HNCO
Influence of the learning rate on the performance of PBIL

August 31, 2021

Abstract

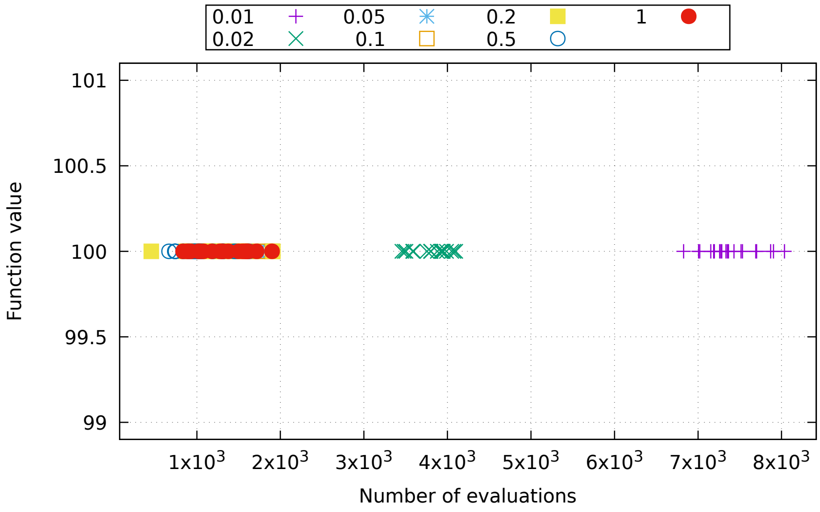
PBIL is applied many times to the same collection of fitness functions (bit vector size $n = 100$), each time with a different learning rate taken from a finite set of values. All learning rates are ranked according to their median fitness over 20 independent runs, first for each fitness function, then across the entire collection of fitness functions. The mean and standard deviation of fitness are also plotted as a function of the learning rate.

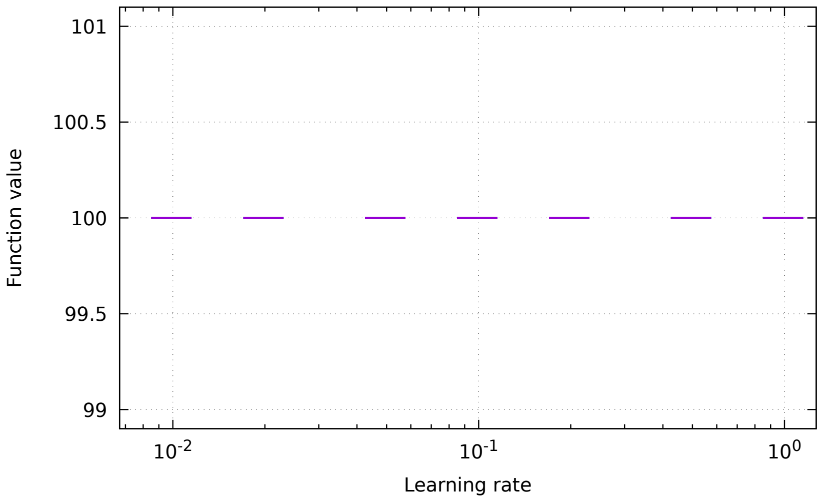
Contents

1 Global results

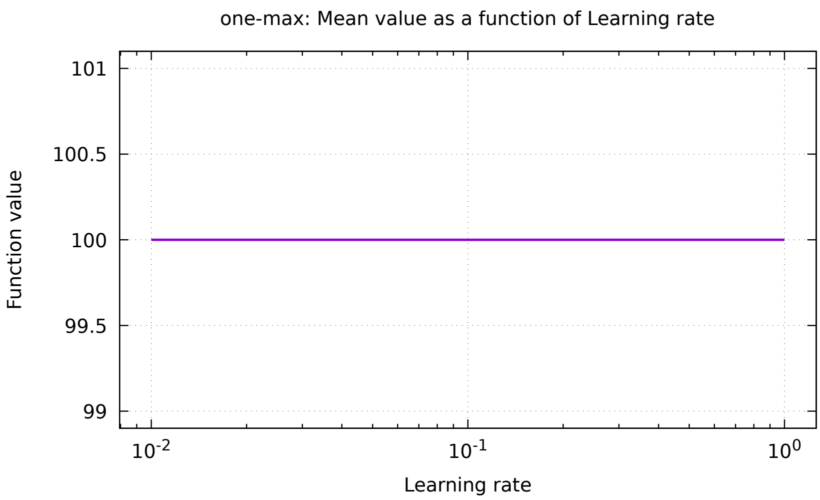
Learning rate	Rank				
	min	Q_1	med.	Q_3	max
1	1	1.00	1.0	1.00	6
0.01	1	1.00	2.0	2.00	7
0.02	1	1.00	3.0	3.00	4
0.5	1	2.00	3.0	6.00	7
0.05	1	1.00	4.0	6.00	7
0.1	1	4.00	4.0	6.00	7
0.2	1	3.00	5.0	5.00	7

