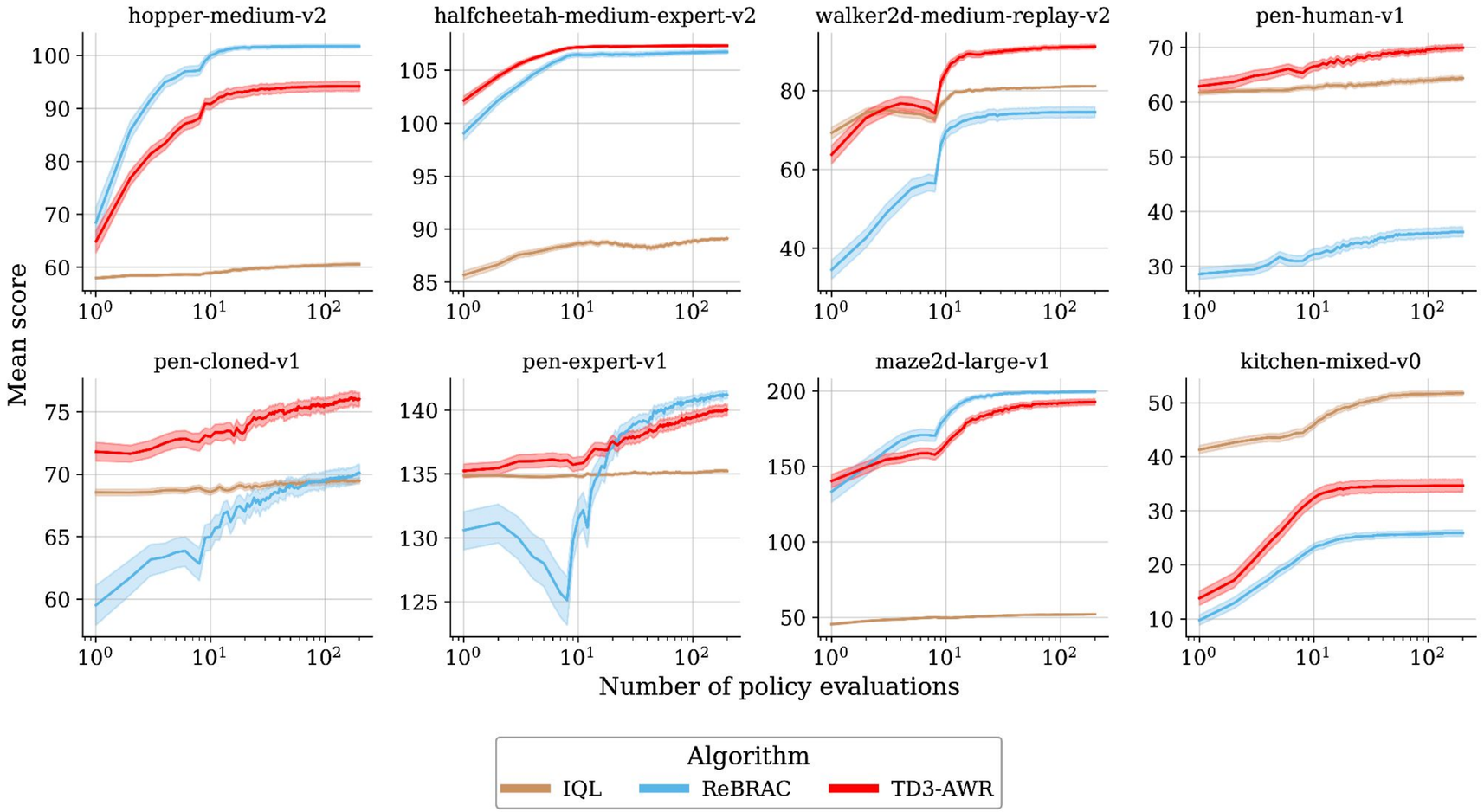
=

TD3-AWR

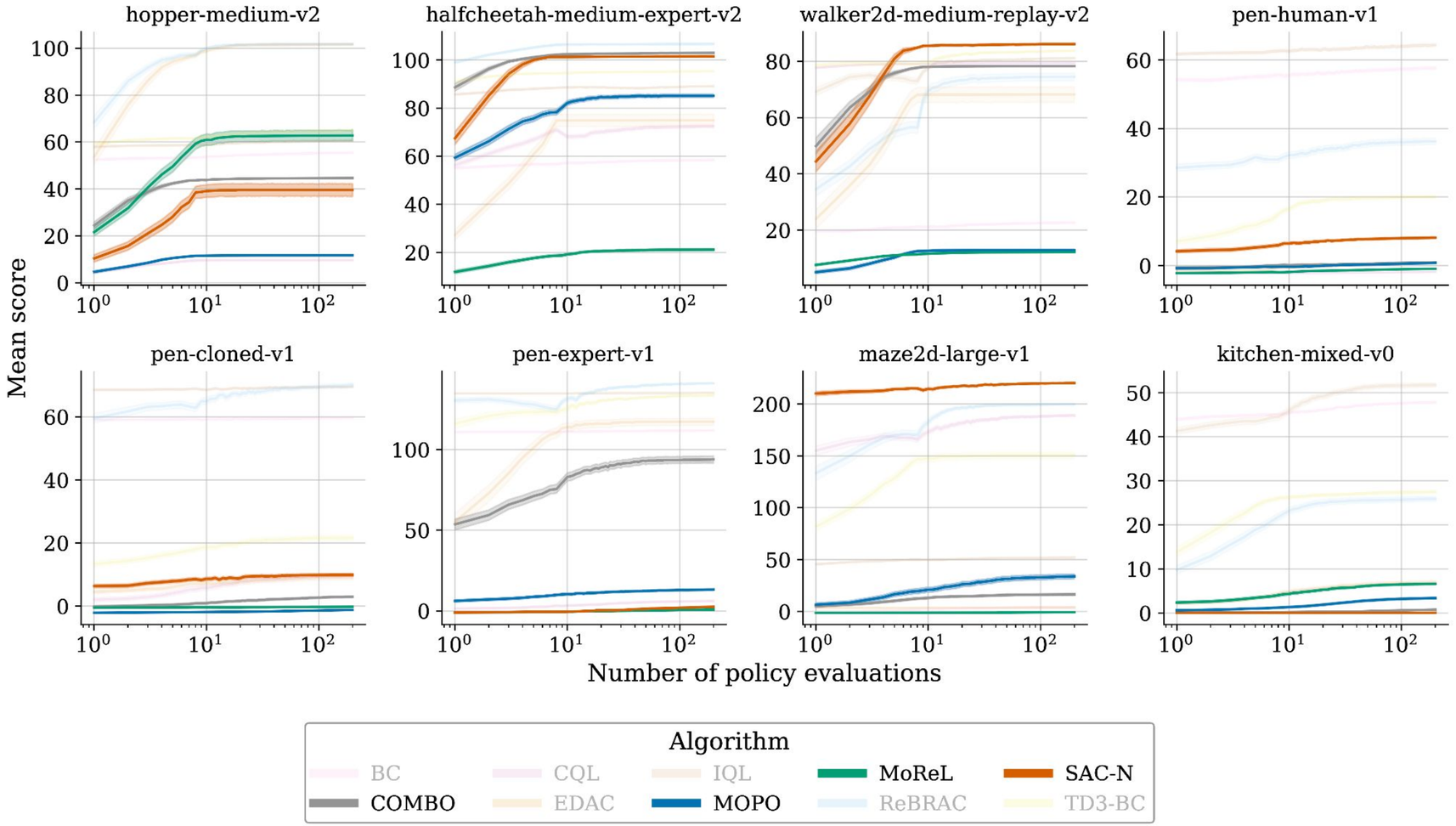
td3_awr.yaml

```yaml
lr: 1e-3
actor_lr: 1e-3
lr_schedule: constant
batch_size: 1024
gamma: 0.99
polyak_step_size: 0.005
norm_obs: false
actor_num_layers: 3
actor_layer_width: 256
actor_ln: true
deterministic: true
deterministic_eval: false
use_tanh_mean: true
use_log_std_param: false
log_std_min: -5.0
log_std_max: 2.0
num_critics: 2
critic_num_layers: 3
critic_layer_width: 256
critic_ln: true
actor_bc_coef: [0.0005, 0.001, 0.002, 0.003, 0.03, 0.1, 0.3, 1.0]
actor_q_coef: 1.0
use_q_target_in_actor: false
normalize_q_loss: true
aggregate_q: min
use_awr: true
awr_temperature: [0.5, 3.0, 10.0]
awr_exp_adv_clip: 100.0
num_critic_updates_per_step: 2
critic_bc_coef: [0, 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.1]
diversity_coef: 0.0
policy_noise: 0.2
noise_clip: 0.5
use_target_actor: true
use_value_target: false
value_expectile: [0.5, 0.7, 0.9]
use_entropy_loss: false
ent_coef_init: 1.0
actor_entropy_coef: 0.0
critic_entropy_coef: 0.0
```