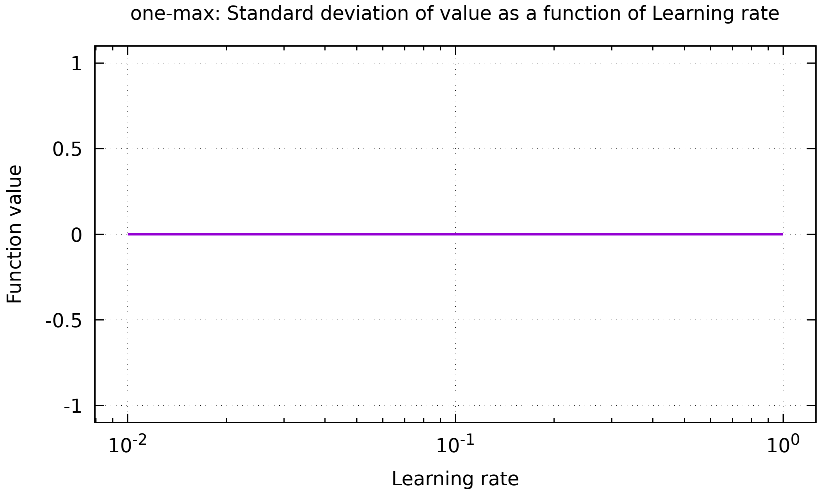
2 Function one-max



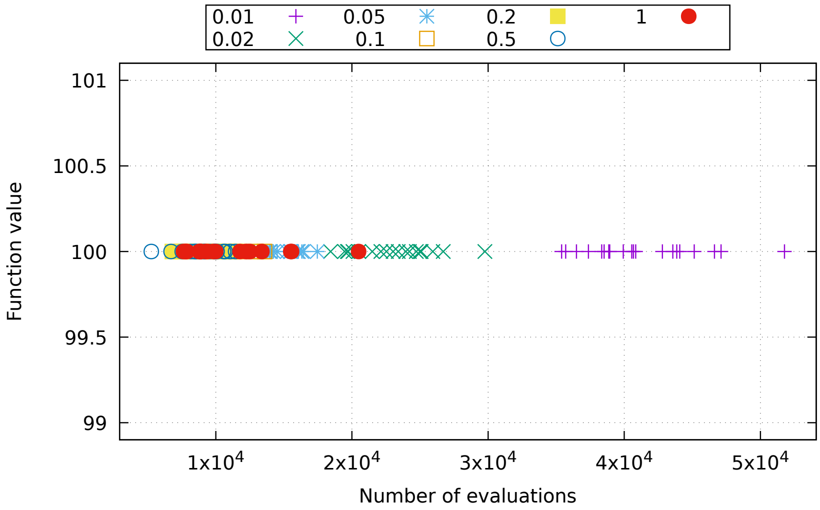


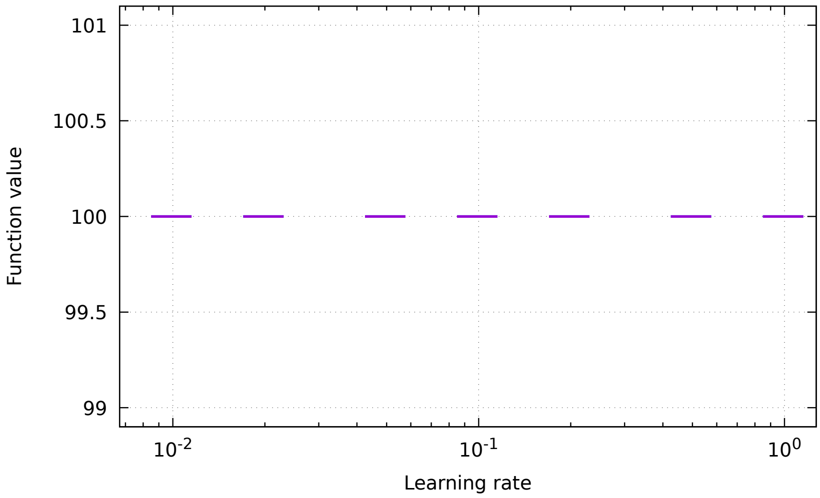
Learning rate	Function value				
	min	Q_1	med.	Q_3	max
0.01	**100**	**100.00**	**100.0**	**100.00**	**100**
0.02	**100**	**100.00**	**100.0**	**100.00**	**100**
0.05	**100**	**100.00**	**100.0**	**100.00**	**100**
0.1	**100**	**100.00**	**100.0**	**100.00**	**100**
0.2	**100**	**100.00**	**100.0**	**100.00**	**100**
0.5	**100**	**100.00**	**100.0**	**100.00**	**100**
1	**100**	**100.00**	**100.0**	**100.00**	**100**



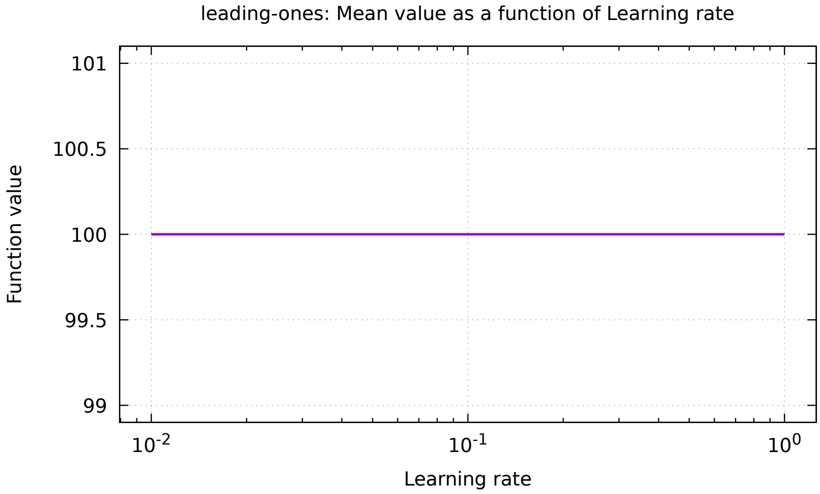

one-max: Standard deviation of value as a function of Learning rate
Function value
1
0.5
0
-0.5
-1
10^{-2}
10^{-1}
10^{0}
Learning rate

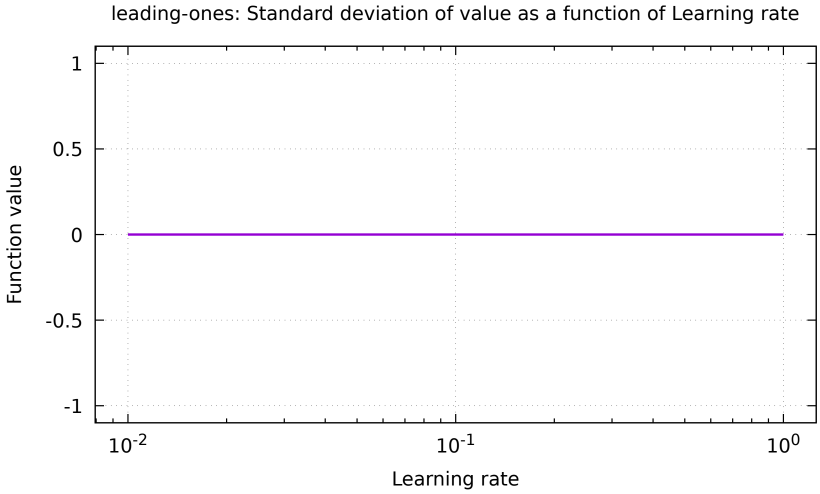
3 Function leading-ones



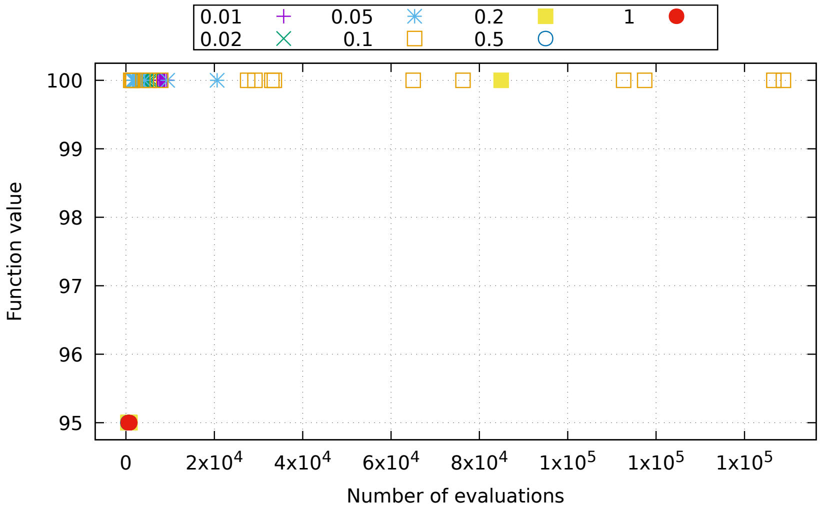


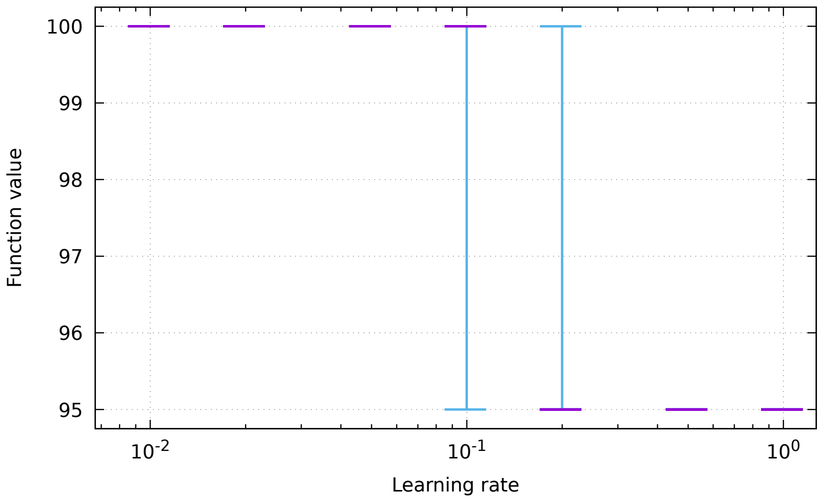
Learning rate	Function value				
	min	Q_1	med.	Q_3	max
0.01	**100**	**100.00**	**100.0**	**100.00**	**100**
0.02	**100**	**100.00**	**100.0**	**100.00**	**100**
0.05	**100**	**100.00**	**100.0**	**100.00**	**100**
0.1	**100**	**100.00**	**100.0**	**100.00**	**100**
0.2	**100**	**100.00**	**100.0**	**100.00**	**100**
0.5	**100**	**100.00**	**100.0**	**100.00**	**100**
1	**100**	**100.00**	**100.0**	**100.00**	**100**



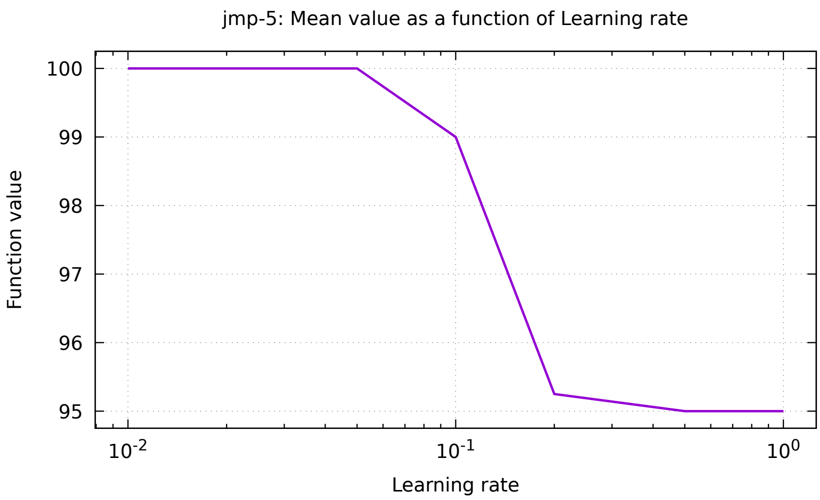

leading-ones: Standard deviation of value as a function of Learning rate
Function value
1
0.5
0
-0.5
-1
10^{-2}
10^{-1}
10^{0}
Learning rate

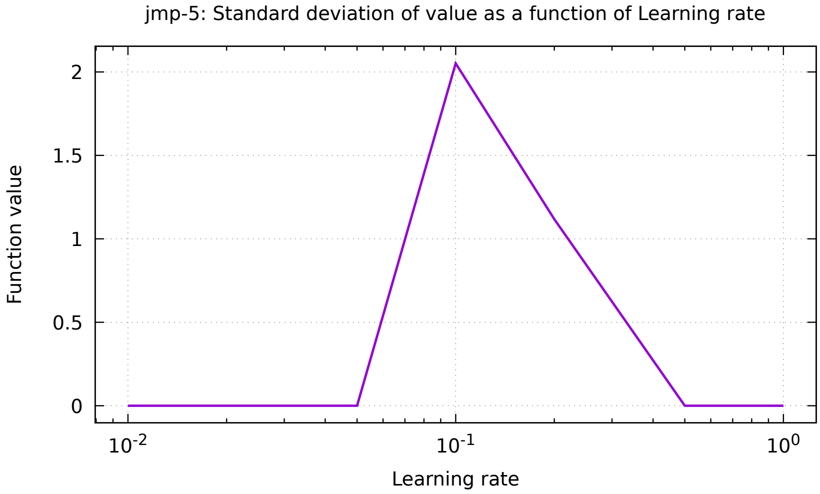
4 Function jmp-5



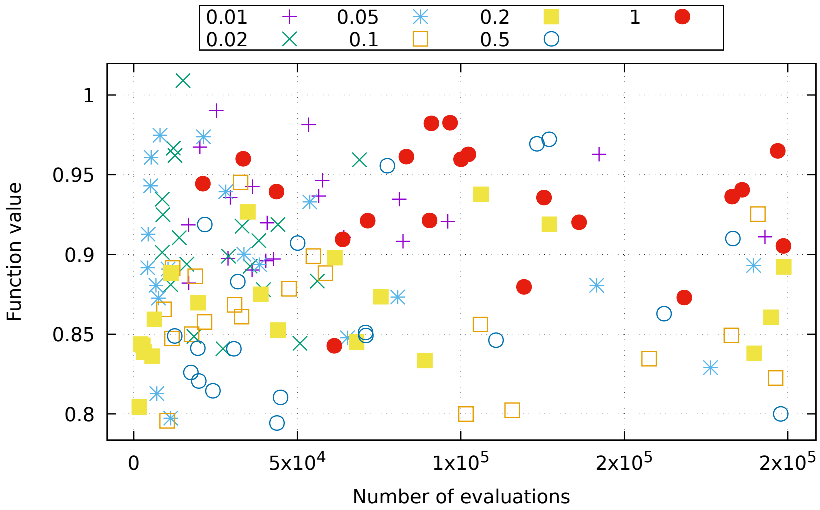


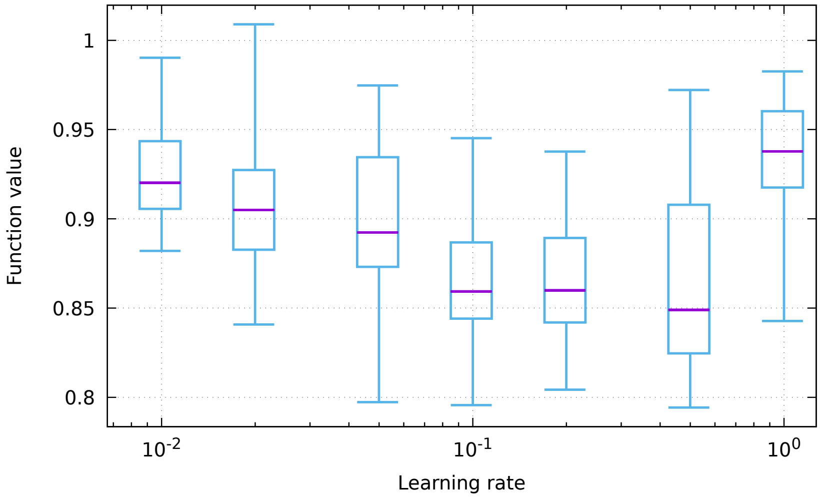
Learning rate	Function value				
	min	Q_1	med.	Q_3	max
0.01	**100**	**100.00**	**100.0**	**100.00**	**100**
0.02	**100**	**100.00**	**100.0**	**100.00**	**100**
0.05	**100**	**100.00**	**100.0**	**100.00**	**100**
0.1	95	**100.00**	**100.0**	**100.00**	**100**
0.2	95	95.00	95.0	95.00	**100**
0.5	95	95.00	95.0	95.00	95
1	95	95.00	95.0	95.00	95




jmp-5: Standard deviation of value as a function of Learning rate
Function value
2
1.5
1
0.5
0
10^-2
10^-1
10^0
Learning rate