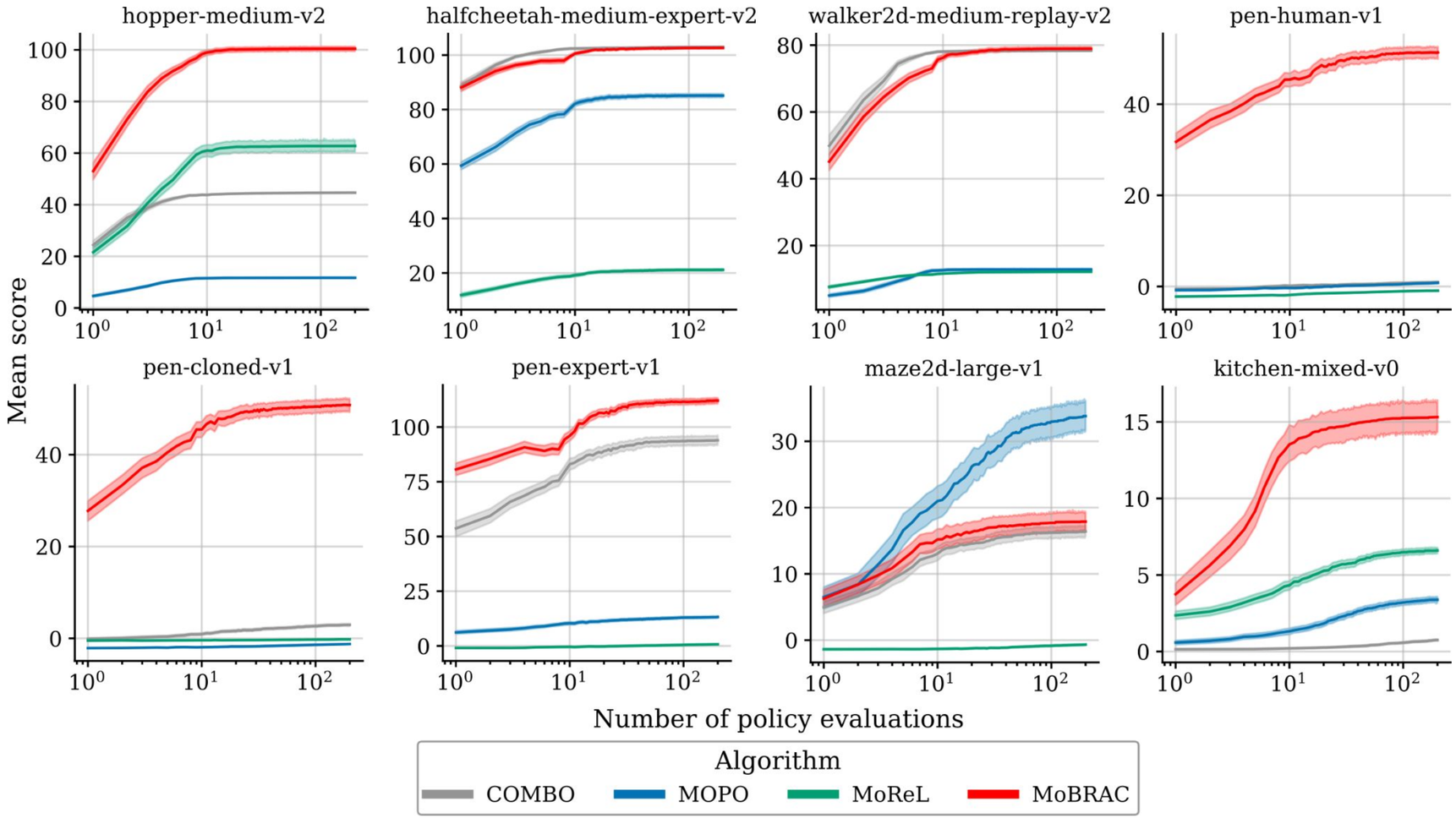
Novel Algorithms in Unifloral - TD3-AWR



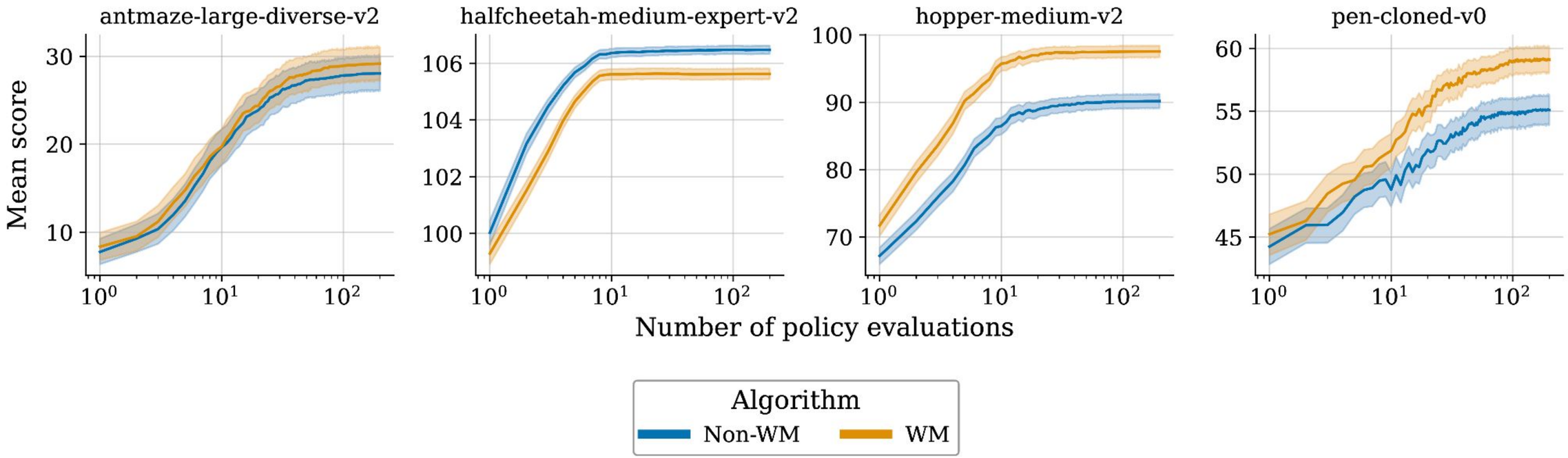
Novel Algorithms in Unifloral - MoBRAC



Novel Algorithms in Unifloral - MoBRAC