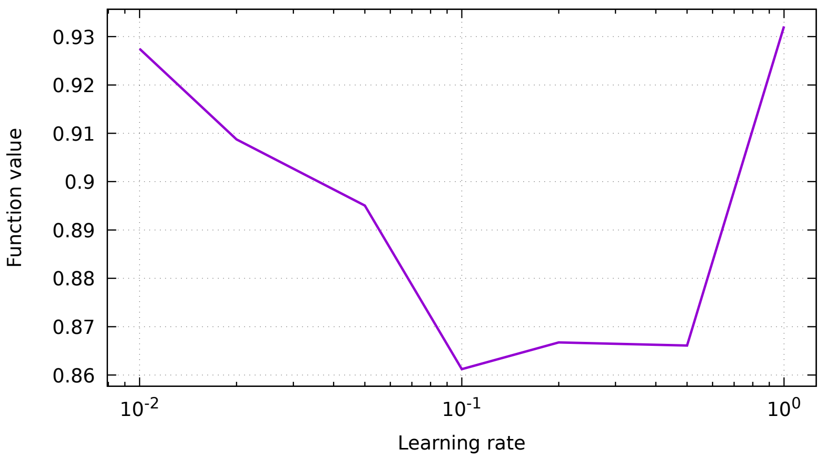
5 Function nk

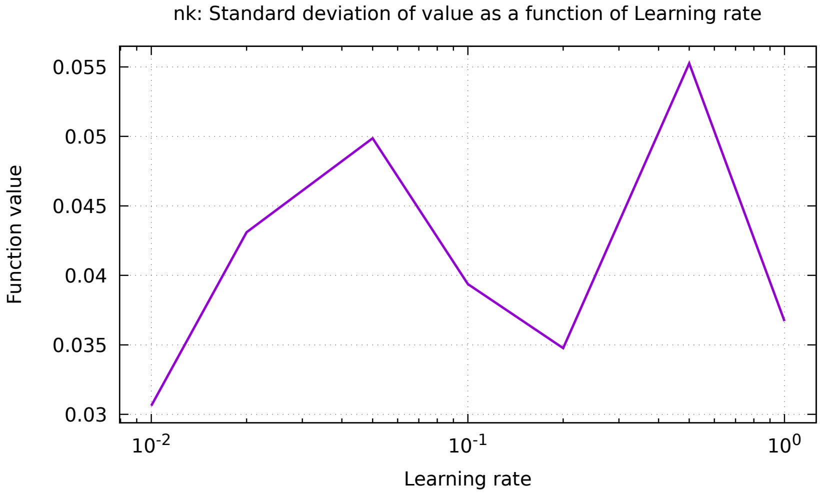




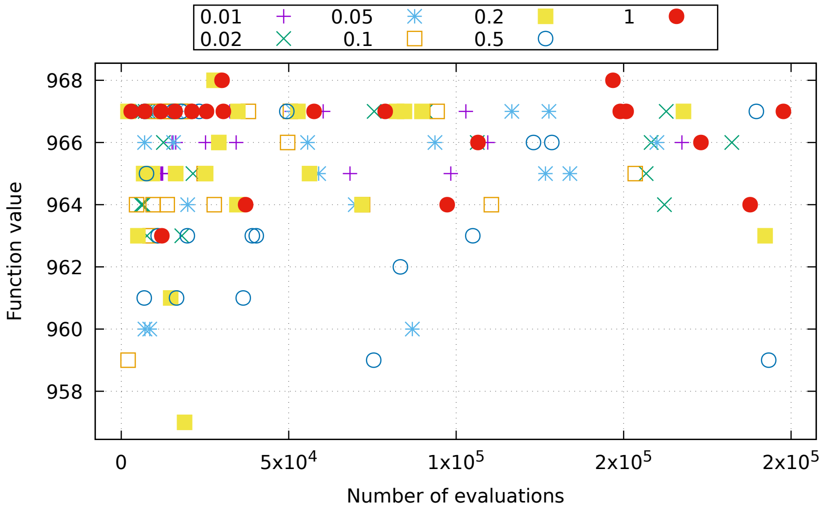
Learning rate	Function value				
	min	Q_1	med.	Q_3	max
1	0.84	**0.917,5**	**0.938**	**0.960,3**	0.98
0.01	**0.88**	0.905,6	0.920	0.943,5	0.99
0.02	0.84	0.882,7	0.905	0.927,3	**1.01**
0.05	0.80	0.873,1	0.892	0.934,5	0.97
0.2	0.80	0.842,0	0.860	0.889,3	0.94
0.1	0.80	0.844,1	0.859	0.886,8	0.95
0.5	0.79	0.824,7	0.849	0.907,9	0.97

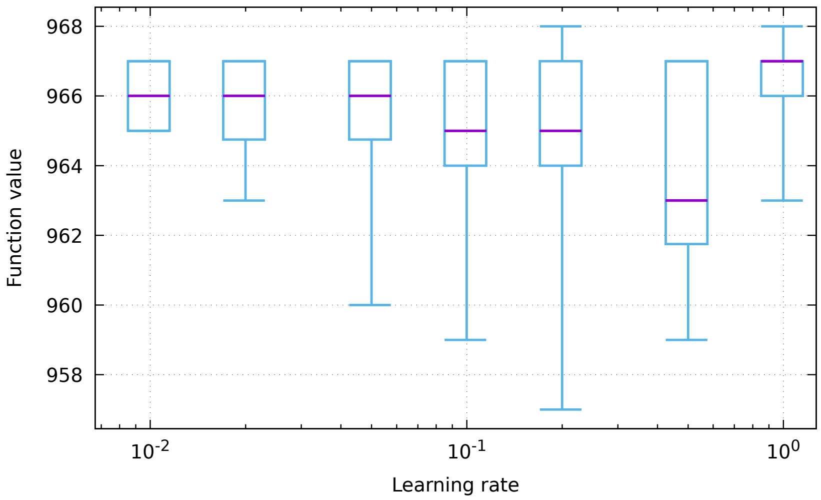
nk: Mean value as a function of Learning rate



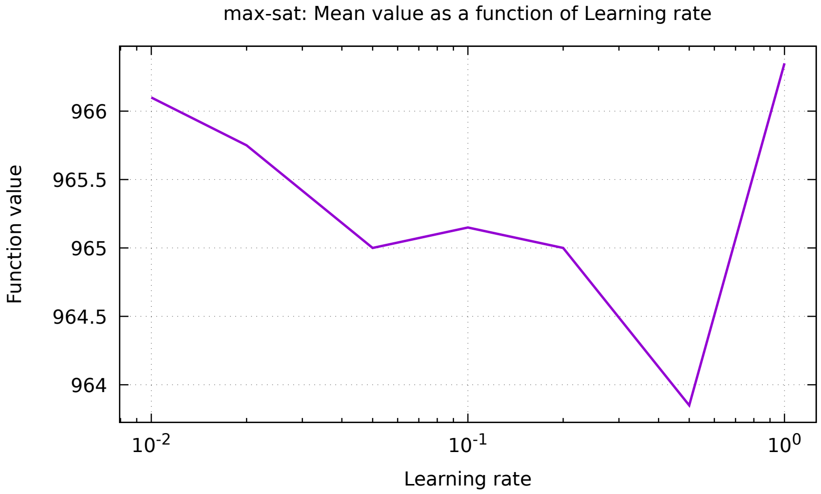

nk: Standard deviation of value as a function of Learning rate
Function value
0.055
0.05
0.045
0.04
0.035
0.03
10^{-2}
10^{-1}
10^{0}
Learning rate

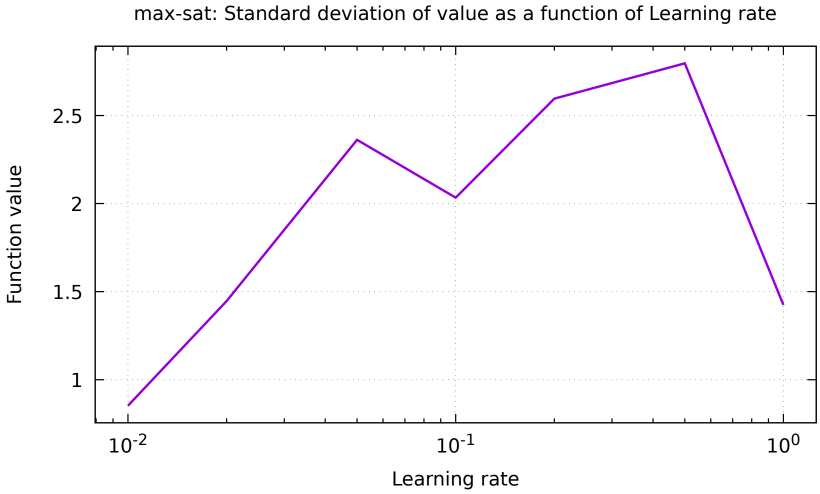
6 Function max-sat



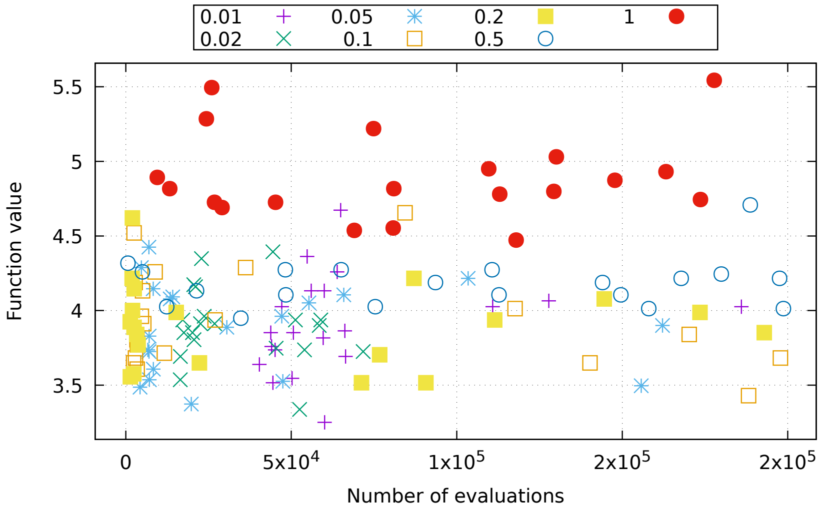


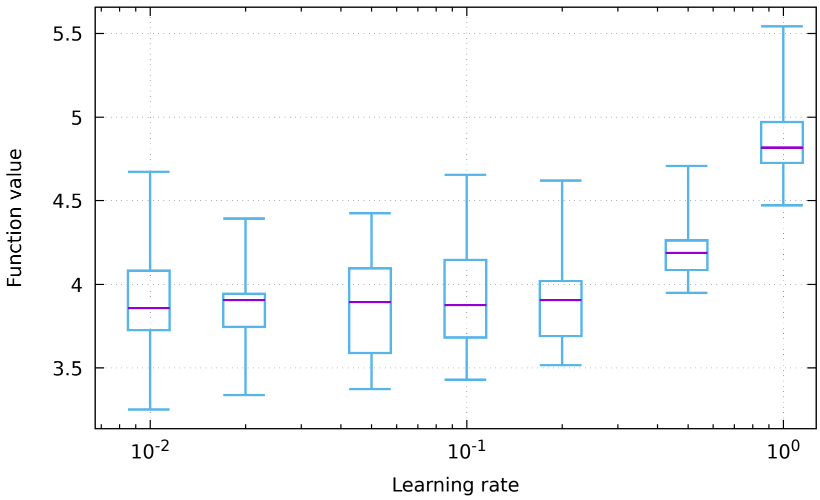
Learning rate	Function value				
	min	Q_1	med.	Q_3	max
1	963	**966.00**	**967.0**	**967.00**	**968**
0.01	**965**	965.00	966.0	**967.00**	967
0.02	963	964.75	966.0	**967.00**	967
0.05	960	964.75	966.0	**967.00**	967
0.2	957	964.00	965.0	**967.00**	**968**
0.1	959	964.00	965.0	**967.00**	967
0.5	959	961.75	963.0	**967.00**	967



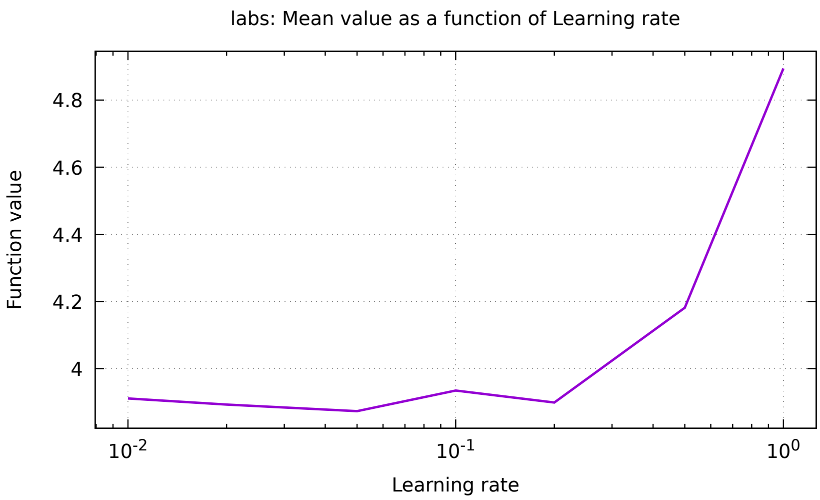

max-sat: Standard deviation of value as a function of Learning rate
Function value
2.5
2
1.5
1
10^{-2}
10^{-1}
10^{0}
Learning rate