Extending Unifloral - TD3-AWR-WM



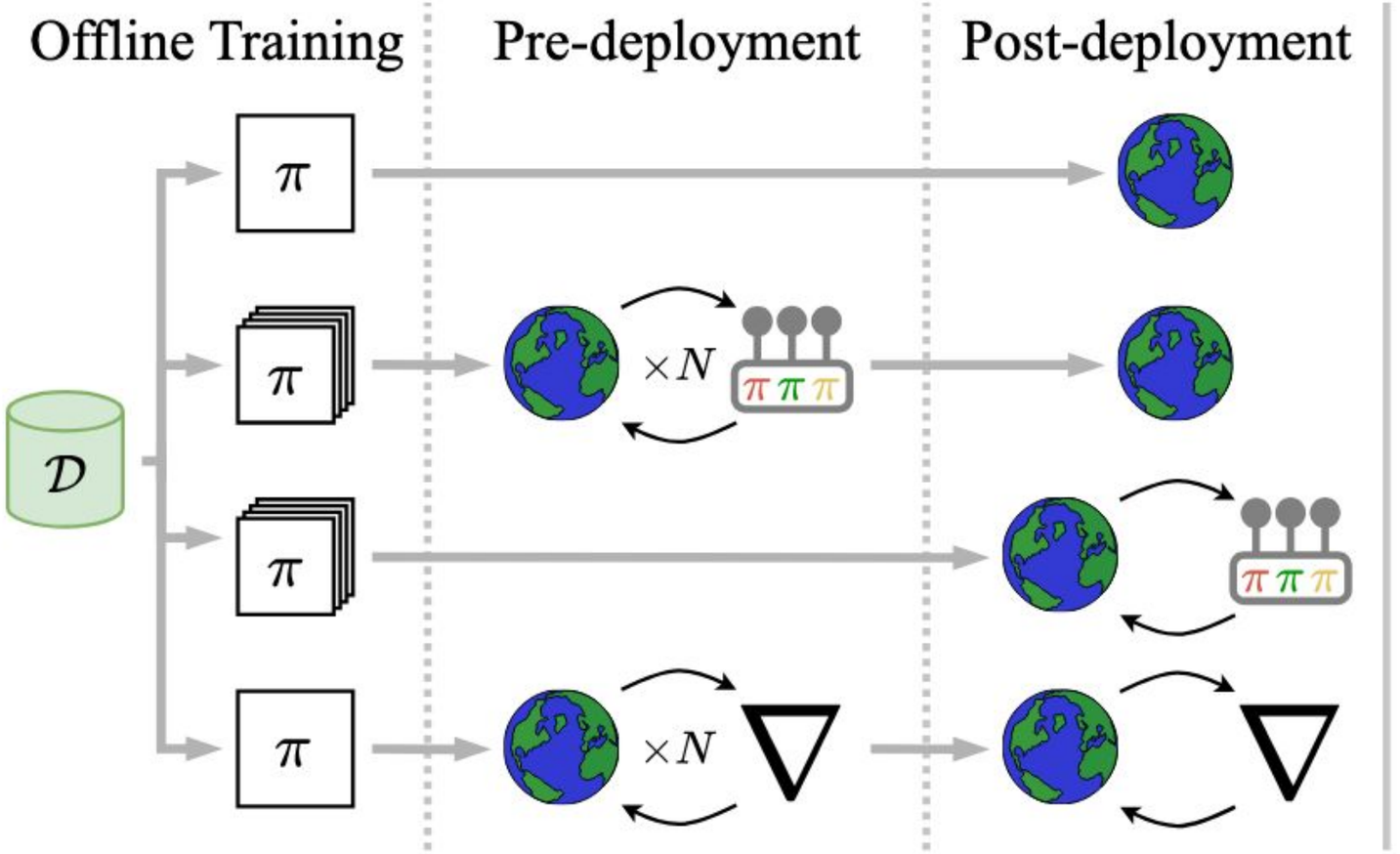
Unifloral: A Clean Slate of Offline RL

🔬 Robust, grounded evaluation

🌹 Unified algorithm space

⚡ Minimal, fast implementation