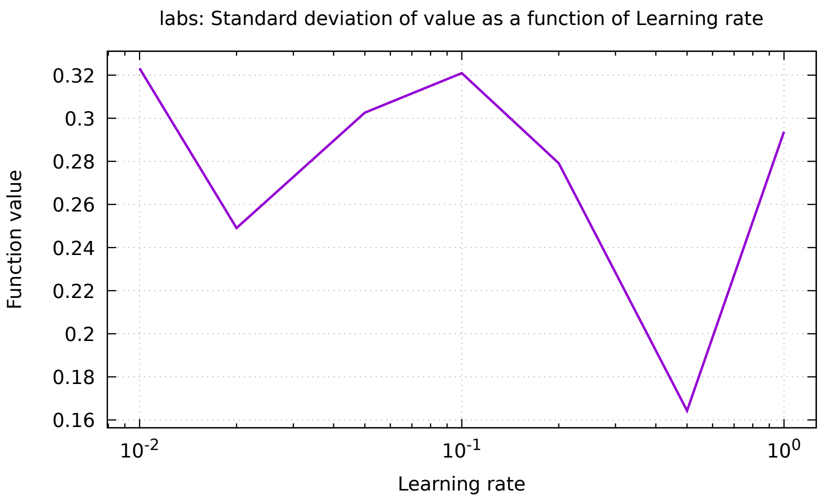
7 Function labs



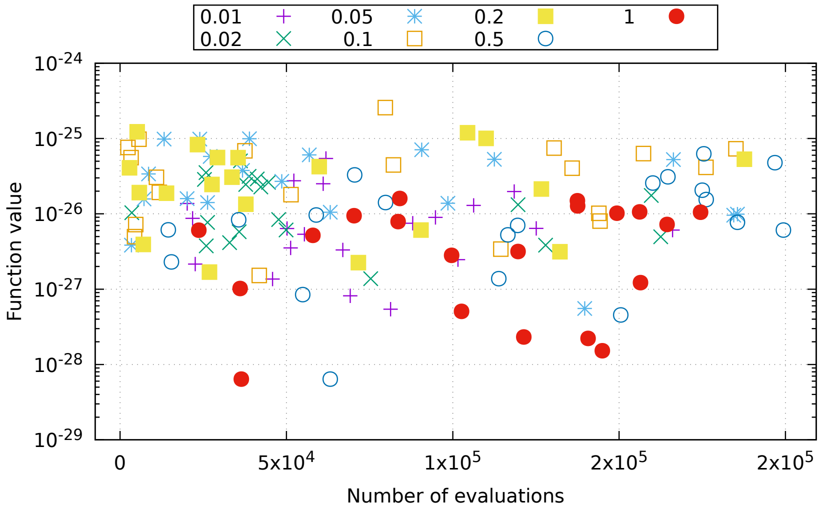


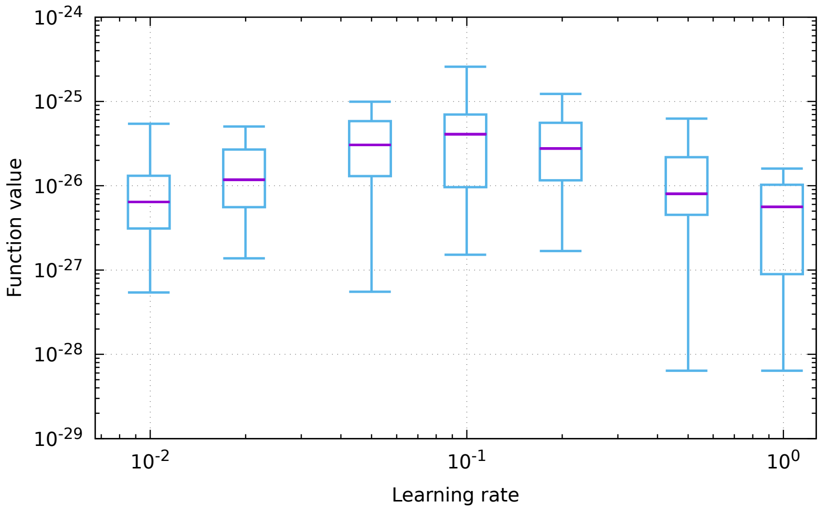
Learning rate	Function value				
	min	Q_1	med.	Q_3	max
1	**4.47**	**4.725,9**	**4.817**	**4.970,4**	**5.54**
0.5	3.95	4.085,3	4.188	4.262,6	4.71
0.2	3.52	3.690,2	3.906	4.019,6	4.62
0.02	3.34	3.745,3	3.906	3.943,3	4.39
0.05	3.37	3.589,6	3.894	4.095,0	4.42
0.1	3.43	3.681,9	3.876	4.146,1	4.66
0.01	3.25	3.725,9	3.858	4.081,8	4.67



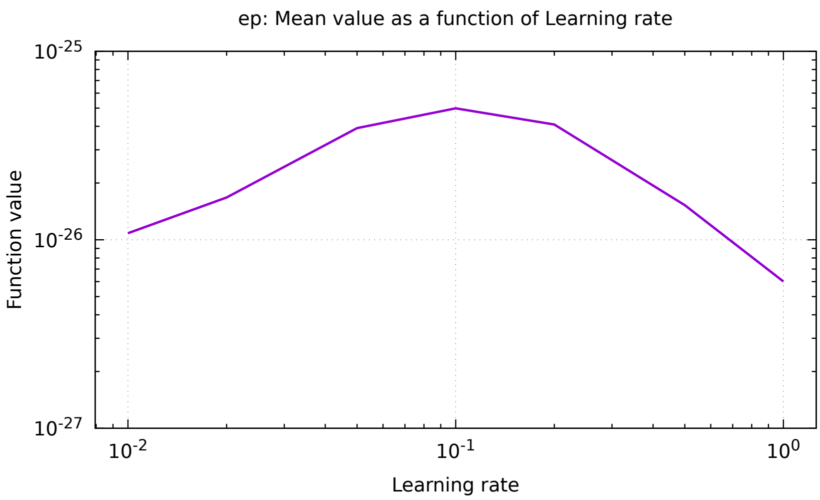

labs: Standard deviation of value as a function of Learning rate
Function value
0.32
0.3
0.28
0.26
0.24
0.22
0.2
0.18
0.16
10^-2
10^-1
10^0
Learning rate

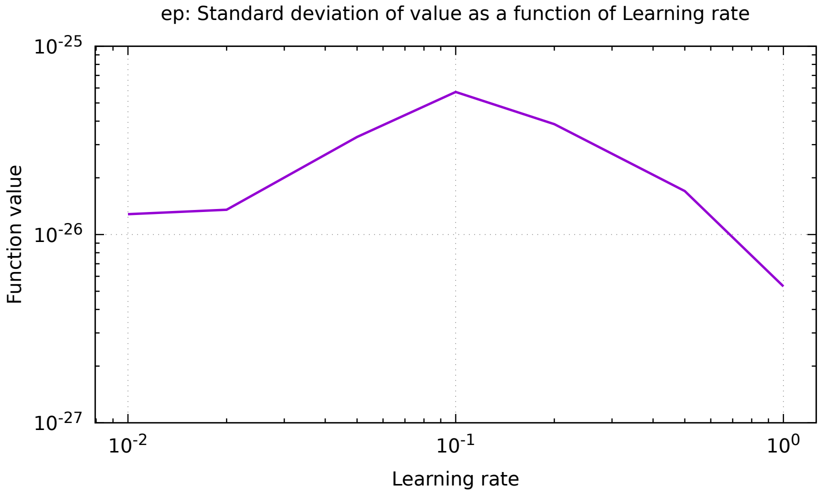
8 Function ep



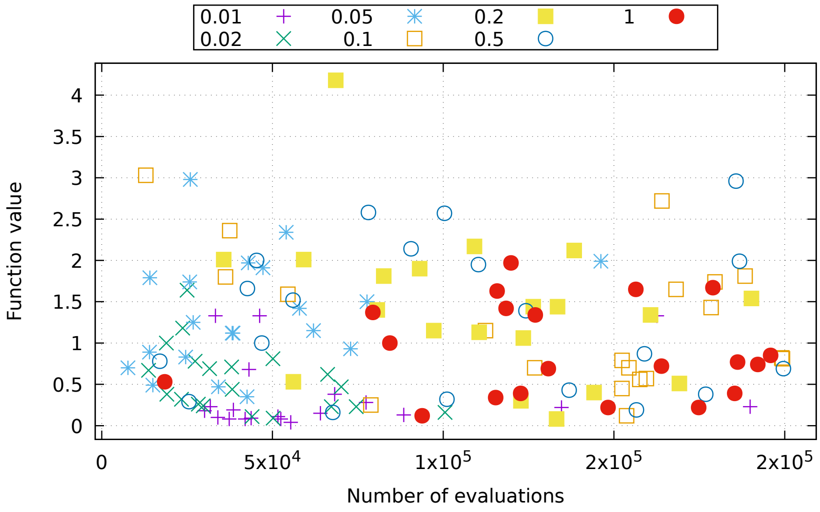


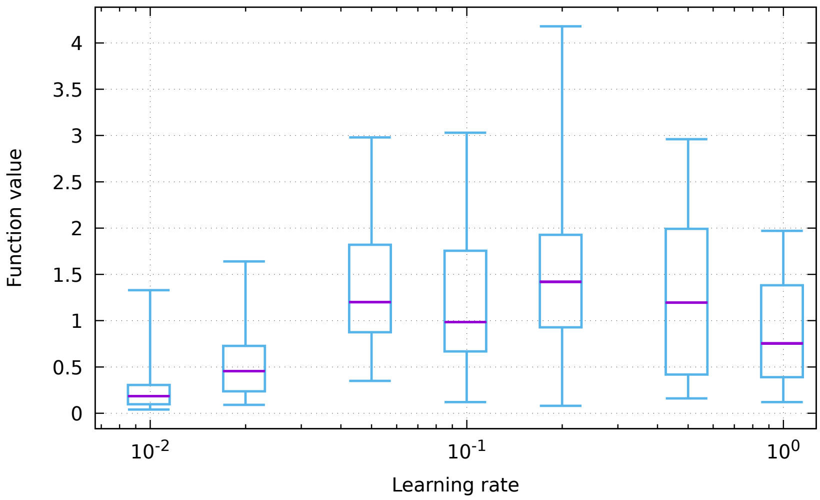
Learning rate	Function value				
	min	Q_1	med.	Q_3	max
1	$\mathbf{6.4 \times 10^{-29}}$	$\mathbf{8.927 \times 10^{-28}}$	$\mathbf{5.62 \times 10^{-27}}$	$\mathbf{1.028 \times 10^{-26}}$	$\mathbf{1.6 \times 10^{-26}}$
0.01	5.4×10^{-28}	3.108×10^{-27}	6.40×10^{-27}	1.314×10^{-26}	5.4×10^{-26}
0.5	6.4×10^{-29}	4.523×10^{-27}	8.04×10^{-27}	2.183×10^{-26}	6.3×10^{-26}
0.02	1.4×10^{-27}	5.575×10^{-27}	1.18×10^{-26}	2.686×10^{-26}	5.0×10^{-26}
0.2	1.7×10^{-27}	1.160×10^{-26}	2.76×10^{-26}	5.574×10^{-26}	1.2×10^{-25}
0.05	5.5×10^{-28}	1.298×10^{-26}	3.04×10^{-26}	5.844×10^{-26}	9.9×10^{-26}
0.1	1.5×10^{-27}	9.633×10^{-27}	4.09×10^{-26}	6.983×10^{-26}	2.6×10^{-25}



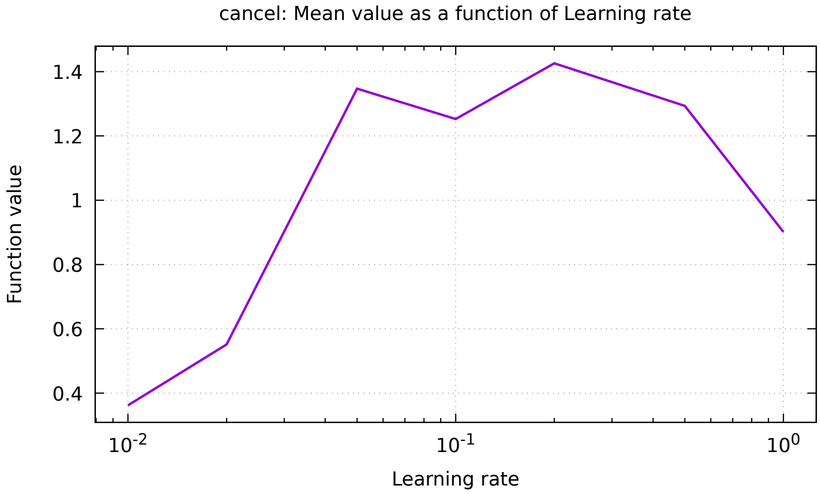

ep: Standard deviation of value as a function of Learning rate
Function value
Learning rate
10^-25
10^-26
10^-27
10^-2
10^-1
10^0

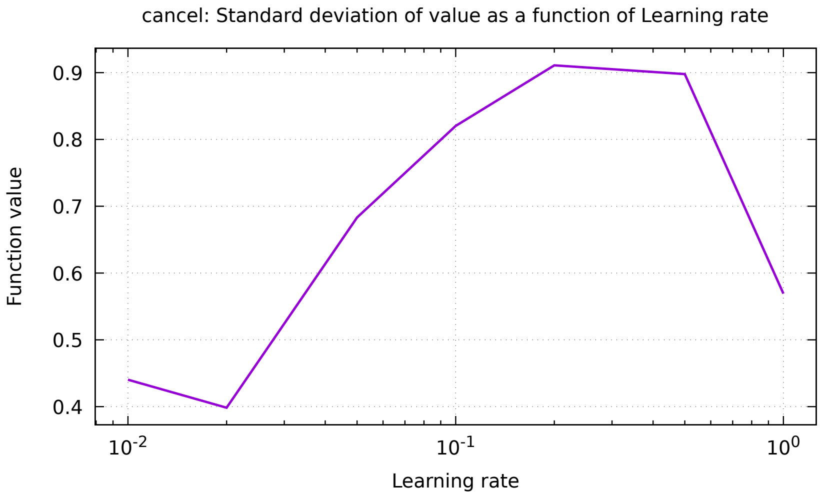
9 Function cancel



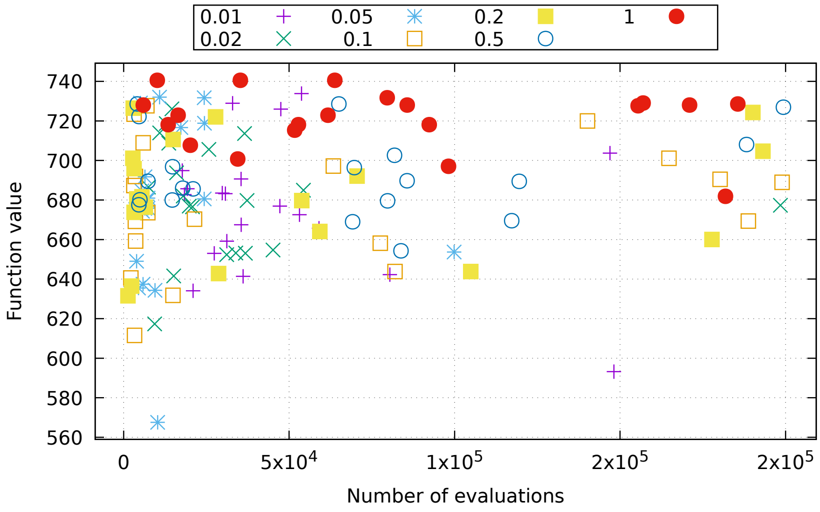


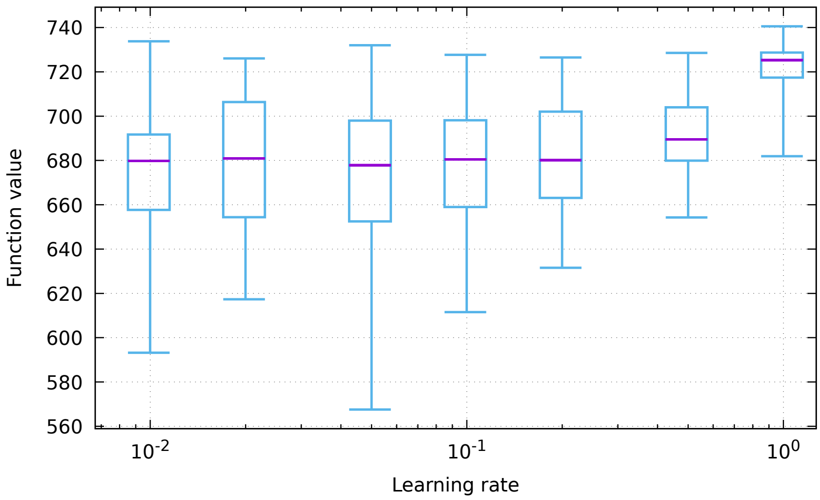
Learning rate	Function value				
	min	Q_1	med.	Q_3	max
0.01	**0.04**	**0.097,5**	**0.185**	**0.305,0**	**1.33**
0.02	0.09	0.237,5	0.455	0.727,5	1.64
1	0.12	0.390,0	0.755	1.382,5	1.97
0.1	0.12	0.667,5	0.985	1.755,0	3.03
0.5	0.16	0.417,5	1.195	1.992,5	2.96
0.05	0.35	0.875,0	1.200	1.820,0	2.98
0.2	0.08	0.927,5	1.420	1.927,5	4.18




cancel: Standard deviation of value as a function of Learning rate
Function value
0.9
0.8
0.7
0.6
0.5
0.4
10^{-2}
10^{-1}
10^{0}
Learning rate