Appendix



Setting

1. Zero-shot

Example: Autonomous search and rescue

2a. Pre-deployment policy selection

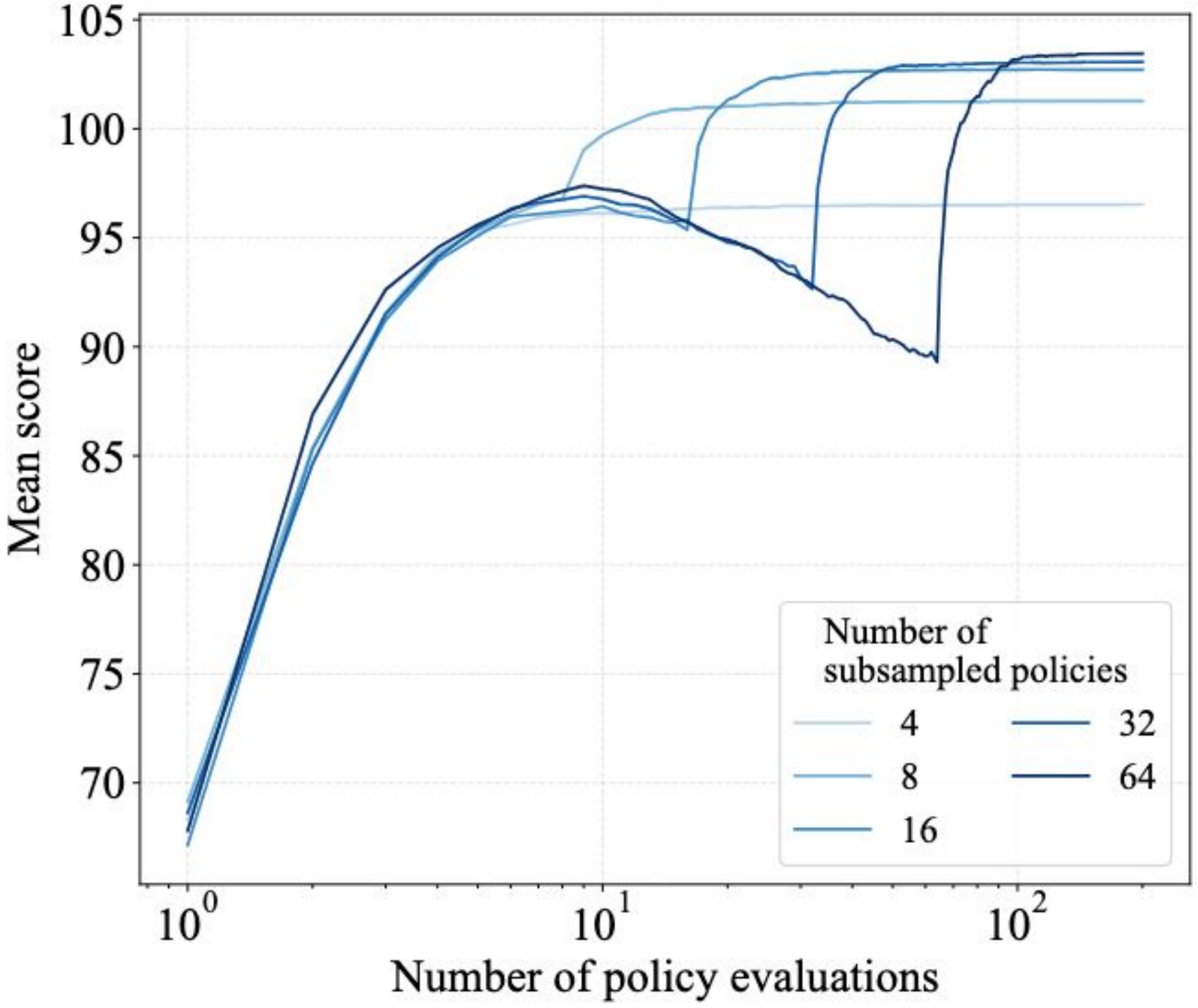
Example: Autonomous vehicles with safety driver testing

2b. Post-deployment policy selection

Example: Autonomous search and rescue

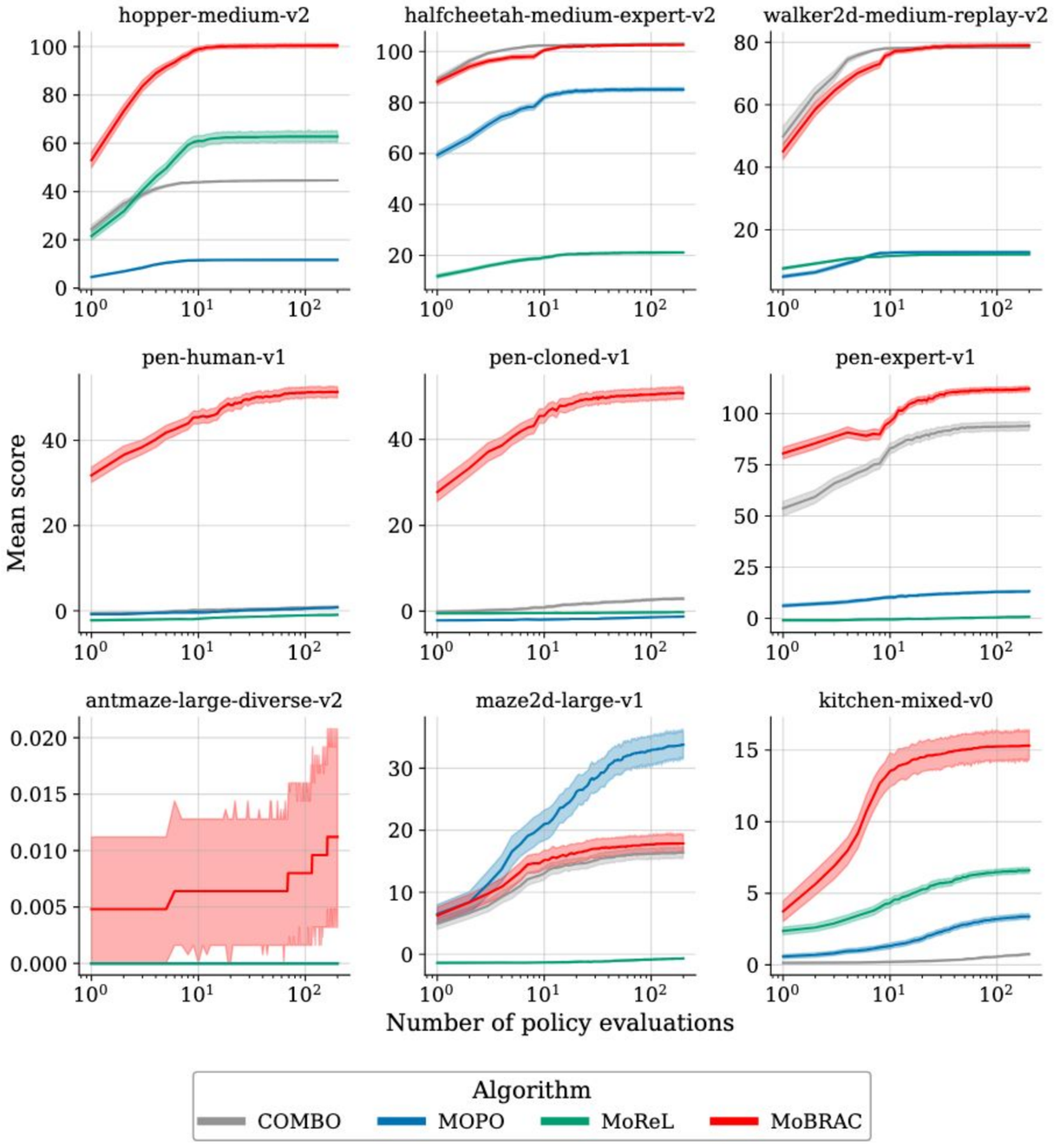