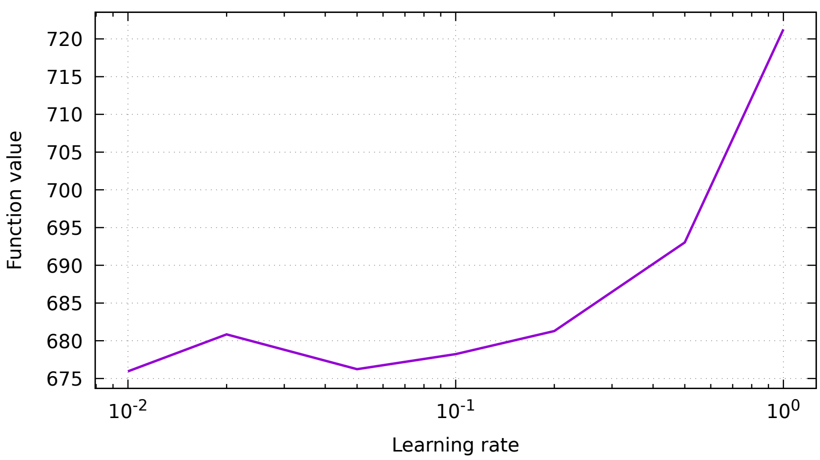
10 Function walsh2

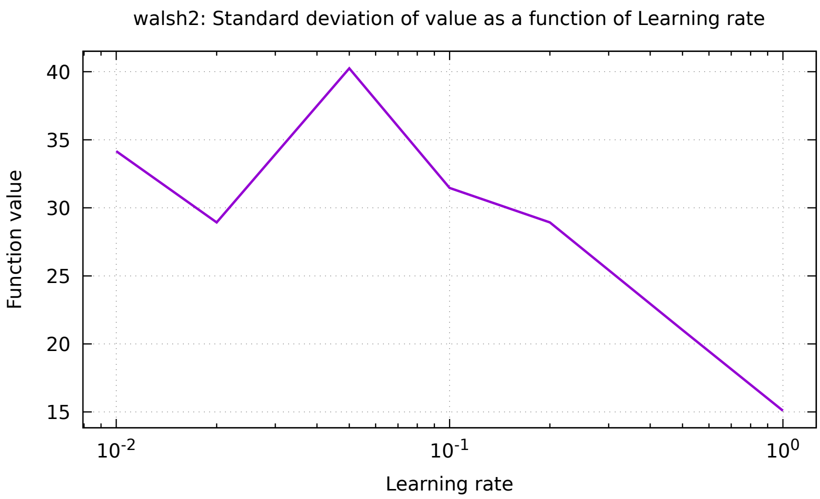




Learning rate	Function value				
	min	Q_1	med.	Q_3	max
1	**681.90**	**717.404,8**	**725.264**	**728.678,5**	**740.55**
0.5	654.29	679.893,3	689.477	703.958,3	728.55
0.02	617.32	654.363,8	680.890	706.381,8	726.07
0.1	611.54	658.972,0	680.462	698.162,8	727.67
0.2	631.53	663.066,8	680.143	702.004,5	726.48
0.01	593.18	657.673,3	679.825	691.700,0	733.81
0.05	567.56	652.509,0	677.868	697.966,5	732.02

walsh2: Mean value as a function of Learning rate





A Plan

```
{
    "exec": "hnco",
    "opt": "-A 500 -x 10 -y 1 --print-results --map 1 --map-random -s 100",
    "budget": 200000,
    "num_runs": 20,
    "parallel": true,
    "parameter": {
        "id": "learning-rate",
        "name": "Learning rate",
        "values": [ 1e-2, 2e-2, 5e-2, 1e-1, 2e-1, 5e-1, 1 ]
    },
    "graphics": {
        "logscale": true,
        "candlesticks": {
            "title": true,
            "boxwidth": "$1 * 0.3"
        }
    },
    "functions": [
        {
            "id": "one-max",
            "opt": "-F 0 --stop-on-maximum",
            "rounding": {
                "value": { "before": 3, "after": 0 },
                "time": { "before": 1, "after": 2 } }
        },
        {
            "id": "leading-ones",
            "opt": "-F 10 --stop-on-maximum",
            "rounding": {
                "value": { "before": 3, "after": 0 },
                "time": { "before": 1, "after": 2 } }
        },
        {
            "id": "jmp-5",
            "opt": "-F 30 --stop-on-maximum -t 5",
            "rounding": {
                "value": { "before": 3, "after": 0 },
                "time": { "before": 1, "after": 2 } }
        },
        {
            "id": "nk",
            "opt": "-F 60 -p instances/nk.100.4",
```

```
            "rounding": {
                "value": { "before": 1, "after": 2 },
                "time": { "before": 1, "after": 2 } }
        },
        {
            "id": "max-sat",
            "opt": "-F 70 -p instances/ms.100.3.1000",
            "rounding": {
                "value": { "before": 3, "after": 0 },
                "time": { "before": 1, "after": 2 } }
        },
        {
            "id": "labs",
            "opt": "-F 81",
            "rounding": {
                "value": { "before": 1, "after": 2 },
                "time": { "before": 1, "after": 2 } }
        },
        {
            "id": "ep",
            "opt": "-F 90 -p instances/ep.100",
            "reverse": true,
            "logscale": true,
            "rounding": {
                "value": { "before": 1, "after": 1 },
                "time": { "before": 1, "after": 2 } }
        },
        {
            "id": "cancel",
            "opt": "-F 100 -s 99",
            "reverse": true,
            "rounding": {
                "value": { "before": 1, "after": 2 },
                "time": { "before": 1, "after": 2 } }
        },
        {
            "id": "walsh2",
            "opt": "-F 162 -p instances/walsh2.100",
            "rounding": {
                "value": { "before": 3, "after": 2 },
                "time": { "before": 1, "after": 2 } }
        }
    ]
}
```


B Default parameters

```
# algorithm = 100
# bm_mc_reset_strategy = 1
# bm_num_gs_cycles = 1
# bm_num_gs_steps = 100
# bm_sampling = 1
# budget = 10000
# bv_size = 100
# description_path = description.txt
# ea_lambda = 100
# ea_mu = 10
# expression = x
# fn_name = noname
# fn_num_traps = 10
# fn_prefix_length = 2
# fn_threshold = 10
```

```
# fp_expression = (1-x)^2+100*(y-x^2)^2
# fp_lower_bound = -2
# fp_num_bits = 8
# fp_precision = 0.01
# fp_upper_bound = 2
# function = 0
# ga_crossover_bias = 0.5
# ga_crossover_probability = 0.5
# ga_tournament_size = 10
# hea_bit_herding = 0
# hea_num_seq_updates = 100
# hea_reset_period = 0
# hea_sampling_method = 0
# hea_weight = 1
# learning_rate = 0.001
# map = 0
# map_input_size = 100
# map_path = map.txt
# map_ts_length = 10
# map_ts_sampling_mode = 0
# mutation_rate = 1
# neighborhood = 0
# neighborhood_iterator = 0
# noise_stddev = 1
# num_iterations = 0
# num_threads = 1
# path = function.txt
# pn_mutation_rate = 1
# pn_neighborhood = 0
# pn_radius = 2
# population_size = 10
# pv_log_num_components = 5
# radius = 2
# rep_categorical_representation = 0
# results_path = results.json
# rls_patience = 50
# sa_beta_ratio = 1.2
# sa_initial_acceptance_probability = 0.6
# sa_num_transitions = 50
# sa_num_trials = 100
# seed = 0
# selection_size = 1
# solution_path = solution.txt
# target = 100
# print_defaults
# last_parameter
# exec_name = hnco
# version = 0.18
# Generated from hnco.json
```