3. Offline-to-online

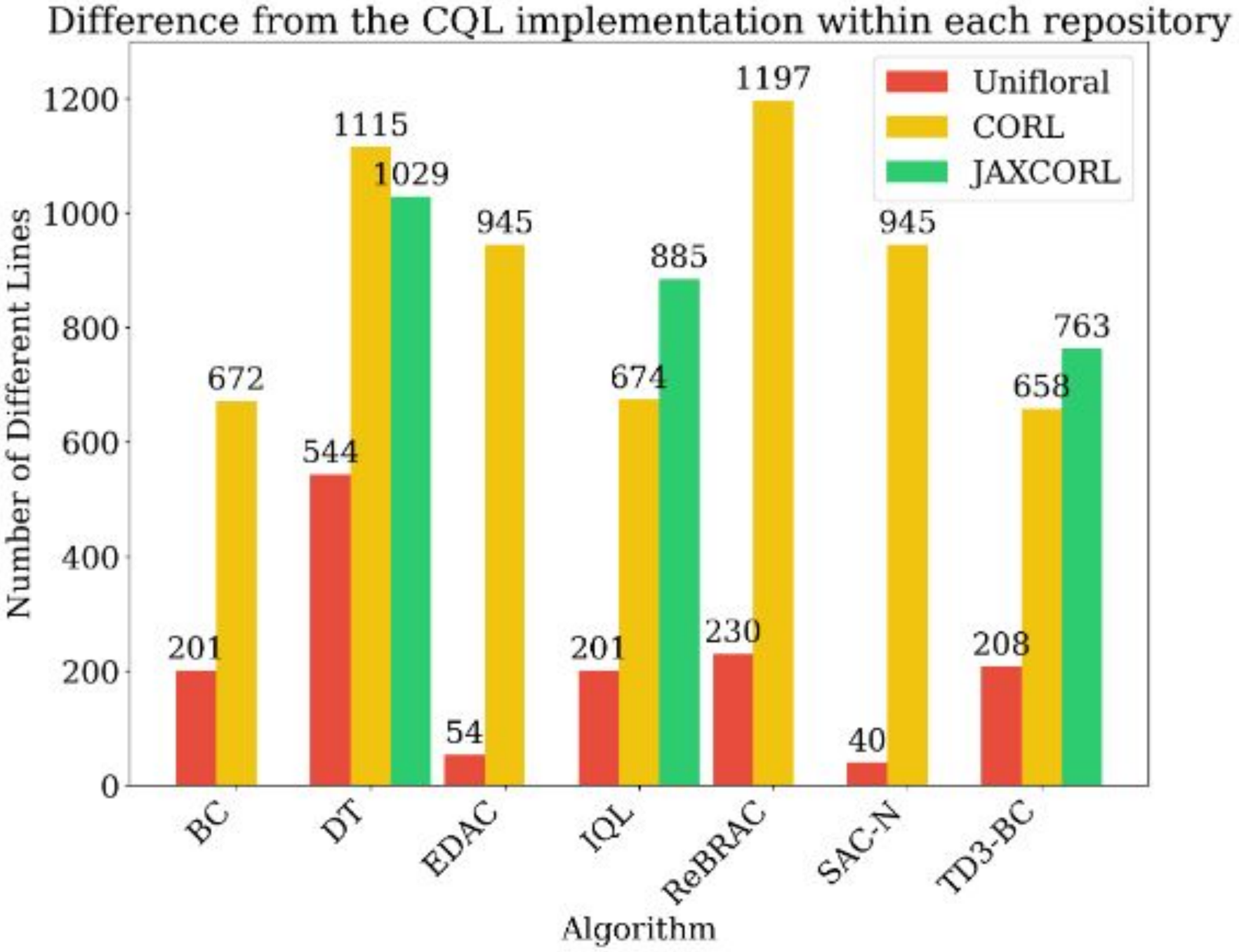
Example: Multi-step language reasoning models


105
100
95
90
85
80
75
70
Mean score
10^0
10^1
10^2
Number of policy evaluations
Number of subsampled policies
4
8
16
32
64

Algorithm	D4RL Fu et al. [26]								Atari [45]	Extra
	Locomotion	Adroit	Kitchen	Maze2d	AntMaze	Minigrid	Carla	Flow		
CQL [11]	∼	✓	✓	–	✓	–	–	–	∼	
DT [43]	∼	–	–	–	–	–	–	–	∼	KeyToDoor [43]
EDAC [15]	✓	✓	–	–	–	–	–	–	–	
IQL [38]	∼	✓	✓	–	✓	–	–	–	–	
ReBRAC [5]	✓	✓	✓	✓	✓	–	–	–	–	V-D4RL [46]
SAC-N [15]	✓	✓	–	–	–	–	–	–	–	
TD3-BC [3]	✓	–	–	–	∼	–	–	–	–	
COMBO [13]	∼	–	–	–	–	–	–	–	–	Additional MuJoCo
MOPO [2]	∼	–	–	–	–	–	–	–	–	
MOReL [10]	∼	–	–	–	–	–	–	–	–	

Table 2: Evaluations performed in the papers introducing the offline RL algorithms we consider. A "✓" indicates complete evaluation, "∼" indicates a partial evaluation, and "–" indicates that the domain was not evaluated. MuJoCo locomotion is the most widely studied domain, although random and expert datasets are often omitted. Atari experiments are limited to only 5 datasets (Breakout, Qbert, Pong, Seaquest and Asterix). Notably, the model-based offline RL works referenced here only evaluate on locomotion tasks, which may explain their dramatic performance collapse on non-locomotion tasks.


hopper-medium-v2
halfcheetah-medium-expert-v2
walker2d-medium-replay-v2
pen-human-v1
pen-cloned-v1
pen-expert-v1
antmaze-large-diverse-v2
maze2d-large-v1
kitchen-mixed-v0
Mean score
Number of policy evaluations
Algorithm
COMBO
MOPO
MoReL
MoBRAC


Difference from the CQL implementation within each repository
Unifloral
CORL
JAXCORL
Number of Different Lines
0
200
400
600
800
1000
1200
201
672
544
1115
1029
54
945
201
674
885
230
1197
40
945
208
658
763
BC
DT
EDAC
IQL
ReBRAC
SAC-N
TD3-BC
Algorithm